

原

纵横中“初品”Boost程序库探秘

一、话说“初品”Boost程序库探秘

C++最新标准C++11的公布，对一直游走在C++98的C++开发者（包括我自己）来说是福音亦是挑战。C++11引入的新特性让C++更具魅力变得更加友好易用，这是福音；但随之而来的语言复杂度的增加，无疑是一种挑战。面对魅力的诱惑和全新的挑战，我们如何从沉迷已久的C++98平滑地过度到C++11呢，罗剑锋老师教了我们一招“杀手锏”：探秘Boost程序库。Boost程序库作为C++的“准”标准库，在诠释和扩充着C++98在各个领域的能力的同时，不断地融入C++11的元素，积极地参与了C++11的实现，因此，从这个层面来讲，Boost是一个从C++98到C++11的最佳“引路人”，可以让我们以很低的成本无缝地过渡到C++11。

鉴于Boost的博大精深和罗老师的深厚内功，在仅仅试读几章的情况，笔者不敢在此就《Boost程序库探秘——深度解析C++“准”标准库》（以下简称“探秘”）的内容大放阙词，胡乱评价。故此，笔者只敢用“初品”二字来作为该文的标题，意在阐述自己试读《探秘》的心得以及个人认为本书的亮点所在，并在下一节分享自己的阅读方式和过程。

亮点一：各个章节之间的合理布局 and 内容的递进关系

在阅读第三章迭代器的内容时，给我最大的感受是章节之间的合理布局。前文中提到过Boost是C++98到C++11过渡的“引路人”，本章在阐述中亦是遵循着该主线，先详细地阐述了C++98中STL标准迭代器的定义、分类以及用法，再深入分析Boost的新式迭代器定义和分类，最后重点研究iterators库里提供的迭代器工具。这条主线的贯穿让我们清晰地掌握了STL中标准迭代器和Boost中新式迭代器的区别和使用的不同之处，实现了用已知的知识面来理解新的知识点，从而达到平滑过渡的效果。

亮点二：“理解—使用—扩展”的模式

侯捷大师在《C++标准程序库—自修教程与参考手册》中把学习STL分为三个阶段：第一境界：熟用STL；第二境界：了解泛型技术的内涵与STL的学理乃至实作；第三境界：扩充STL。同样，在《探秘》一书中亦采用了与之相似的模式“理解—使用—扩展”。本书在讲述迭代器和函数对象时，首先是“类摘要”，即理解函数的定义以及各个参数的说明，其次是“用法”，也就是模式中的“使用”，通过简单的示例来让读者掌握Boost库中迭代器和仿函数的使用，最后是“扩展”，即在熟练使用的基础上自己进行灵活运用到实际项目中，并根据实际情况进行相应的扩充。

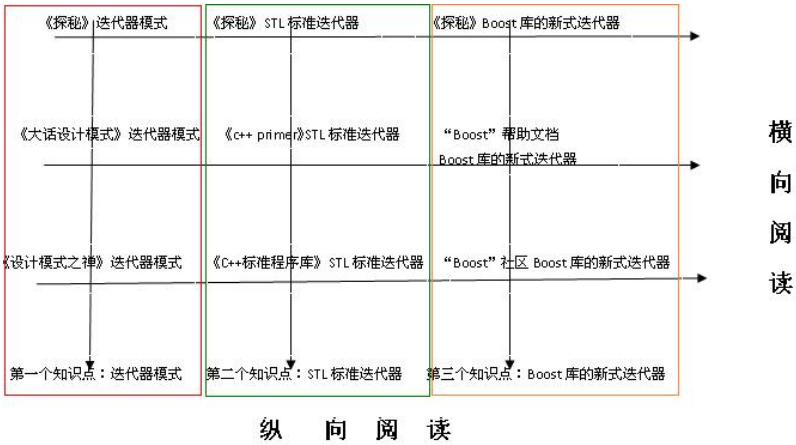
亮点三：C++开发中知识点的串联和推荐书目的快速索引

个人比较喜欢的第三点就是知识点的串联，在迭代器一章中涵盖的重要知识点就设计模式、STL标准迭代器以及Boost库的新式迭代器，同时推荐书目的索引也很给力，这也许是由于我的个人阅读习惯的影响吧。

二、再谈“纵横”

在“初品”完Boost程序库探秘之后，我们再来谈谈何为“纵横”。前文我有说过这节主要是分享阅读过程和方式，的确，我要讲的就是结合阅读《探秘》的试读过程来阐述“横向”和“纵向”的阅读方式。

首先我来解释一下我所认为的“横向”和“纵向”的阅读方式。顾名思义，“横向”涉及的范围比较广，而“纵向”涉及的范围则比较深。“横向”的阅读方式就是按照书的目录的物理顺序一章一节地读完一本又一本，而“纵向”的阅读方式就是按照每本书的相关章节深入挖掘完每一个知识点之后再阅读下一个知识点。比如说，在阅读《探秘》中迭代器一章时，如果按照物理章节顺序“设计模式->STL迭代器->Boost库的新式迭代器”这样阅读就是“横向”阅读，如果按照如下顺序阅读就是“纵向”阅读：《探秘》中“迭代器模式”->《大话设计模式》中“迭代器模式”章节->《设计模式之禅》中“迭代器模式”章节->...->。我个人阅读的习惯就是这种纵横交错的阅读模式。下面我用图表的形式来展现我试读《探秘》的过程。



后记：

由于笔者文笔欠佳，文章中观点如有误之处还请各位看官拍砖指正（不要拍人哈），小生在此感激不尽，^_^。ou 嘞，时间已晚，各位看官欲知后事如何，请细品《Boost程序库探秘——深度解析C++“准”标准库》，晚安！

C++经典书目索引及资源下载：<http://blog.csdn.net/jerryjbiao/article/details/7358796>

阅读更多

登录后自动展开