<<More Effective C++ （>>

图书基本信息

书名：<<More Effective C++（中文版）>>

13位ISBN编号：9787121125706

10位ISBN编号：7121125706

出版时间：2011-1

出版时间：电子工业出版社

作者：梅耶

页数：317

译者：侯捷

<<More Effective C++ （>>

前言

C++ 是一门难学易用的语言！

C++ 的难学，不仅在其广博的语法、语法背后的语义、语义背后的深层思维、深层思维背后的对象模型；C++ 的难学，还在于它提供了4种不同（相辅相成）的编程思维模型：procedural-based，object-based，object-oriented，generic paradigm。

世上没有白吃的午餐。

又要有效率，又要有弹性，又要前瞻望远，又要回溯相容，又要能治大国，又要能烹小鲜，学习起来当然就不可能太简单。

在如此庞大复杂的机制下，万千使用者前赴后继的动力是：一旦学成，妙用无穷。

C++ 相关书籍之多，车载斗量，如天上繁星，如过江之鲫。

广博如四库全书者有之（The C++ Programming Language、C++ Primer），深奥如重山复水者有之（The Annotated C++ Reference Manual, Inside the C++ Object Model），细说历史者有之（The Design and Evolution of C++, Ruminations on C++），独沽一味者有之（Polymorphism in C++, Genericity in C++），独树一帜者有之（Design Patterns, Large Scale C++ Software Design, C++ FAQs），程序库大全有之（The C++ Standard Library），另辟蹊径者有之（Generic Programming and the STL），工程经验之累积亦有之（Effective C++, More Effective C++, Exceptional C++）。

这其中，“工程经验之累积”对已具 C++ 相当基础的程序员而言，有着致命的吸引力与立竿见影的帮助。

Scott Meyers 的Effective C++ 和 More Effective C++ 是此类佼佼，Herb Sutter 的 Exceptional C++ 则是后起之秀。

这类书籍的一个共同特色是轻薄短小，并且高密度地纳入作者浸淫于 C++/OOP 领域多年而广泛的经验。

它们不但开扩读者的视野，也为读者提供各种 C++/OOP 常见问题或易犯错误的解决模型。

某些小范围主题诸如“在base classes 中使用 virtual destructor”、“令operator= 传回 *this 的 reference”，可能在百科型 C++ 语言书籍中亦曾概略提过，但此类书籍以深度探索的方式，让我们了解问题背后的成因、最佳的解法，以及其他可能的牵扯。

至于大范围主题，例如smart pointers，reference counting，proxy classes，double dispatching，基本上已属design patterns的层级！

这些都是经验的累积和心血的结晶！

我很高兴将以下两本优秀书籍，规划为一个系列，以郑重的形式呈现给您： 1.Effective C++ 2/e, by Scott Meyers, AW 1998 2.More Effective C++, by Scott Meyers, AW 1996 本书不但与英文版页页对译，保留索引，并加上译注、交叉索引 、读者服务 。

这套书将对于您的程序设计生涯带来重大帮助。

翻译这套书籍的过程中，我感觉来自技术体会上的极大快乐。

我祈盼（并相信）您在阅读此书时拥有同样的心情。

内容概要

较少，页数倒是多了一些，原因是这次选材比“第一集”更高阶，尤其是第5章。
Meyers将此章命名为技术（techniques），并明白告诉你，其中都是一些patterns，例如virtual constructors，smart pointers，reference counting，proxy classes，double dispatching……这一章的每个条款篇幅都达15~30 页之多，实在让人有“山重水复疑无路，柳暗花明又一村”之叹。
虽然出版年代稍嫌久远，但本书并没有第2版，原因是当其出版之时（1996），C++ Standard已经几乎定案，本书即依当时的标准草案而写，其与现今的C++ 标准规范几乎相同。
而且可能变化的几个弹性之处，Meyers也都有所说明与提示。
读者可以登录作者提供的网址，看看上下两集的勘误与讨论（数量之多，令人惊恐。
幸好多是技术讨论或文字斟酌，并没有什么重大误失）。

作者简介

作者：（美国）梅耶（Scott Meyers） 译者：侯捷Scott Meyers，世界顶级的C++软件开发技术权威之一。
他是两本畅销书Effective C++和More Effective C++的作者，以前曾经是C++ Report的专栏作家。
他经常为C/C++ Users Journal和Dr. Dobb's Journal撰稿，也为全球范围内的客户做咨询活动。
他也是Advisory Boards for NumeriX LLC和InfoCruiser公司的成员。
他拥有Brown University的计算机科学博士学位。
侯捷，集计算机技术读物之著、泽、评于一身，足《多态与虚拟》和《STL源码剖析》的作者，《Run！
Pc》杂志、《程序员》杂志的知名专栏作家，也是极其号召力的大学教师与研讨会讲师。
他于1989年获得中国台湾清华大学动机系硕士学位。

<<More Effective C++ （>>

书籍目录

章节摘录

版权页：插图：我们终于抵达了最后一站。
本章内含难以归类的准则。
一开始的两个条款讨论C++软件开发过程如何设计出能够容纳日后变化的系统。
是的，面向对象方法应用于系统构造的一个强大力量就是，它支持日后的变化。
这些条款描述了一些特定步骤，你可以用来强化你的软件工事，抵抗这个拒绝停滞的世界带来的刀戟箭弩。
接下来我将验证如何在同一个程序中结合C和C++。
这个需求导致语言上的额外考虑，不过C++毕竟生存于真实世界之中，有时候我们必须面对这样的问题。
最后，我把“ C++标准规格 ”公开之后的各项语言变化做一番摘要整理。
在此特别涵盖标准程序库中翻天覆地的大变化（亦请参考条款E49）。
如果你未曾密切跟随标准化的脚步，对于这些变化可能会有很大的惊喜。
是的，标准程序库中有许多让人愉悦的东西。
条款32：在未来时态下发展程序 世事永远在变。
身为软件开发人员，我们可能不是知道得很多，但我们确切知道世事永远在变。
我们不一定知道改变的是什么，改变如何到来，改变何时发生，或为什么会发生，但我们真的知道：事情会改变。
好的软件对于变化有良好的适应能力。
好的软件可以容纳新的性质，可以移植到新的平台，可以适应新的需求，可以掌握新的输入。
软件具备如此的弹性、健壮性、可信赖度，并非是天上掉下来的礼物，而是那些“ 即使面对今天的束缚，仍然对明天可能的需求念兹在兹 ”的设计者和实现者共同努力的结果。

媒体关注与评论

《Effective c++》(Scott Meyers第一本书)的荣耀：“ 对于任何渴望在中阶或高阶层面精通c++的人，我慎重推荐《Effective c++》， ”　　——(The C／C++User's Journal)

<<More Effective C++ （>>

编辑推荐

《More Effective C++:35个改善编程与设计的有效方法(中文版)》：传世经典书丛

版权说明