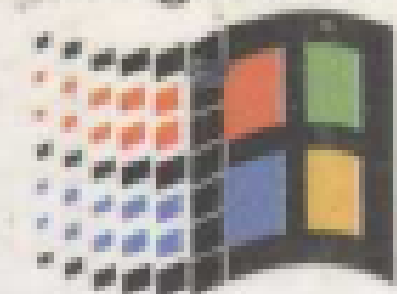


Designed for



Microsoft®
Windows NT®
Windows 98



CD-ROM
Included

Microsoft® 程序设计系列

Microsoft® Windows® 用户界面开发

提高用户经验的实践
和有效方法

北京大学出版社

Microsoft® Press

Microsoft Windows

用户界面开发

[美] Everett N. McKay 著

天宏工作室 译

北京大学出版社

·北京·

著作权合同登记 图字：01-1999-3653 号

Developing User Interfaces for Microsoft Windows

Everett N. McKay

本书版权为 Everett N. McKay 所有, 1999。(Copyright © 1999 by Everett N. McKay. All rights reserved.)

本书中文版由美国 Microsoft 出版社授权北京大学出版社独家出版, 2001。

本书封面贴有北京大学出版社的激光防伪标签, 无标签者不得销售。

版权所有, 翻印必究。

图书在版编目(CIP)数据

Microsoft Windows 用户界面开发/(美)麦凯(McKay, E. N.)著;天宏工作室译. —北京: 北京大学出版社, 2001.7

ISBN 7-301-00989-5

I. M… II. ①麦… ②天… III. 窗口软件, Windows IV. TP316.7

中国版本图书馆 CIP 数据核字(2000)第 78218 号

书 名: Microsoft Windows 用户界面开发

著作责任者: [美]Everett N. McKay 著 天宏工作室 译

责任编辑: 邱淑清 范彦 杨海涛

标准书号: ISBN 7-301-00989-5/TP·26

出版者: 北京大学出版社

地址: 北京市海淀区中关村北京大学校内 100871

网址: <http://cbs.pku.edu.cn>

电话: 出版部 62752015 发行部 62754140 编辑室 62757065

电子信箱: zpup @ pup.pku.edu.cn

排印者: 北京大学印刷厂

发行者: 北京大学出版社

经销者: 新华书店

787 毫米×1092 毫米 16 开本 28 印张 593 千字

2001 年 7 月第 1 版 2001 年 7 月第 1 次印刷

定 价: 64.00 元(含光盘)

北京大学出版社
Peking University Press

Microsoft® Press 是具备 **Microsoft®** 技术保证的出版社，您可以信赖我们所给予您的完整性、权威性、启发性及正宗的信息。要获得全方位及最完整的技术信息，请选择 **Microsoft® Press** 的产品。

敬请访问：

<http://www.microsoft.com/china/mspress>

<http://www.mspress.microsoft.com/china>

致谢

我没有真正写书的经验就写了这本书,就像我曾经没做充分准备去跑马拉松一样。像这两种情况,做完事情当然是靠自己,但过程还是需要别人善意的帮助。

首先我得感谢《Windows Developer's Journal》的编辑 Ron Burk,是他使我产生了写本书的想法,尽管他本人并没在意这一点。我就一篇文章向他提了一个建议,就 Microsoft Windows 程序的以后界面提供了一个简单的测试过程。要是他对我的方法不感兴趣的话,或者觉得我的陈述太抽象而一无用处,本书也就写不成了。我首先的反应就觉得,如果按他的详细建议阐述这个问题将需要一本又长又乏味的书,后来我意识到还是可以将明确实用的 Windows 用户界面开发信息写在一本相对简短有趣(希望如此)的书中,也就是您手中的这本书。在 1997 年 8 月的《WDJ》杂志上, Ron 在文章“如何写一本好的有关 95 用户界面设计的书”中给了我明确的说明,这对我帮助很大,我也是尽力按他的建议做的。

我很希望书中的观点都很新颖,这样大家也就会觉得我是个聪明人。事实上,书中内容只是综合了我作为 Windows 程序员的个人经验、用户界面专家的介绍,以及那些开发我每天都要用的卓越程序的用户界面专家和程序员的聪明才智而已。在此我要感谢那些以其对用户界面的独特见解而帮助过我的专家们: Cooper Interaction Design 的 Alan Cooper, Microsoft 公司的 Tandy Trower, Nielsen Norman 集团的 Donald Norman 和 Jakob Nielsen, Healtheon 的 Bruce Tognazzini, 以及 Blue Sky 设计有限公司的 Virginia Howlett。感谢 Microsoft 的 Hillel Cooperman、Chris Brown、Jan Miksovsky、Jordan Schwartz, 以及 Kent Sullivan, 他们在 1998 年 Microsoft 专业开发者大会上的发言对我来说非常及时,并给了我很多启示和帮助。也感谢您访问 Isys Information Architects 的 Brian Hayes, 其中的 Interface Hall of Shame and Hall of Fame(蹩脚的界面和出色的界面)Web 站点是对我的观点最好的检验。

我要特别感谢在我职业生涯中给予我进行用户界面开发机会的人们,尽管那时我的开发技术还不够出色。他们包括 Bob Headrick、Greg Smith、Rick Fisher、Dennis Clark、Richard Speers、Flora Perski、Tom Weisz、Jacci Winchester, 以及 Sean Cryan 等。感谢加州

Webb 学院的 Lawrence McMillin, 是他在并未认识到写作技巧重要性时指导我如何写书的。感谢我的硕士论文导师——MIT 的 Nancy Lynch, 是她使我记住了那些技巧。还要感谢 Meryle Sachs 和 Dave Myers, 他们使我从技术作家的角度理解了用户界面开发。

深深感谢 Microsoft 出版社给予我这个难得的机会。感谢我的项目编辑 Devon Musgrave 和我的技术编辑 Jack Beaudry, 感谢他们的帮助、指导和细心编辑, 使我能够专心致力于写作。从这次的经历中, 我对编辑过程充满了深深的敬意, 它是项繁重而枯燥的工作, 做得好了也会被读者忽视。特别感谢我的责任编辑 Ryan, 感谢他查找我可能出现的错误。感谢 Paula Gorelick、Alton Lawson、Karen Lenburg、Patricia Masserman、Joel Panchot、Julia Stasio, 以及 Bill Teel, 感谢他们出色的编辑和出版工作。

最后, 特别感谢我的妻子 Marie 和我的孩子 Philippe 和 Michèle, 他们忍受着我长年累月地呆在电脑前, 没有他们的支持是无法完成这项工作的。我在想, 下几周我要尽力回忆一下我在空闲时间都做了些什么。

前言

好的用户界面十分重要,实际上,它对程序的成功影响很大。对用户来说,程序的设计和技术影响的是整体功能和性能,而用户界面影响的是程序本身。界面不好,程序也就不好。不要指望用户去发觉与承认蹩脚的界面后面所包含的东西,他们根本不会这样做。

市场上众多的 Microsoft Windows 软件产品中,其中用户界面不好却取得商业成功的有几个呢?我熟知的只有一个流行的图像处理程序(在此隐去其名字)。当问到用户对此程序的看法时,他们的答案都是:喜欢程序,但讨厌界面。但这个程序还是有点特别的,它提供其他程序没有的复杂的图像处理功能,并且其界面虽然差劲,但具有用户十分喜欢的批处理功能。显然,如果多个程序竞争一片特定的市场,那么最终获胜的将是用户界面最好的产品。

因为软件开发的最终目的就是要赚钱(这无疑是我的目标),所以绝大多数软件开发者都费力费神地了解尽可能多的有关用户界面的设计知识。事实上,这一点并不让人惊讶,理由如下:

- 软件开发及其管理者通常主要关心的是进度,而很少关心软件质量。既然每个管理者都声称提供更出色的用户界面,大量的管理者就会声称其软件就要上市了。对程序员来说,软件的成功主要依赖其对快速开发工具的理解,光盯着质量,用户界面就会受到不利的影响。
- 经常有人鼓励程序员别去做用户界面。大多数人认为程序员做不出好界面。在大软件公司,界面设计通常由专门人员去做,他们一般是界面设计人员或图形设计人员,结果是程序员也就无意去提高自己界面设计能力。事实上,用户界面包含有相当一部分先进程序,上述做法无疑与此事实背道而驰。
- 用户界面设计既不是大多数计算机科学课程的内容,在大多数程序员杂志中也并不重视。有的程序员杂志根本不提界面设计的问题。
- 程序员们认为针对用户界面设计的许多讨论要么不关痛痒,要么理论性太强,所以没什么用。很多情况下确实如此,许多用户界面的书都集中在理论而不是实用上,

里面常常是胡乱塞了一些东西而已。

- 有些程序员可能根本注意不到自己用户界面的不足之处,他们只会发现别人设计的程序中不好的用户界面,而对自己设计的程序的用户界面不好效果则可能熟视无睹。

既然用户界面是现代软件开发十分重要的一部分,那么程序员也就要很清楚用户界面设计问题。现在的编程工作很少有根本不需要进行用户界面设计的。即使程序员所在公司的用户界面设计是由专门人员完成,但如果程序员不理解所设计的用户界面,那么他又怎么能够做出高质量的程序呢?又怎么能够识别问题,发现 bug(错误)或进行修改呢?理解 Windows 用户界面设计的基本原理,对每一位要开发高质量软件的程序员来说都是必要的。

读者

本书内容针对 Windows 程序员,特别是利用 MFC、Windows API 或 Microsoft Visual Basic 开发 Windows 应用软件或实用工具的程序员。书中的许多观点也适用于其他类型的编程工作,比如开发别的操作系统、Web 页,以及瘦客户机(thin-client)的 Windows DNA(分布式网间应用程序结构)程序,但在书中没有对它们进行专门的阐述。同样,书中的许多观点除适用于程序员外,对开发组的其他成员更为有用,如管理人员、质量保证测试人员,以及技术作家等,但书中没有专门针对这类读者进行阐述。

目标

为了向 Windows 程序员讲清楚用户界面设计的问题,我为自己设定了几个目标:

- 集中在专门的、实用的、有经验基础的信息上。主要介绍如何避免犯我曾犯过或曾见到的错误。
- 以简短、独立的章节进行阐述。
- 避免含糊的一般性的讨论,避免老生常谈、过于理论化,以及其他形式的无用的东西。
- 对用户界面设计不必面面俱到,而是集中在每个程序员都应该清楚的关键概念上。必要时请参阅书中推荐的其他材料。
- 尽量避免让读者(至少是本书针对的读者)进行“练习”。书中有关实践的地方都尽量提供了示例。

本书基本上概述了我认为程序员需要理解的有关设计出出色用户界面的基本内容。这些观点一直对我的工作帮助很大,我最终的目的就是希望与读者分享。

什么是用户界面？

那么什么是用户界面呢？有趣的是，我喜欢的用户界面书籍都未对此进行定义。我采用一个相对广泛的概念。我认为，用户界面不仅指窗口和对话框——它是用户使用程序时的整个过程。除了标准的用户界面元件，我认为用户界面还包括一些详细的内容，例如，如何安装程序，程序如何与 Windows 结合，程序加载的速度，程序如何进行响应，出错消息和文档的效果，以及程序的打印效果等。而且，开发用户界面是开发小组成员共同努力的结果，因此，理解其他成员的职能以及他们如何帮助您开发出出色的用户界面是十分重要的。

所有这些元素都将影响用户对程序的看法。试想用户碰到一个反应迟钝的程序会说什么呢？他们不会说“看起来这么好的程序有一个低效率的内部算法”。而是说“这个程序的用户界面太糟了”。不管用户界面是否直接导致了这样的慢速度，界面都是有责任的。因此，本书涉及了大量与用户整个操作经验相关的内容，而在绝大多数用户界面书籍中都没有讲到这样的内容。

老生常谈

Alfred Hitchcock 常常说他制作电影的一个首要目标就是避免老生常谈。我与他并不完全一样，我确实努力去避免标准用户界面的套话和含糊的一般性阐述，尽量避免采用类推的办法泛泛地描述用户界面的概念。讨论接吻的原则是没有必要的。（如果您真的不知道这些原则，那么我也不会在这里告诉您。）我不打算把什么都说成是艺术，而不是科学；我也不打算去说什么“让用户觉得友好”。书中确实有一章名为“保持简单”，这个题目显然含糊而一般化，但其中内容绝大部分是专门的实用技术，Windows 程序员利用它们能够简化用户界面。

我尽量多介绍实用的用户界面技术。我讨论的用户界面技术要么是标准技术，要么是我在使用或以前使用，或者在需要时很容易就可以使用的技术。我不建议您去做我从未做过的事情，也尽量不夸大事情的重要性。许多事情的确重要，但它们也不总是如此，很多好的用户界面技术使用不好或者使用过度必然对界面有害。我尽量同时讨论正确和错误的做法，以确保您不会做得过于夸张。

我所做的许多建议在开发过程前期是很容易做到的，也就是说，它们易于在新的开发项目中实现而不容易应用到已有的程序中。这确实是个缺憾。如果您针对的是一个现有的程序，又要保持进度的话，那么要修改所有的用户界面问题恐怕是心有余而力不足的事情，您必须有选择地进行修改。

用户界面开发是一项集体工作，实事求是地对待每个成员的进展情况是十分重要的。能够尽早地将每件事情与每个人结合起来，这无疑很好，但这显然是不现实的。没

有人能够这样工作,因此,如果我建议您早期利用技术作家的话,那么我提供的只是指导方针。

最后,规则都有例外,因此我也尽量列出例外情形。实事求是地说,谁也找不到一个没有例外的绝对的论述。

用户界面设计不再神秘

我相信用户界面设计不再像从前那样神秘。今天,典型的 Windows 98 用户界面比以前的 Windows 3.1 界面要好得多,操作更方便,功能更强,外观也更整洁和专业化。总的说来,由于 Windows API 和 MFC 这样的开发工具带有对用户界面更好的支持,所以 Windows 程序更为一致。现在,利用 Microsoft Visual C++ AppWizard,几秒钟就可以建立一个很好的应用程序框架。MFC 程序不进行任何附加编程就能够提供像打印预览等这样的高级功能。虽然这样的工具有助于更容易地实现很好的用户界面,但同时也阻碍了达到用户期望的质量程度。

最重要的是,我们对优秀的 Windows 程序的外观有了更清楚的了解,Windows 95 用户界面实际上设置了一套标准。我们现在已经理解了基本用户界面组件,如菜单、工具栏、状态栏、对话框、属性表,以及消息框等,并较好地理解了应用用户界面组件的方法及其局限性。例如,MDI 就不是所有程序的正确解决方法。

能够创造更好的用户界面方案,其操作必然要求更为简单。现在,要开发出色的用户界面,所要做的就是理解用户界面设计的基本概念,理解 Windows 标准,理解基本的用户界面选项,进行恰当的选择以帮助用户完成其工作。可以将很多优秀的 Windows 用户界面作为参考模型,但不要什么从头开始——那样您能做的就不多了。您没必要采用双面镜和录像机这样的用户测试设备,也不需要特意进行用户测试来选择对象的属性。如果在设计用户界面时采用标准控件并遵循 Windows 标准,那么这些繁重的工作都已经为您做好了,许多用户界面已经被广泛接受,而且我们已经知道这些控件都是非常有用的。

本书的一个重要目的就是帮助您完全理解 Windows 用户界面组件,并进行恰当的选择。

用户界面是为用户设计的

在设计用户界面时,需要记住设计的最终目的是帮助用户完成操作。实际上,最好的用户界面是对用户帮助最大,这也正是它们被称为“用户界面”而不是“程序界面”的原因。

尽管这样的出发点可能并不特别深奥,但程序员常常不是根据最适合用户的要求来进行用户界面设计,而是常常根据对自己的习惯和想法来进行设计。程序员加入不恰当的功能,往往是因为觉得这个功能很“酷”,或者觉得有趣、容易实现,而很少是因为

想方便用户的操作。

要创造好的用户界面,您就必须完全为用户着想,将他们的想法置于您的想法之上。

程序员能够创造出出色的用户界面

在一本优秀的书《About Face: The Essentials of User Interface Design》中,作者 Alan Cooper 坦率地说,程序员由于太关心算法的效率而设计不出出色的用户界面。的确,有的程序员设计的用户界面很蹩脚,但我不认为程序员注定就不能设计出出色的用户界面。我也关心算法的有效性,但这并不会削弱我开发便于使用的软件的能力。

而且,我认为这不仅是一个基本原理的问题,而且许多程序员根本就没有接受过有关用户界面设计基本原理的培训。用户界面设计不是典型计算机课程的一部分(至少我是这样的),在编程杂志中也不常见。向程序员灌输的常常是“如何……”之类的信息来完成他们的工作,而对用户界面设计却从不这样做。不能深刻理解这些基本原理,别说程序员,谁也设计不出出色的用户界面。毕竟,不能指望任何人在其没接受任何培训的领域创造出专家级的成绩。

我认为,大多数用户界面设计工作是由程序员完成的,不管您是否喜欢,理解这一点十分重要。很少有公司拥有专门的界面设计人员。(我本人做的大多数项目都是内核代码不多的界面设计,如果除去我的项目中的界面工作,那么就没多少东西留下了。)因此,很有必要提高程序员的用户界面设计水平,我也相信,通过正确的指导,程序员能够设计出出色的用户界面。

争议观点

有人认为,用户界面设计是相当个性化的话题,没有唯一的或正确的途径;因此,本书中给出的大多数建议并没有什么惊人之处,其他的软件开发资料中早就说过了。但是,我不仅坚信程序员能够设计出出色的用户界面,而且提供了一些并未被广泛接受的观点,特别是:

- 用户不是设计者,不能指望他们为您设计用户界面。用户可以向您提供有用的信息,对程序的实用性问题进行测试,但设计是由您完成的。用户无法把设计不好的程序变得好起来,充其量将其变得可以使用而已。我认为进行软件开发最好的方法是与用户及其他开发成员协作,制订方案,然后根据方案决定程序框架。
- 好的原型设计能够大大提高效率,但常常是事与愿违。我所见的许多原型设计工作都不过是浪费时间,并带来悲惨的结局。要想有效地进行原型设计,就需要解决一个专门的问题,即需要清楚且可实现的目标,并需要了解如何进行用户测试以获

题。

- 程序员主要对质量负责。程序员可以也应该对自己的用户界面进行测试,其最大障碍是如何作为一个用户进行测试并执行用户操作,像调试程序一样进行测试是于事无补的。我提供了许多程序员可以采用的用户界面测试技术。
- 用户测试的确有用,但首先必须进行程序员测试和 QA 测试。用户测试比其他任何测试都更为实用,但采用更有效的技术可以发现更多的实用性问题。

这些就是我的有争议的观点,您可以不同意它们,但我将尽力说服您。它们反映了我的个人经验,并且对我的工作一直很有帮助,根本忘不掉。

关于 Microsoft 软件

我在书中尽量提供可行的、基于经验的信息。理解好程序好在那里,差程序差在那里无疑是很重要的,因此有必要利用现实中关于用户界面正反两方面的例子进行论述。这样的例子既要实用,又要有经验基础,所以我认为应该挑选我和绝大多数程序员都经常使用的产品。这些都是我已用了几年而不只是几个小时的程序。

本书原稿中,我选择了几个 Microsoft 产品作为好用户界面的例子,并选择了几个非 Microsoft 产品作为不好的用户界面的示例。但是当 Microsoft Press 同意出版本书时,我立即意识到,这对我产生了一个问题。一本由 Microsoft Press 出版的有关用户界面设计的书只把 Microsoft 产品作为好例子,而把非 Microsoft 产品作为差的示例,这无疑是不明智的,这会让人觉得我有偏见,尽管我是无意如此的。

我的解决办法就是全部选择 Microsoft 产品作为示例(只有很少的几个例外)。这样达到一种平衡。很高兴 Microsoft Press 并没有反对我对 Microsoft 产品提出的批评。我知道有许多其他公司开发了十分杰出的用户界面,但遗憾的是,这样的产品没有一个是我经常要用到的。

目录

致谢	1
前言	3

第一部分 基础知识

第一章 了解标准	3
1.1 什么是标准?	3
1.1.1 《Microsoft Windows User Experience》	3
1.1.2 《Microsoft Manual of Style for Technical Publications》	3
1.1.3 Microsoft Windows 徽标设计要求	4
1.2 调查测试	5
1.2.1 测试问题答案	6
1.3 怎么办?	7
1.4 什么时候违背标准?	8
1.5 推荐书目	8
1.6 光盘资源	9
第二章 阅读其他有关用户界面的书	11
2.1 Alan Cooper 著《About Face》	11
2.2 Microsoft 著《Microsoft Windows User Experience》	13
2.3 Virginia Howlett 著《Visual Interface Design》	13
2.4 Donald A. Norman 著《The Design of Everyday Things》	14
2.5 Bruce Tognazzini 著《Tog on Interface》	15
2.6 Jakob Nielsen 著《Usability Engineering》	16
2.7 Laura Arlov 著《GUI Design for Dummies》	17
2.8 Peter Bickford 著《Interface Design》	17
2.9 推荐书目	18

2.10	光盘资源	19
第三章	建立统一的术语	21
3.1	术语的重要性	21
3.2	命什么名?	22
3.3	何时命名?	22
3.4	如何命名?	23
3.5	命名是严肃的事情	23
3.6	有问题的术语	24
3.7	用用户的语言说话	26
3.8	要避免的术语	27
3.9	推荐书目	27
第四章	建立一致的用户界面风格	29
4.1	外观与行为	29
4.2	指导方针的目标	30
4.3	指导方针的指导方针	30
4.4	创建资源模板	31
4.5	力争“出类拔萃”	33
4.5.1	用户界面指导方针示例	33
4.6	相关章节	52
4.7	推荐书目	53
4.8	光盘资源	54
第五章	关注其他程序	55
5.1	一般特点	55
5.2	学习容易	55
5.3	可配置性	56
5.4	操作	56
5.5	细节	56
5.6	案例研究: Visual C++	56
5.7	推荐书目	59
第二部分 理解用户		
第六章	初级用户与高级用户	63
6.1	用户的类型	63

6.1.1 初级用户	63
6.1.2 中级用户	64
6.1.3 高级用户	64
6.2 选择合适的界面功能	64
6.2.1 初级用户	64
6.2.2 中级/高级用户	67
6.3 针对所有类型的用户	68
6.4 不采用用户模式	69
6.5 仔细选择默认配置	69
6.6 相关章节	70
6.7 推荐书目	70
第七章 应用程序与实用程序	71
7.1 程序类型	71
7.1.1 应用程序	71
7.1.2 实用程序	72
7.2 选择合适的界面特性	72
7.2.1 应用程序	72
7.2.2 实用程序	73
7.3 规则的例外	74
7.4 程序种类	74
7.5 相关章节	75
7.6 推荐书目	75
第八章 用户不是设计者	77
8.1 确定目标用户	77
8.2 与目标用户交谈	78
8.3 以用户为中心的设计	79
8.3.1 用户不是设计者	80
8.3.2 原型设计的问题	81
8.4 以市场为中心的设计	82
8.5 以远景为中心的设计	83
8.5.1 设立产品的远景	84
8.5.2 实例研究：信息在您的指尖下	84
8.6 以远景为中心的设计和以用户为中心的设计	85
8.7 相关章节	86
8.9 推荐书目	87

第九章 用户不是您	89
9.1 用户具有不同的目标	89
9.1.1 典型的程序员目标	89
9.1.2 典型的用户目标	89
9.2 用户具有不同的知识	90
9.2.1 典型的程序员知识	90
9.2.2 典型的用户知识	90
9.3 用户有着不同的个人爱好	90
9.4 要问的关键问题	91
9.5 所有这些意味着什么?	91
9.6 相关章节	92
9.7 推荐书目	92

第三部分 设计概念

第十章 好的用户界面是看得见的	95
10.1 Windows 可视的暗示	97
10.2 可视的界面	97
10.2.1 命令	98
10.2.2 窗口	98
10.2.3 控件	98
10.2.4 可视反馈	99
10.2.5 其他	100
10.3 可视反馈	101
10.4 不可视界面	101
10.4.1 不可视的特性	102
10.4.2 不可视的功能	102
10.5 可视界面的例子	102
10.5.1 任务栏与 Alt + Tab	102
10.5.2 复选框列表与多选列表	103
10.5.3 “这是什么?”按钮与 Shift + F1 键	104
10.5.4 “属性”按钮与“属性”上下文菜单命令	104
10.6 相关章节	104
10.7 推荐书目	105
10.8 光盘资源	106