Parallel Programming with Intel Parallel Studio

Intel Parallel Studio 环境下的并行程序设计

Stephen Blair-Chappell
Andrew Stokes 著

罗秋明 孔畅 刘成健 等译

清华大学出版社

Parallel Programming with Intel Parallel Studio

Intel Parallel Studio 环境下的并行程序设计

Stephen Blair-Chappell ...
Andrew Stokes

... 畅 刘成健 等译

清华大学出版社
北京

Stephen Blair-Chappell, Andrew Stokes
Parallel Programming with Intel Parallel Studio
EISBN: 9780470891650

图书在版编目（CIP）数据

Intel Parallel Studio 环境下的并行程序设计/（美）布莱尔查普尔（Blair-Chappell, S.），斯托克斯（Stokes, A.）著；罗秋明等译. --北京：清华大学出版社，2013.4
ISBN 978-7-302-30976-5

Ⅰ. ①I…　Ⅱ. ①布…　②斯…　③罗…　Ⅲ. ①并行程序–程序设计　Ⅳ. ①TP311.11

中国版本图书馆 CIP 数据核字（2012）第 301683 号

责任编辑： 龙启铭
封面设计： 何凤霞
责任校对： 时翠兰
责任印制： 王静怡

出版发行： 清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社　总　机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者： 北京鑫海金澳胶印有限公司
经　　销： 全国新华书店
开　　本： 185mm×260mm　　**印　　张：** 28.25　　**字　　数：** 673 千字
版　　次： 2013 年 4 月第 1 版　　**印　　次：** 2013 年 4 月第 1 次印刷
印　　数： 1～3000
定　　价： 69.00 元

产品编号：041697-01

译 者 序

并行计算因多核处理器的出现和普及已逐步从高校和研究机构走向大众。虽然并行编程技术渐渐被更多的人所掌握，编写并行应用程序也不再神秘，但大多数编程开发人员对串行程序的并行改造流程未必有完整和清晰的认识，而且相应的并行开发和调试工具也依旧不令人满意。

首先说说工具问题。如果缺少相应工具的支持，很难判断自己编写和生成的代码是否可以充分利用硬件资源，从而不能确保程序按预期的并行方式工作，甚至出现并行代码比串行代码的性能还要低的情况。在开源领域里有一些不错的工具，可以做调试开发，例如用 GDB、Oprofile、Valgrind 等工具协助查找问题，但是这些工具的集成性不太好，使用起来并不那么方便。Intel 的 Parallel Studio XE 相对上述未整合的开源工具而言，方便性提升了很多（虽然也不是尽善尽美，仍在发展完善中）。因此当清华大学出版社的龙启铭编辑建议翻译此书时，译者初步浏览原稿后便欣然接受，并急切地希望能将此工具的图书介绍给国内的广大编程开发人员。

本书在介绍 Intel Parallel Studio XE 的过程中是按照串行代码的并行化改造的四步骤为主线展开的。按照热点代码的定位、添加并行语法构造、错误检查（调试）、性能调优的流程，以样例代码为素材，以 Parallel Studio XE 为工具，实施这四个步骤完成串行应用的并行改造。因此本书首先解决了前面提到的第一个问题——帮助读者建立起对串行应用的并行改造流程的完整认识，其次才是一个开发工具的使用说明（由于此书并不是 Parallel Studio XE 的使用手册，因此关于 Parallel Studio XE 更详细的内容需要参考其他相关手册）。对刚接触并行开发的编程人员在**“形成完整的并行改造流程的认识”**和**“熟悉 Parallel Studio XE 工具使用”**的过程中都有很大帮助。这也是此书相对于现有的关于 Parallel Studio XE 的其他中文书籍的明显优势及特色。

译者认为读者可以结合陈国良院士的“并行计算系列丛书”、各种并行编程模型（OpenMP、MPI、Cilk Plus 等）教材以及编程开发的工具类书籍（例如本书）来掌握并行程序开发的知识和技能。

此书由我和两位研究生共同完成翻译，其中孔畅主要完成第 6～10 章初稿和早期校对，刘成健主要完成第 11～16 章初稿和早期校对。在此需要特别感谢周远远同学在全书校对过程中付出的艰辛劳动。刘国强和詹家辉同学也参与了校对工作。

由于限于译者水平，虽然我们尽力避免翻译错误并经过多次校订，可能难免仍有错误遗留，欢迎读者指正。

希望各位读者喜欢，更期望读者有所收获。

罗秋明

深圳大学

作 者 简 介

Stephen Blair-Chappell 在过去 15 年里一直在 Intel 软件和服务团队（SSG）工作。在 Intel 的这些时间里，Stephen 是编译器小组的开发者，最近则作为技术咨询工程师帮助用户更好地充分利用 Intel 的软件工具。在进入 Intel 工作之前，Stephen 曾经是德国的一个编译器和调试器公司 CAD-UL 在英国办事处的经营主任。在 CAD-UL 的期间 Stephen 主要负责英国的技术支持。其间他所经手的项目包括设计和规划一个图形化的链接器、面向编程者的保护模式下的编程开发和培训、对多个电信、汽车和嵌入式产业的技术支持。

Stephen 早期在马修巴顿技术学院（Matthew Boulton Technical College）作为技术人员学习电子学，然后在伯明翰城市大学（Birmingham City University，BCU）学习应用软件工程，最终在那里任教。在工作之余，Stephen 还经常去当地伯明翰布尔林圣马丁教堂事奉，演奏手风琴、布道以及偶尔主领礼拜。

Andrew Stokes 是英国伯明翰城市大学（Birmingham City University，BCU）软件与电子专业的退休教员。在教书之前，Andrew 是研究与商业领域的软件开发者。最早在 20 世纪 80 年代的剑桥大学工程实验室开始软件开发，从事扫描电子显微镜的软件工作。这些软件开发工作延伸到商业领域，在商业领域里他从事有限元分析软件包的图形支持程序的工作。

在 BCU 的期间，Andrew 开发了许多仿真工具软件，包括人工神经网络仿真、CPU 仿真、处理器设计、代码开发工具以及 PROLOG 专家系统。Andrew 在退休之后继续保持对软件的兴趣，把游戏编程当成健身活动，例如 3D 象棋，其中并行编程是最重要的。在工作之余，Andrew 是一个热情的园丁，尤其喜欢典型英国花园的艳丽色彩。

技术编辑简介

Kittur Ganesh 是 Intel 的高级技术咨询工程师，拥有超过 7 年的针对 Intel 架构的各种软件产品的咨询、技术支持和培训经历。在此之前的 6 年多的时间里，Kittur 设计和开发主要用于 Intel 芯片的压力测试数据的软件。在进入 Intel 的这 13 年之前，Kittur 从事开发 EDA 产业的商业软件超过 10 年。Kittur 拥有计算机科学和工业工程两个硕士学位和一个机械工程学士学位。

Pablo Halpern 是 Intel 公司的一位高级软件工程师，在并行运行库团队中工作。他是 C++标准委员会的成员并且协助制定该标准最近的 C++ 11 版本。Pablo 是热销书“The C++ Standard Library from Scratch”的作者以及论文“Reducers and Other Cilk++ Hyperobjects”的作者之一，后者被提名为 2009 年 ACM SPAA 的最佳论文。他拥有超过 30 年的软件行业资历，精通 C++、语言与编译器设计、大规模开发和测试、网络管理协议。近期，他完成了 C++编程的基础及高级课程的开发和教学。他现在和他的妻子和两个小孩住在新罕布尔什。

致　　谢

首先，我们要感谢我们的家人和朋友在我们完成此书时给予的支持和耐心。

我们要感谢 Wrox 的每一个人给了我们机会。谢谢 Paul Reese 编辑，是他首先邀请我们写这本书的；对于 John Sleeva 项目编辑，如果没有他的帮助和指导将无法完成这项工作；对 Kim Cofer 排版编辑、Kittur Ganesh 和 Pablo Halpern 技术编辑，他们的及时的建议帮助极大；Gastón Hillar 给予了进一步的技术复查；对于其他图形分部的所有人，感谢你们的图表。

我们要特别感谢其他参与本书工作的人，特别是 Mark Davis 写了第 10 章“Parallel Advisor 设计指导”和 Fred Tedeschi 写了第 11 章“并行程序的调试”。

我们还要感谢那些让我们在写案例分析时使用他们的经验的人们。谢谢 Lars Peters Endresen 和 Håvard Graff 在第 13 章的工作“世界上第一个数独‘Thirty-Niner’”；对 Yann Golanski 博士，感谢他在第 14 章给出的“通往并行编程殿堂的 9 条建议”的内容；对 Hans Pabst，感谢他在第 15 章“CERN 对撞机的并行轨迹拟合”中的帮助。

我们的感激也属于 Intel 的许多同事，他们不厌其烦地阅读各章——特别是 Levent Akyil、Bernth Andersson、Julian Horn、Martyn Corden、MaxymDmytrychenko、Max Domeika、Hubert Haberstock、Markus Metzger、Mark Sabahi 和 Thomas Zipplies。

最后，感谢 James Reinders，是他鼓励我们编写此书，他也很和蔼。

— Stephen Blair-Chappell
Andrew Stokes

序

从真实的例子中学习可以去除理论上的干扰并且加入不那么迷人的现实。真实的经历和样例帮助我们看清什么才是起关键作用的因素。

在此书中，我非常高兴 Stephen 能够分享他在学会如何真正使用工具及开发并行代码的访谈中的一些秘诀。其结果就是本书远比目录所显示的内容要更加有价值。

例如，我知道数据布局对数据并行处理能力有重要影响，但是我更愿意被实例所说服。在数据布局的主题中，例如要用数组结构体（structures of arrays，SOA）替代结构体数组（arrays of structures，AOS），因 Stephen 在第 15 章“CERN 对撞机的并行轨迹拟合”访谈中提出的挑战性的问题“如果你将这个项目重做一遍，你能做出什么不同？”而将这个问题推到了最前端。在回答中，被采访的开发者强调数据模型对并行程序效率的重要性。第 13 章“世界上第一个数独‘Thirty-Niner’”强调“大多数时间都花在了反复地编码以使得不同的运行任务之间的共享数据更少”。

无处不在的并行性影响了现代编程的各个方面。我被 Stephen 的工作所鼓舞，他涉及了各个方面而不仅仅是编程。涵盖发现、调试、调优等问题对理解并行编程的挑战是很关键的。我希望本书将激励每一位读者。

“按并行方式思考问题”

—James Reinders

主管，并行倡导者，Intel

波特兰，俄勒冈

前　言

现在几乎所有市面上销售的计算机都有多个处理器核，但是只有极少数应用软件可以利用这些额外的处理器核。大多数程序员都在努力跟上形势。最近一项在高级编程工程师团队的调查表明三个阻碍采用并行的障碍是：遗留代码的转换问题、缺少教育培训以及缺少合适的编程工具。本书将有助于解决其中一些障碍。

本书的目的是帮助你使用Intel Parallel Studio XE来编写可以利用多核CPU最新特性的程序。借助于本书，你应当能够编写出快速、安全和并行的代码。除了帮助你编写并行代码之外，一些章节涵盖其他可以用在代码开发中的优化专题——不论是开发并行代码或串行代码都有帮助。全书各章的大多数都包含一些容易上手的练习，用于帮助你应用所讲解的内容。

本书对象

如果你在编写并行代码或者对编写并行代码感兴趣，你就是本书的读者。目标读者包括：

- 正在给代码增加并行性的C和C++开发人员，所需要的技能是“一般”或“优秀”，对C语言的知识是必要条件。
- 寻求代码并行化的实践经验的学生或研究人员。
- Intel Parallel Studio XE的用户或买主。

本书内容

本书使用Parallel Studio XE 2011来编写，介绍如何进行性能测评、优化和代码并行化。通过阅读，你应当掌握：

- 通过分析应用程序从而确定实现并行性的最佳位置。
- 用各种语言扩展/标准来实现并行性。
- 检测和修正难以发现的并行错误。
- 并行程序的性能调优。
- 编写更安全的代码。
- 利用编译器开关选项对代码进行优化以便利用最新CPU的扩展特性。
- 执行结构相关的分析以便回答“我的程序充分利用CPU了吗？”这样的问题。

本书结构

本书包含以下部分：

- 第 1 篇　并行简介。
- 第 2 篇　Parallel Studio XE 教程。
- 第 3 篇　案例分析。

书中的每一章（除了前两章以外），提供了易于上手的练习。这些练习是本书的重要部分，当然也可以不做练习而直接把整本书阅读完。

第 6～9 章需要按顺序学习，展示了如何通过经实践检验的、四步骤方法（分析、实现、错误检查和调优）来给代码增加并行性。书中提供了各种使用 Cilk Plus、OpenMP 和线程构造块 TBB 的并行例子。

案例分析是基于大型项目的，展示了如何使用 Parallel Studio XE 来将它们并行化。

阅读本书所需的其他条件

你在阅读时需要：

- Intel Parallel Studio XE。你可以从 Intel 软件评估中心（http://software.intel.com/en-us/articles/intel-software-evaluation-center/）下载评估版。
- 如果你使用 Windows：
  Visual Studio（不是 Express 版）2005、2008 或 2010
  Windows XP、Windows 2008 或 Windows 7
- 如果你使用 Linux：
  已安装的 GNU GCC 编译器开发工具
  Debian* 6.0、Redhat Enterprise Linux*4 （推荐使用）/5/6、SUSE Linux Enterprise Server* 10/11 SP1 或 Ubuntu* 10.04
- 一台 PC，支持 Intel 流式 SIMD 扩展（SSE）指令的 IA-32 或 64 架构的处理器（Intel Pentium 4 或更新的处理器），或者非 Intel 处理器的兼容机。如果使用非 Intel 处理器，将无法完成第 12 章“基于事件的 VTune Amplifier XE 分析”里的练习。

源代码

在学习本书的例子时，你可能会自己输入代码，或者使用本书附带的源代码文件。本书使用的所有源代码都可以从 www.wrox.com 下载。在该网站上，找到本书的标题（用搜索框或其中一个标题列表）然后单击本书的详细网页上的 Download Code 链接就可以获得本书所有源代码。

一旦下载好了代码，用你所喜欢的解压工具解压缩。你也可以在 Wrox 的下载主页面 www.wrox.com/dynamic/books/download.aspx 去看看本书以及 Wrox 其他书籍的可下载代码。

勘误表

我们尽一切努力让本书没有文字上或代码中的错误。然而，没有人是十全十美的，难免会出错。如果你找到我们书中的错误，例如拼写错误或有误的代码，我们将非常欢迎你

的反馈。通过给出勘误表，你可能减少另一位读者几个小时的烦恼，同时你也帮助我们提供更加高质量的信息。

为了查找本书的勘误表，到 www.wrox.com 网站并用搜索框或标题列表查找本书标题。然后，在标题的详细内容页面，单击 Book Errata 链接。在这个页面上，你可以看到 Wrox 编辑所发布的本书收集到的所有勘误。有一个书籍的完整列表，包含每本书的勘误链接

www.wrox.com/misc-pages/booklist.shtml

如果你不能在 BookErrate 页面上找到你所发现的错误，访问

www.wrox.com/contact/techsupport.shtml

在网页上填写你所发现的错误并发送给我们。我们将检查你的信息，如果合适的话，将此消息公布到本书的勘误网页上，并且在本书的下一版本中修正。

关于作者和同行讨论，请加入 p2p.wrox.com 的 P2P 论坛。该论坛基于 Web 系统，用于张贴关于 Wrox 书籍、技术的消息以及和其他读者或专业用户交流。该论坛具有订阅功能，一旦你所选择的主题有新的帖子时可以将它发送到你的电子邮箱。Wrox 作者、编辑、其他业内专家以及像你这样的读者都会出现在该论坛中。

在 http://p2p.wrox.com，你将发现许多其他对你有所帮助的论坛，不仅仅限于阅读此书，甚至在开发你自己的应用时也能有所帮助。如果想加入论坛，请按如下步骤操作：

（1）访问 p2p.wrox.com 页面，单击 Register 链接。

（2）阅读使用条款然后单击 Agree。

（3）完成加入论坛所需的信息填写，以及其他任何你愿意提供的信息，然后单击 Submit。

（4）你将收到一封关于如何验证你的账号以及完成加入论坛的过程的邮件。

你可以在不加入论坛的情况下阅读各种论坛信息，但是如果想发表你自己的消息就必须加入论坛。

一旦加入论坛，你就可以发布消息并可以回复其他人的消息。你可以在任何时间阅读 Web 上的消息。如果希望收到特定论坛上的新消息，请在论坛列表中论坛名字上单击 Subscribe to this Forum 图标。

关于更多如何使用 Wrox P2P 的信息，请阅读 P2P FAQ 关于论坛是如何工作的问题答案，以及其他关于 P2Ph 和 Wrox 书籍的常见问题。如果想阅读 FAQ，单击任何一个 P2P 页面上的 FAQ 链接。

目　录

第1篇　并行简介

第 2 篇 Parallel Studio XE 教程

第3篇 案例分析