<<深入理解C++11>>

图书基本信息

书名：<<深入理解C++11>>

13位ISBN编号：9787111426608

10位ISBN编号：7111426606

出版时间：2013-6

出版时间：Michael Wong、 IBM XL编译器中国开发团队 机械工业出版社 (2013-06出版)

前言

前言为什么要写这本书相比其他语言的频繁更新，C++语言标准已经有十多年没有真正更新过了。
而上一次标准制定，正是面向对象概念开始盛行的时候。
较之基于过程的编程语言，基于面向对象、泛型编程等概念的C++无疑是非常先进的，而C++98标准的制定以及各种符合标准的编译器的出现，又在客观上推动了编程方法的革命。
因此在接下来的很多年中，似乎人人都在学习并使用C++。
商业公司在邀请C++专家为程序员讲课，学校里老师在为学生绘声绘色地讲解面向对象编程，C++的书籍市场也是百花齐放，论坛、BBS的C++板块则充斥了大量各种关于C++的讨论。
随之而来的，招聘启事写着“要求熟悉C++编程”，派生与继承成为了面试官审视毕业生基础知识的重点。
凡此种种，不一而足。
于是C++语言“病毒性”地蔓延到各种编程环境，成为了使用最为广泛的编程语言之一。
十来年的时光转瞬飞逝，各种编程语言也在快马加鞭地向前发展。
如今流行的编程语言几乎无一不支持面向对象的概念。
即使是古老的语言，也通过了制定新标准，开始支持面向对象编程。
随着Web开发、移动开发逐渐盛行，一些新流行起来的编程语言，由于在应用的快速开发、调试、部署上有着独特的优势，逐渐成为了这些新领域中的主流。
不过这并不意味着C++正在失去其阵地。
身为C的“后裔”，C++继承了C能够进行底层操作的特性，因此，使用C/C++编写的程序往往具有更佳的运行时性能。
在构建包括操作系统的各种软件层，以及构建一些对性能要求较高的应用程序时，C/C++往往是最佳选择。
更一般地讲，即使是由其他语言编写的程序，往往也离不开由C/C++编写的编译器、运行库、操作系统，或者虚拟机等提供支持。
因此，C++已然成为了编程技术中的中流砥柱。
如果用个比喻来形容C++，那么可以说这十来年C++正是由“锋芒毕露”的青年时期走向“成熟稳重”的中年时期。
不过十来年对于编程语言来说也是个很长的时间，长时间的沉寂甚至会让有的人认为，C++就是这样一种语言：特性稳定，性能出色，易于学习而难于精通。
长时间使用C++的程序员也都熟悉了C++毛孔里每一个特性，甚至是现实上的一些细微的区别，比如各种编译器对C++扩展的区别，也都熟稔于心。
于是这个时候，C++11标准的横空出世，以及C++之父Bjarne Stroustrup的一句“看起来像一门新语言”的说法，无疑让很多C++程序员有些诚惶诚恐：C++11是否又带来了编程思维的革命？
C++11是否保持了对C++98及C的兼容？
旧有的C++程序到了C++11是否需要被推倒重来？
事实上这些担心都是多余的。
相比于C++98带来的面向对象的革命性，C++11带来的却并非“翻天覆地”式的改变。
很多时候，程序员保持着“C++98式”的观点来看待C++11代码也同样是合理的。
因为在编程思想上，C++11依然遵从了一贯的面向对象的思想，并深入加强了泛型编程的支持。
从我们的观察来看，C++11更多的是对步入“成熟稳重”的中年时期的C++的一种改造。
比如，像auto类型推导这样的新特性，展现出的是语言的亲和力；而右值引用、移动语义的特性，则着重于改变一些使用C++程序库时容易发生的性能不佳的状况。
当然，C++11中也有局部的创新，比如lambda函数的引入，以及原子类型的设计等，都体现了语言与时俱进的活力。
语言的诸多方面都在C++11中再次被锤炼，从而变得更加合理、更加条理清晰、更加易用。
C++11对C++语言改进的每一点，都呈现出了经过长时间技术沉淀的编程语言的特色与风采。

所以从这个角度上看，学习C++11与C++98在思想上是一脉相承的，程序员可以用较小的代价对C++的知识进行更新换代。
而在现实中，只要修改少量已有代码（甚至不修改），就可以使用C++11编译器对旧有代码进行升级编译而获得新标准带来的好处，这也非常具有实用性。
因此，从很多方面来看，C++程序员都应该乐于升级换代已有的知识，而学习及使用C++11也正是大势所趋。
在本书开始编写的时候，C++11标准刚刚发布一年，而本书出版的时候，C++11也只不过才诞生了两年。
这一两年，各个编译器厂商或者组织都将支持C++11新特性作为了一项重要工作。
不过由于C++11的语言特性非常的多，因此本书在接近完成时，依然没有一款编译器支持C++11所有的新特性。
但从从业者的角度看，C++11迟早会普及，也迟早会成为C++程序员的首选，因此即使现阶段编译器对C++新特性的支持还不充分，但还是有必要在这个时机推出一本全面介绍C++11新特性的中文图书。
希望通过这样的图书，使得更多的中国程序员能够最快地了解C++11新语言标准的方方面面，并且使用最新的C++11编译器来从各方面提升自己编写的C++程序。
读者对象本书针对的对象是已经学习过C++，并想进一步学习、了解C++11的程序员。
这里我们假定读者已经具备了基本的C++编程知识，并掌握了一定的C++编程技巧（对于C++的初学者来说，本书阅读起来会有一定的难度）。
通过本书，读者可以全面而详细地了解C++11对C++进行的改造。
无论是试图进行更加精细的面向对象程序编写，或是更加容易地进行泛型编程，或是更加轻松地改造使用程序库等，读者都会发现C++11提供了更好的支持。
本书作者和书籍支持本书的作者都是编译器行业的从业者，主要来自于IBM XL编译器中国开发团队。IBM XL编译器中国开发团队创立于2010年，拥有编译器前端、后端、性能分析、测试等各方面的人员，工作职责涵盖了IBM XL C/C++及IBM XL Fortran编译器的开发、测试、发布等与编译器产品相关的方方面面。
虽然团队成立时间不长，成员却都拥有比较丰富的编译器开发经验，对C++11的新特性也有较好的理解。
此外，IBM北美编译器团队成员Michael（他是C++标准委员会的成员）也参加了本书的编写工作。
在书籍的编写上，Michael为本书拟定了提纲、确定了章节主题，并直接编写了本书的首章。
其余作者则分别对C++11各种新特性进行了详细研究讨论，并完成了书稿其余各章的撰写工作。
在书稿完成后，除了请Michael为本书的部分章节进行了审阅并提出修改意见外，我们又邀请了IBM中国信息开发部及IBM北京编译器团队的一些成员对本书进行了详细的审阅。
虽然在书籍的策划、编写、审阅上我们群策群力，尽了最大的努力，以保证书稿质量，不过由于C++11标准发布时间不长，理解上的偏差在所难免，因此本书也可能在特性描述中存在一些不尽如人意或者错误的地方，希望读者、同行等一一为我们指出纠正。
我们也会通过博客、微博发布与本书相关的所有信息，并与本书读者共同讨论、进步。
如何阅读本书读者在书籍阅读中可能会发现，本书的一些章节对C++基础知识要求较高，而某些特性很可能很难应用于自己的编程实践。
这样的情况应该并不少见，但这并不是这门语言缺乏亲和力，或是读者缺失了背景知识，这诚然是由于C++的高成熟度导致的。
在C++11中，不少新特性都会局限于一些应用场景，比如说库的编写，而编写库却通常不是每个程序员必须的任务。
为了避免这样的状况，本书第1章对C++11的语言新特性进行了分类，因此读者可以选择按需阅读，对不想了解的部分予以略过。
一些本书的使用约定，读者也可以在第1章中找到。
致谢在这里我们要对IBM中国信息开发部的陈晶（作者之一）、卢昉、付琳，以及IBM北京编译器团

队的冯威、许小羽、王颖对本书书稿详尽细致的审阅表示感谢，同时也对他们专业的工作素养表示由衷的钦佩。
此外，我们也要感谢IBM XL编译器中国开发团队的舒蓓、张嗣元两位经理在本书编写过程中给予的大力支持。
而IBM图书社区的刘慎峰及华章图书的杨福川编辑的辛勤工作则保证了本书的顺利出版，在这里我们也要对他们以及负责初审工作的孙海亮编辑说声谢谢。
此外，我们还要感谢各位作者的家人在书籍编写过程中给予作者的体谅与支持。
最后要感谢的是本书的读者，感谢你们对本书的支持，希望通过这本书，我们能够一起进入C++编程的新时代。
IBM XL编译器中国开发团队

<<深入理解C++11>>

内容概要

《深入理解C++11:C++11新特性解析与应用》内容简介：国内首本全面深入解读C++11新标准的专著，由C++标准委员会代表和IBM XL编译器中国开发团队共同撰写。
不仅详细阐述了C++11标准的设计原则，而且系统地讲解了C++11新标准中的所有新语言特性、新标准库特性、对原有特性的改进，以及如何应用所有这些新特性。

《深入理解C++11:C++11新特性解析与应用》一共8章：第1章从设计思维和应用范畴两个维度对C++11新标准中的所有特性进行了分类，呈现了C++11新特性的原貌；第2章讲解了在保证与C语言和旧版C++标准充分兼容的原则下增加的一些新特性；第3章讲解了具有广泛可用性、能与其他已有的或者新增的特性结合起来使用的、具有普适性的一些新特性；第4章讲解了C++11新标准对原有一些语言特性的改进，这些特性不仅能让C++变得更强大，还能提升程序员编写代码的效率；第5章讲解了C++11在安全方面所做的改进，主要涵盖枚举类型安全和指针安全两个方面的内容；第6章讲解了为了进一步提升和挖掘C++程序性能和让C++能更好地适应各种新硬件的发展而设计的新特性，如多核、多线程、并行编程方面的新特性；第7章讲解了一些颠覆C++一贯设计思想的新特性，如lambda表达式等；第8章讲解了C++11为了解决C++编程中各种典型实际问题而做出的有效改进，如对Unicode的深入支持等。
附录中则介绍了C++11标准与其他相关标准的兼容性和区别、C++11中弃用的特性、编译器对C++11的支持情况，以及学习C++11的相关资源。

作者简介

作者：（加拿大）Michael Wong IBM XL编译器中国开发团队Michael Wong，C++11标准委员会（WG21）委员，WG21加拿大代表团团长及IBM公司代表（投票人），WG21 SG5(transactional Memory study group)学习小组组长。
除此之外，Michael还是OpenMP委员会的CEO，SCC(Standards Council of Canada)的副主席。
Michael是多个C++11/OpenMP/STM新特性的合作编写者，活跃于各种技术会议，而且为Programming Committee of Boost以及IWOMP等国际会议服务。
Michael也是IBM公司的高级技术专家，领导IBM XL C++编译器、XL C编译器的开发工作，当前致力于IBM编译器的C++11新特性部署。
Michael在并行编程、C++基准测试、对象模型、泛型编程、模板元编程等多个技术领域均有涉猎，并且在对象模型以及事务内存等方面取得过多个专利。
早年Michael于多伦多大学获取得了天体物理学的学士学位，并在滑铁卢大学获得了硕士学位。
在加入IBM之前，Michael曾先后供职于大卫?邓禄普天文台（David Dunlap Observatory），CDC公司（Control Data Corporation），BMO Nesbitt Burns公司等。
IBM XL编译器中国开发团队，IBM拥有悠久的编译器开发历史（始于上世纪80年代），在全球有将近400名高素质工程师组成的研发团队，其中包括许多世界知名的研究学者和技术专家。
IBM一直以来都是编程语言的制定者和倡导者之一，并将长期在编译领域进行研发和投资。
IBM编译器中国开发团队于2010年在上海成立，现拥有编译器前端开发人员（C/C++）、后端开发人员、测试人员，以及性能分析人员共16人。
团队与IBM北美编译器团队紧密合作，共同开发、测试和发布基于POWER系统的AIX及Linux平台下的XL C/C++和XL Fortran系列产品，并对其提供技术支持。
虽然团队成立时间不长，但已于2012年成功发布最新版本的XL C/C++ for Linux V12.1 & XL Fortran for Linux V14.1，并获得7项发明专利。
团队成员拥有较丰富的编译器开发经验，对编译技术、编程语言、性能优化和并行计算等各领域都有一定的研究，也对C++11标准的各种新特性有较早的研究和理解，并正在实际地参与C++11新特性的开发工作。

<<深入理解C++11>>

书籍目录

<<深入理解C++11>>

编辑推荐

《深入理解C++11:C++11新特性解析与应用》编辑推荐：C++标准委员会成员和IBM XL编译器中国开发团队共同撰写，权威性毋庸置疑。
系统、深入、详尽地讲解了C++11新标准中的新语言特性、新标准库特性、对原有特性的改进，以及所有这些新特性的应用。

<<深入理解C++11>>

版权说明