<<Boost程序库完全开发指南>>

图书基本信息

书名：<<Boost程序库完全开发指南>>

13位ISBN编号：9787121190896

10位ISBN编号：7121190893

出版时间：2013-1

出版时间：电子工业出版社

作者：罗剑锋

前言

屈指算来，接触C＋＋语言至今已经有十余个年头了。
回首往事，不禁感慨良多。
缘起1996年我上大学最开始学的是Pascal，不得不说，Pascal严谨的程序风格确实很适合作为一门教学语言，然而用于实际开发就不那么合适了（直到出现Delphi）。
由于当时学校并未开设C语言课程，因此在Pascal课程结束后我就买书自学C／C＋＋语言，并在次年报名国家计算机水平考试，靠着一点点编程和考试的“天分”获得了高级程序员资质（当年很热衷考级考证，后来就“淡定”多了）。
虽然有了资格证，但我仍然算是个C＋＋的初学者，对于C＋＋的认识还处于C的面向过程和简单的基于对象层次上。
新千年伊始我考入了北京理工大学就读研究生，因为跟导师做项目开始接触STL与C＋＋标准库，大概是2005年从1．33版结识了Boost，这才真正领略了C＋＋的精髓。
那段时期Java和C＃正在国内大行其道，C＋＋则势单力薄，有关STL和C＋＋标准的技术书籍寥寥无几，而讲解Boost的书更是为零，故对Boost的学习基本只能靠自己的摸索与实践。
好在Boost自带的文档相当丰富（虽然看全英文的资料十分辛苦），而且源码也写得比较清晰规范，在熟悉了STL的基础上学习Boost倒也并不算太难。
但Boost的一个最大的特点就是“庞大”，功能组件众多，要想把它全部装进脑子里融会贯通基本上是不可能的，使用时需要经常查阅英文文档，相当的麻烦。
因此，在学习的过程中，我逐渐产生了编写学习笔记的想法。
一开始只是一个简单的纯文本文件，记录了一些使用经验的片断，随着积累的不断增加，纯文本形式已经不能够满足知识整理的需求了，于是我又把这些文字迁移到了Word文档里，把使用经验分类编目，进行较系统的归纳梳理。
慢慢地，这份学习笔记居然有了上万字的规模，成为了一份很好的Boost备忘参考，在日常的开发工作中给了我很大的帮助——就像《设计模式》一书中所说的那样，捕获了很多使用Boost解决问题的实践经验，避免了重复发现。
不过，这份资料一直是仅限于我个人使用，属于“自娱自乐”的作品，从未示人。
时间一晃到了2010年1月份的某天夜里，不知道是什么原因我忽然失眠了，躺在床上翻来覆去怎么也睡不着。
突然，一个念头闯入了脑海：把Boost开发经验整理出版吧，让更多人能够分享这些知识，正所谓“独乐乐，与众乐乐，孰乐？
”这个大胆的想法的出现让我那天的失眠又延长了几个小时——关于书的各种构想在头脑中“肆虐横行”。
随后的几天里我就把这个想法付诸行动了，虽然以前也写过并发表很多文章，也在网上印刷了几本个人文集，但出版正式的书还是第一次。
在把学习笔记进一步整理完善，编写出较完整的结构和一个样章后，我就开始联系出版社了。
当初并没有多大的信心，毕竟我这个作者名不见经传，也没有什么资历、背景和名气（而且还是个“网盲”，从未跟随潮流开个人博客）。
很幸运，发出的第一个E－mail就是电子工业出版社，而且编辑也在第一时间回复了我，这才给了我以持续写作完成全书的动力。
写作过程中我也进一步加深了对Boost的认识，澄清了许多原来未曾注意到的细节。
原本只打算写20万字左右、三百多页，但写到中途才发现Boost库的博大精深远非当初的理解，也意识到了自己当初学习的肤浅。
历经了近半年近乎不眠不休的努力，最终呈现给读者的是这本厚达500多页的图书，文字量是最初学习笔记的数十倍，内容也翔实丰满了很多——达成这个结果，我个人可以说是问心无愧了。
C＋＋与BoostC＋＋较Java和C＃等语言的一个最大不同在于它并非是由某个公司或个人把持的，它的真正发展动力来自于广大程序员。

Boost就是这样的一个典范，它成功地填补了从C＋＋98到C＋＋0X这“失落的十年”间的空白，在竞争对手Java和C＃不断更新版本新增特性的时候以库的形式极大地增强了C＋＋的能力，使C＋＋不至于因为标准规范的滞后而落后于时代，而且Boost还深层次挖掘了C＋＋的潜力，开创了泛型编程、模板元编程、函数式编程等崭新的境界。
就个人来说，我比较喜欢的Boost版本有两个，分别是1．35和1．39。
1．35版增加了asio、bimap、circular__buffer等许多重要组件，而1．39版则增加了signals2库，这两个版本都在我的工作用机上停留了相当长的时间。
落笔之时，Boost已经更新到了1．43版，成长为一个相当完善、全面、强大的C＋＋程序库。
可以毫不夸张地说，现在的C＋＋程序员，如果不熟悉Boost，那么至少丧失了一半使用C＋＋的好处，同时会多耗费数倍的开发精力和时间。
随着C＋＋0X标准的即将来临，Boost程序库的发展也出现了加速的趋势，由原来间隔数月不定期更新版本，改为定期（每3个月左右）发布新版本，而且每个新版本都会包含大量极有价值的更新内容。
因此，希望读者在阅读本书时及时访问Boost的官网，以便获取最新的版本。
感谢读者选择本书，再说一句真心的“套话”（笑）：限于作者水平有限，书中错漏在所难免，敬请读者原谅、指正。
致谢首先我要感谢整个C＋＋群体，特别是：C＋＋语言的发明者Bjarne Stroustrup博士——他给我们带来了美妙的C＋＋；然后是Alexander Stepanov和C＋＋标准委员会——他们把STL引入了C＋＋，开创了C＋＋的现代编程风格；以及Beman G． Dawes、Boost程序库的所有作者和Boost社区——他们为我们奉献了如此高水准的程序库。
其次我要感谢电子工业出版社博文视点公司，他们给了我这个把自己的开发经验出版成书的机会，在把潦草的个人学习笔记变成正式图书的过程中他们付出了艰辛的努力。
还要感谢陈硕先生，他审阅了本书的部分手稿，提出了很多有价值的参考意见，并慨然为本书撰写序言。
接下来我要感谢我的家人：感谢我的父母和弟弟，他们永远是我生命中最重要的人；感谢我的妻子，她自始至终都支持我的写作，并担负了大部分照顾孩子的家务（虽然偶有怨言）；还要对已满一岁半的女儿说声抱歉，为了写作本书，我已经牺牲了很多陪她玩耍的时间。
我还要感谢黄美华、冯薇、戚天龙、罗玉震、颜静、陈刚、张秋香、缪泽波等同事，长期的共事令我们建立了深厚的友谊。
对后两位同事致以特别的感谢，他们对完成本书提供了大力的支持和帮助。
最后，感谢多年以来的好友岳大海、时吉斌、王峰，感谢我的中学老师邓英、杜爱芹、练鑫云、陈静，感谢我的研究生导师贾云得，以及所有在我成长过程中曾经给予我关心和帮助的朋友们！
罗剑锋北京王府井

<<Boost程序库完全开发指南>>

内容概要

《Boost程序库完全开发指南:深入C++"准"标准库(第2版)》基于2012年8月发布的Boostl.51版，介绍了其中的所有117个库，并且结合c++11标准详细、深入地讲解了其中数十个库，同时实现了若干颇具实用价值的工具类和函数，可帮助读者迅速地理解、掌握Boost的用法及其在实际开发工作中的应用。
《Boost程序库完全开发指南:深入C++"准"标准库(第2版)》内容丰富、结构严谨、详略得当、讲解透彻，带领读者领略了C++的最新前沿技术，相信会是每位c++程序员的必备工具书。

作者简介

罗剑锋，1996年就读于东北财经大学，1997年开始接触C／C++，1998年参加计算机软件专业技术资格和水平考试，获高级程序员资质，2003年毕业于北京理工大学，获计算机专业硕士学位，目前任项目经理，主要研究方向为C／C++、设计模式、密码学、数据库、嵌入式系统开发，业余爱好是阅读、欣赏音乐和旅游。

<<Boost程序库完全开发指南>>

书籍目录

章节摘录

版权页： 时间的处理很复杂，因此在使用date_time库之前，我们需要明确一些基本概念。
如果把时间想象成一个向前和向后都无限延伸的实数轴，那么时间点就是数轴上的一个点，时间段就是两个时间点之间确定的一个区间，时长（时间长度）则是一个有正负号的标量，它是两个时间点之差，不属于数轴。
时间点、时间段和时长三者之间可以进行运算，例如时间点+时长＝时间点，时长+时长＝时长，时间段∩时间段=时间段、时间点∈时间段等等，但有的运算也是无意义的，如时间点+时间点、时长+时间段等等。
这些运算都基于生活常识，很容易理解，但在编写时间处理程序时必须注意。
date_time库支持无限时间和无效时间（NADT，Not Available Date Time）这样特殊的时间概念，类似于数学中极限的含义。
时间点和时长都有无限的值，它们的运算规则比较特别，例如+∞时间点+时长=+∞时间点，时间点+∞时长=+∞时间点。
如果正无限值与负无限值一起运算将有可能是无效时间，如+时长－∞NADT。
date_time库中用枚举special_values定义了这些特殊的时间概念，它位于名字空间boost：：date_time，并被using语句引入其他子名字空间。
pos_infin ： 表示正无限； neg_infin ： 表示负无限； not_a_date_time ： 无效时间； min_date_time ： 可表示的最小日期或时间； max_date_time ： 可表示的最大日期或时间。

<<Boost程序库完全开发指南>>

编辑推荐

《Boost程序库完全开发指南:深入C++"准"标准库(第2版)》编辑推荐：好评连连，经典畅销书全面升级，可以一次性窥视C＋＋的许多新特性。

版权说明