

<<多态与虚拟>>

图书基本信息

书名：<<多态与虚拟>>

13位ISBN编号：9789572229477

10位ISBN编号：9572229478

出版时间：1998

出版时间：松岗

作者：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>

## &lt;&lt;多态与虚拟&gt;&gt;

## 作者简介

## 【自序】

虑而後能得（自序）

## 故事

接触 C++ 大约是 1989 年的事。

那时候的 PC 以现在的眼光看，除了「蛮荒」之外没有更合适的形容词了。

横扫千军的 Windows 3.0 还没有诞生，如今以 C++ 编译器技术闻名於世的 Borland 公司那个时候也还没有踏进 C++ 领域，遑论动作更慢的 Microsoft。

当时我用的编译器是 Zortech C++。

将近 10 年的历史，并不表示我在 C++ 技术上就怎麼地超凡入圣了。

事实上初期我很排斥这个语言。

我没有用心去体会，也就看不到其中的美好。

我用数年的时间把心力全花在 Windows 技术的钻研。

那时候 C++ 风潮未开，我也不以为意。

决定好好研究 C++，不是为了要开发自己的 class library，而是为了要使用别人的 class library。

92 年软体界开始出现一些所谓的 application framework，是一种用来帮助程式员发展 Windows 应用软体的大型 C++ class library（如今也有非 C++ 语言的类似产品）。

为了运用这些有着巨大架构以及巨大能量的 C++ class library，我必须重拾 C++。

然後，为了更能操控掌握那些复杂的架构，同时也因为对技术原理有一股强烈的好奇心，我开始看 Microsoft Foundation Class（MFC）的原始码。

这使我有了一种异於常人的学习经验。

我看的是行销全世界、无数程式员使用、市场占有率最高的一套 C++ class library 的原始码。

从其中，我得到许多宝贵的知识与经验。

然後我慢慢体会了物件导向（Object Oriented）之美、多型（Polymorphism）之美、虚拟（Virtual）之美。

学习的路线，不一定每个人都相同。

但是黑暗中摸索的日子，实在不必人人走一遭。

對於位居物件导向精髓地位的「多型与虚拟」观念有了深刻体会之後，我有一股强烈的欲望，要把心得写下来。

於是 I 写了这本书。

## 层次

一般而言，C++ 是一个难学易用的语言。

C++ 的难学，初始在於其重重的布幕，布幕之中编译器对我们的程式码做了太多的手脚，使我们惯於循序思考的工程脑袋一无所措。

及长又面临新的思维模式，使我们必须扭转惯常的思考习惯。

C++ 的易用则在於其巨大的弹性，能够以多型（polymorphism）、虚拟（virtual）、模板（template）等种种方式，让现有的码去处理未知的、未来的资料型态。

## &lt;&lt;多态与虚拟&gt;&gt;

当然，易用必须先能用。

用不好或不能用的话，「写 C++ 程式」最後就成了只是「使用 C++ 编译器」，这是大家常拿来彼此调侃的笑话。

在「难学」的背景下，「易用」是使我们依然前仆後继的动力。

愈来愈多的大学资讯科系把 C++ 开在大一课程，这虽然说明 C++ 是多麽地重要，可也苦了资讯新兵们。

其实「难学」的最大症结，在於很难得有一本书，能够一针见血地指出多型与虚拟的重要性；在我们粗具语法基础之後，直接把我们导引到最核心最重要的思想，并且在建立这个思想体系的过程中，提供足够的必要基础。

我希望这本书能够做到这一点。

同样是使用 C++ 语言，有人进入了物件导向的殿堂，有人只学到 C++ 的语法；

有人练书法；

有人写毛笔字；

有人拿有毛的笔写字。

我们要向书法的境界挑战，不要只是拿有毛的笔写字。

定位

讲述 C++ 语法 (syntax) 的书籍很多，

讲述 C++ 语意 (semantics) 的书籍很少，

讲述 C++ 物件导向 (object oriented) 精神的书籍更少，

深入 C++ 物件模型 (object model) 的书籍则几乎没有。

这本书第一章对 C++ classes 的语法和语意做了一个梗概整理。

第二章谈 C++ 的物件模型，第三章谈型别转型 (RTTI 的重要服务)，第四章谈多型与虚拟的精神，第五章实作一个大型例子，示范如何在一个 class library 中完成 RTTI、Dynamic Creation、Persistence 等基础建设。

第六章为 Component Object Model (COM) 奠基。

很容易看得出来，你不能够完全依赖这本书学习 C++ 语法 (因为第一章不过才 96 页)。

但是在学习了 C++ 语法之後，你可以依赖这本书学习物件导向的精髓。

我自己一直存在一个疑惑。

别人常认为我写的是高阶技术书籍，我却认为我写的是高阶技术的入门书籍。

也许我挖掘得很深很广，但，不就是要彻底了解重要的基础知识，才得登堂入室吗？

这本书，我仍旧挖得很深，尤其是第二章和第五章。

但是我想粗具 C++ 语言基础的人应该都能循序接受。

我渴望证明，C++ 的难学，问题不在学生，而在老师。

当然，本书的读者，你，不能够只是一位程式设计领域的全然新手。

厚薄

## <<多态与虚拟>>

每次完成一本书，抱着厚厚的一叠完稿北上，父亲看到堆起来有 10 公分高的雪铜纸，总是对我说：『这麽厚的书，怎麽看』？而我总要解释『这是雪铜纸，比较厚。而且单面印刷，厚度加倍。经过雷射印表机的高温，它又有点卷...』。

但是想想，1000 页左右的书籍，再怎麽东折西扣也是厚得吓人。

渐渐地父亲不再问那个问题了，大概已经妥协於「电脑书都那麽厚」的印象之中。而我更是早就习惯看大部头的书，写大部头的书。

大家都习惯了吧！  
这两年 K。

书籍的厚薄，是因於内容的选择，与读者群的界定。  
很难得有这麽一个题材，我以 340 页的篇幅完成了它。

书籍很薄，又经过我精心的章节次序安排与起承转合，您就不要跳着看了吧。  
耐着性子，一页一页，从头到尾把它好好咀嚼消化一遍。  
然後，如果你愿意，写信给我，我希望听到你的喜悦。

批评也是我所欢迎的。

侯俊杰 1998.07.02 于新竹

<<多态与虚拟>>

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>