第一图书网, tushu007.com

<<深度探索C++对象模型>>

图书基本信息

书名：<<深度探索C++对象模型>>

13位ISBN编号：9787560924182

10位ISBN编号：7560924182

出版时间：2001-5

出版时间：华中科技大学出版社

作者：[美] Stanley B. Lippman

页数：320

译者：侯捷

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

内容概要

本书重点：探索“对象导向程序所支持的C++对象模型”下的程序行为。
对于“对象导向性质之基础实现技术”以及“各种性质背后的隐含利益交换”提供一个清楚的认识。
检验由程序变形所带来的效率冲击。
提供丰富的程序范例、图片，以及对象导向观念和底层对象模型之间的效率测量。

C++成山似海的书籍堆中，这一本不是婴幼儿奶粉，也不是较大婴儿奶粉，它是成人专用的低脂高钙特殊奶粉。

对于C++底层机制感兴趣的读者，这本书会给你“漫卷诗书喜欲狂”的感觉。

了解C++ Object Model，是学习Component Object
Model的最短路线。

如果你是一位C++程序员，渴望对于底层知识获得一个完整的了解，那么Inside The
C++ Object Model正适合你。

关于作者：Stanley Lippman目前是华特迪斯尼主题动画公司（Walt
Disney Feature Animation）的主要软件工程师。
他曾经在AT&T贝尔实验室领导cfront
3.0和2.1版的编译器开发小组。
他也是贝尔实验室中由Bjame
Stroustrup所领导的Foundation专案组中的一员，负责对象模型并研究C++程序开发环境。
Stan著有极为成功的C++
Primer一书，也发表过许多C++方面的论文。
Stan最近刚从C++
Report的编辑位置上“退隐”，他曾在那个位置上做了4年。
他的C++论述遍及全球。

关于译者：侯捷，海峡两岸著名的电脑技术自由作家，对于技术的钻研和发表，有独特的品性与坚持。
作品涵盖著、译、评三方面，散见于各种刊特、媒体、网站论坛，深受读者喜爱和尊敬。

作者简介

Stanley
B.Lippman的职业是提供关于C++和面向对象的训练、咨询、设计和指导。
他在成为一名独立咨询顾问之前，曾经是迪士尼动化公司的首席软件设计师。
当他在AT&T
Bell实验室的时候，领导了cfront 3.0版本和2.1版本的编译器开发组。
他也是Bjarne
Stroustrup领导的Bell实验室Fo

书籍目录

本立道生（侯捷 译序）
目录
前言（Stanley B.Lippman）
第0章 导读（译者的话）
第1章 关于对象（Object Lessons）
加上封装后的布局成本（Layout Costs for Adding Encapsulation）
1.1 C++模式模式（The C++ Object Model）
简单对象模型（A Simple Object Model）
表格驱动对象模型（A Table-driven Object Model）
C++对象模型（The C++ Object Model）
对象模型如何影响程序（How the Object Model Effects Programs）
1.2 关键词所带来的差异（A Keyword Distinction）
关键词的困扰
策略性正确的struct（The Politically Correct Struct）
1.3 对象的差异（An Object Distinction）
指针的类型（The Type of a Pointer）
加上多态之后（Adding Polymorphism）
第2章 构造函数语意学（The Semantics of constructors）
2.1 Default Constructor的建构操作
“带有Default Constructor”的Member Class Object
“带有Default Constructor”的Base Class
“带有一个Virual Function”的Class
“带有一个virual Base class”的Class
总结
2.2 Copy Constructor的建构操作
Default Memberwise Initialization
Bitwise Copy Semantics（位逐次拷贝）
不要Bitwise Copy Semantics！

重新设定的指针Virtual Table
处理Virtual Base Class Subobject
2.3 程序转换语意学（Program Transformation Semantics）
明确的初始化操作（Explicit Initialization）
参数的初始化（Argument Initialization）
返回值的初始化（Return Value Initialization）
在使用者层面做优化（Optimization at the user Level）
在编译器层面做优化（Optimization at the Compiler Level）
Copy Constructor：要还是不要？

摘要
2.4 成员们的初始化队伍（Member Initialization List）
第3章 Data语意学（The Semantics of Data）
3.1 Data Member的绑定（The Binding of a Data Member）
3.2 Data Member的布局（Data Member Layout）
3.3 Data Member的存取

Static Data Members
Nonstatic Data Member
3.4 “ 继承 ” 与Data Member
只要继承不要多态（Inheritance without Polymorphism）
加上多态（Adding Polymorphism）
多重继承（Multiple Inheritance）
虚拟继承（Virtual Inheritance）
3.5 对象成员的效率（Object Member Efficiency）
3.6 指向Data Members的指针（Pointer to Data Members）
“ 指向Members的指针 ” 的效率问题
第4章 Function语意学（The Semantics of Function）
4.1 Member的各种调用方式
Nonstatic Member Functions（非静态成员函数）
Virtual Member Functions（虚拟成员函数）
Static Member Functions（静态成员函数）
4.2 Virtual Member Functions（虚拟成员函数）
多重继承下的Virtual Functions
虚拟继承下的Virtual Functions
4.3 函数的效能
4.4 指向Member Functions的指针（Pointer-to-Member Functions）
支持 “ 指向Virtual Member Functions ” 之指针
在多重继承之下，指向Member Functions的指针
“ 指向Member Functions之指针 ” 的效率
4.5 Inline Functions
形式对数（Formal Arguments）
局部变量（Local Variables）
第5章 构造、解构、拷贝 语意学（Semantics of Construction，Destruction，and Copy）
纯虚拟函数的存在（Presence of a Pure Virtual Function）
虚拟规格的存在（Presence of a Virtual Specification）
虚拟规格中const的存在
重新考虑class的声明
5.1 无继承情况下的对象构造
抽象数据类型（Abstract Data Type）
为继承做准备
5.2 继承体系下的对象构造
虚拟继承（Virtual Inheritance）
初始化语意学（The Semantics of the vptr Initialization）
5.3 对象复制语意学（Object Copy Semantics）
5.4 对象的功能（Object Efficiency）
5.5 解构语意学（Semantics of Destruction）
第6章 执行期语意学（Runting Semantics）
6.1 对象的构造和解构（Object Construction and Destruction）
全局对象（Global Objects）
局部静态对象（Local Static Objects）
对象数组（Array of Objects）
Default Constructors和数组

6.2 new和delete运算符
针对数组的new语意
Placement Operator new的语意
6.3 临时性对象（Temporary Objects）
临时性对象的迷思（神话、传说）
第7章 站在对象模型的类端（On the Cusp of the Object Model）
7.1 Template
Template的“具现”行为（Template Instantiation）
Template的错误报告（Error Reporting within a Template）
Template中的名称决议方式（Name Resolution within a Template）
Member Function的具现行为（Member Function Instantiation）
7.2 异常处理（Exception Handling）
Exception Handling快速检阅
对Exception Handling的支持
7.3 执行期类型识别（Runtime Type Identification，RTTI）
Type-Safe Downcast（保证安全的向下转型操作）
Type-Safe Dynamic Cast（保证安全的动态转型）
References并不是Pointers
Typeid运算符
7.4 效率有了，弹性呢？

动态共享函数库（Dynamic Shared Libraries）
共享内存（Shared Memory）

媒体关注与评论

编辑推荐

《深度探索C++对象模型》涵盖了C++对象模型的语意暗示，并指出这个模型是如何影响你的程序的。
C++成山似海的书籍堆中，这一本不是婴幼儿奶粉，也不是较大婴儿奶粉，它是成人专用的低脂高钙特殊奶粉。
对于C++底层机制感兴趣的读者，这《深度探索C++对象模型》会给你"漫卷诗书喜欲狂"的感觉。
了解C++ Object Model，是学习Component Object Model的最短路线。
如果你是一位C++程序员，渴望对于底层知识获得一个完整的了解，那么Inside The C++ Object Model正适合你。

第一图书网, tushu007.com

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com