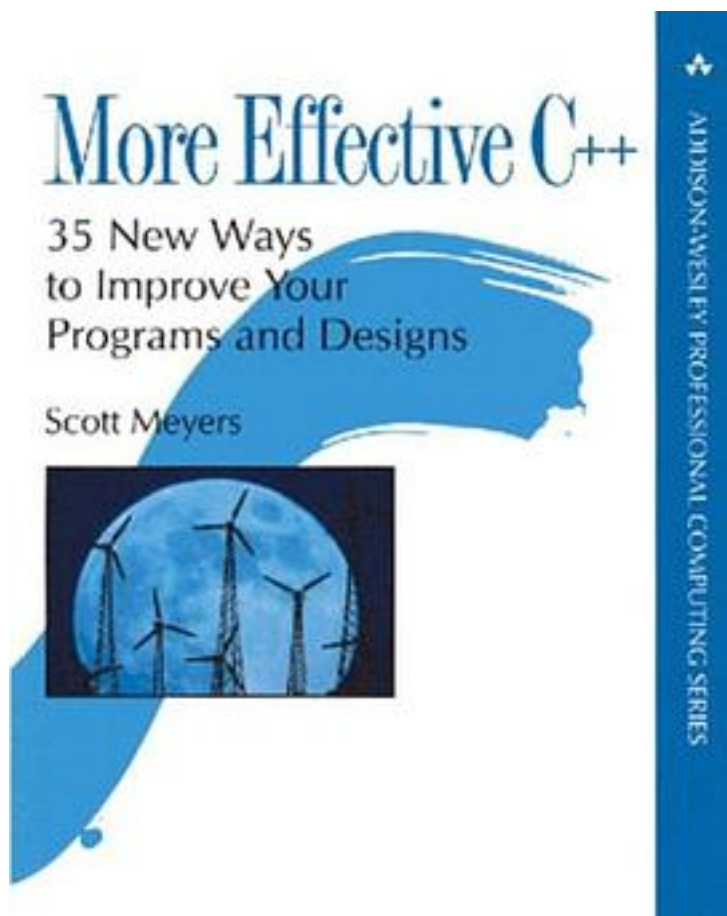


More Effective C++



[More Effective C++ 下载链接1](#)

著者:迈耶斯

出版者:机械工业出版社

出版时间:2006-4

装帧:平装

isbn:9787111188308

一些经过验证的用来改善程序效率的方法，包括检验C++语言特性所带来的时间和空间上的成本。

全面地描述了C++专家所使用的高级技术，包括placement new、virtual

constructors、smart pointers、reference counting、proxy classes和double-dispatching等。

以实例说明异常处理带给C++类和函数的冲击。

介绍新的语言特性，包括bool、mutable、explicit、namespaces、成员模板、标准模板库等。如果你的编译器不支持这些特性，本书还介绍了如何利用它们完成工作。

作者介绍:

Scott Meyers

拥有布朗大学计算机科学博士学位，是世界上最著名的C++软件开发权威之一。他的“Effective C++两卷本”(《Effective C++》和《More Effective C++》)，开辟了技术图书写作的新风格。他曾担任《C++Report》杂志的专栏作家，经常为《C/C++Users Journal》和《Dr.Dobb's Journal》撰稿。

目录: Acknowledgments Introduction Basics Item 1: Distinguish between pointers and references. Item 2: Prefer C++-style casts. Item 3: Never treat arrays polymorphically. Item 4: Avoid gratuitous default constructors. Operators Item 5: Be wary of user-defined conversion functions. Item 6: Distinguish between prefix and postfix forms of increment and decrement operators. Item 7: Never overload &&, ||, or Item 8: Understand the different meanings of new and delete. Exceptions Item 9: Use destructors to prevent resource leaks. Item 10: Prevent resource leaks in constructors. Item 11: Prevent exceptions from leaving destructors. Item 12: Understand how throwing an exception differs from passing a parameter or calling a virtual function. Item 13: Catch exceptions by reference. Item 14: Use exception specifications judiciously. Item 15: Understand the costs of exception handling. Efficiency Item 16: Remember the 80-20 rule. Item 17: Consider using lazy evaluation. Item 18: Amortize the cost of expected computations. Item 19: Understand the origin of temporary objects. Item 20: Facilitate the return value optimization. Item 21: Overload to avoid implicit type conversions. Item 22: Consider using op= instead of stand-alone op. Item 23: Consider alternative libraries. Item 24: Understand the costs of virtual functions, multiple inheritance, virtual base classes, and RTTI. Techniques Item 25: Virtualizing constructors and non-member functions. Item 26: Limiting the number of objects of a class. Item 27: Requiring or prohibiting heap-based objects. Item 28: Smart pointers. Item 29: Reference counting. Item 30: Proxy classes. Item 31: Making functions virtual with respect to more than one object Miscellany Item 32: Program in the future tense. Item 33: Make non-leaf classes abstract. Item 34: Understand how to combine C++ and C in the same program. Item 35: Familiarize yourself with the language standard. Recommended Reading An auto_ptr Implementation General Index Index of Example Classes, Functions, and Templates
• • • • • ([收起](#))

[More Effective C++_下载链接1](#)

标签

C++

编程

C/C++

计算机

经典

进阶

Programming

计算机科学

评论

[More Effective C++_下载链接1](#)

书评

本书写作于1996年，相比她的姊妹篇《Effective C++》（第一版1991年，第二版1997年，第三版2005年），本书没有后续的更新版本，距今已有十六年之遙，因而内容显得有些过时。本书内容的技术深度相对于《Effective C++》确实如书名所说更深一层，尤其是在第五章详细演示的各种...

我刚读完 Effective C++ 后，不打算读这本 More 的。理由是想先在开发实践中贯彻一些 Effective C++ 中的最佳实践原则，在实践中消化、反馈和总结。
后来闲暇时，选择性地读了 More 的一些章节，感觉如获至宝，就有粗有细地通读下来。

对于两本书的比较，我的感觉是：Effective...

成书是在 Effective C++ 3ed 之前，所以书的内容有一部分与 Effective C++ 3ed重复，它在书内批的 E 也是指 Effective C++ 2ed。而且书中一些指的特性在新版的C++标准中已经实现，尤其是在C++0X11标准中。虽然有这些不足，但一点也不失做为经典的份量。

Item 24: Understand the costs of virtual functions, multiple inheritance, virtual base classes, and RTTI
virtual ~C2(); // redefined function 应为： virtual ~C2(); // new virtual function
Item 25: Virtualizing constructors and n...

意味着同样的质量和收获
行文一如既往的流畅幽默，虽然有少数单词不认得，读起来却没什么困难
除了exceptions里的几个item没看，其他基本都看完了
闲暇时翻翻感兴趣的item，用不着学院派式地从头到尾逐个遍历，挺好！ next: 《Inside the C++ Object Model》

总体来说还是很不错的，Scott Meyers大师一贯的简洁、实用的建议。相比之前的《Effective C++》，这本的主题更加“高级”一些，换句话说对于普通程序员可能用户不如之前那本大。但还是很不错的，值得一读。越来越觉得好书需要读三遍才能真正吸收精华知识，粗略的算了一下，读一...

这本more effective c++延续了effective c++的高质量，侯捷老师的翻译同样也是很有味道。这本书从内容上来说讲的还是细致入微，给人以很不错的阅读体验，但是读过了effective c++和more effective c++之后自己越发察觉到了书本知识和编程能力之间转化的鸿沟。这种best pract...

Meyers两本effective C++系列都看过了。两者相比，Effective C++应该算是基础篇，*More*这本则更注重工程实践方面，介绍C++的高级技术。里面提到的Reference Count, Virtual Construtor, Smart Pointer这些技术在大型的C++项目都很常见，自己是配合着Google的Chromium看的...

这本书我看过几遍了，只是看过，每次拿起来的时候发现自己还有不理解的地方。看书不止看书的页数，而是看书的内容，无所谓时间，自己理解就好。这种书该时常翻，好好看。
书的内容和书名很匹配，讲的是通过书中的条款，如何写出更高效的代码。书的内容值得这个标题。看这本书之...

断断续续，虎头蛇尾，把这书翻过一遍（除了附录..）。开始的时候，看得比较慢，都后边就有点浮躁。
有些内容在编程过程中也有体会，看书就是再复习一遍，再学个书上对这种情况的称呼。也有部分内容没遇到过，看了也没能有深刻的印象，以后有时间再翻翻吧。看到智...

继 Effective C++ 之後，Scott Meyers 於 1996 推出这本「续集」。条款变得比较少，页数倒是多了一些，原因是这次选材比「第一集」更高阶，尤其是第五章。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 ctors、smart pointers、r...

侯捷喜欢的东西八成都是些匠气很足的东西；应该去好好读The C++ language,从源头上知道大部分C++的feature都是怎么来的，应该怎么用；看了这书只会对侯捷和此书作者佩服得一塌糊涂，但是编程水平什么的，提高不是很大。有时间读这书不如看The C++ Language.

[More Effective C++_下载链接1](#)