

Windows[®] 95
Windows NT[™]
国际软件开发指南

[美] Nadine Kano 著
郑全战 岚山 夏丽丽 译

清华大学出版社

(京)新登字 158 号

Windows® 95 Windows NT™国际软件开发指南

DEVELOPING INTERNATIONAL SOFTWARE FOR WINDOWS® 95 AND WINDOWS NT™ NA-
DINE KAND

Copyright © 1995 by Nadine Kano

Original English Language Edition Copyright © 1995 by Nadine Kano. Published by arrangement with
the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington
U. S. A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

中华人民共和国国家版权局著作权合同登记章 图字: 01-95-309 号

版权所有,翻印必究。

本书封面贴有 Microsoft Press 激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Windows® 95、Windows NT™国际软件开发指南/(美)卡诺(Kano,N.)著;郑全战等
译. —北京:清华大学出版社,1997. 12

书名原文: Developing International Software for Windows® 95 and Windows NT™
ISBN 7-302-02756-0

I. W… II. ①卡… ②郑… III. 窗口软件, Windows95-程序设计-指南 IV. TP316

中国版本图书馆 CIP 数据核字(97)第 25847 号

出版者: 清华大学出版社(北京清华大学校内,邮编 100084)

因特网地址: www.tup.tsinghua.edu.cn

印刷者: 清华大学印刷厂

发行者: 新华书店总店北京科技发行所

开本: 787×1092 1/16 印张: 40 字数: 943 千字

版次: 1998年3月第1版 1998年3月第1次印刷

书号: ISBN 7-302-02756-0/TP·1435

印数: 0001~3000

定价: 76.00 元

目 录

序	I
前言	III
第 1 章 对国际化的理解	1
1.1 地区	1
1.2 国际化	3
1.2.1 对 Windows 的国际化	4
1.3 本地化	7
1.3.1 本地化过程	8
1.3.2 国际性软件的发行	10
1.4 小结	11
第 2 章 设计全球软件	13
2.1 计划一个全球软件	13
2.1.1 在规格说明中标识出本地化需求	14
2.2 用户化特征	14
2.2.1 设计一个国际性的用户界面	16
2.2.1.1 位图、图符和声音	16
2.2.1.2 菜单和对话框	18
2.2.1.3 键盘	19
2.2.2 法律问题研究	19
2.2.3 保持功能的可用性	20
2.3 组建一个产品组	21
2.4 建立一种开发环境	22
2.4.1 隔离可本地化源程序	23
2.4.2 消除编译依赖性	25
2.5 巧妙的编程实践	28
2.5.1 避免硬代码可本地化的成分	28
2.5.2 使缓冲区足够大以容纳转换后的正文	31
2.5.3 不要限制确定为拉丁语的字符	31
2.5.4 不要假设字符总是 8 位的	32
2.5.5 保存你的文件格式不要将串本地化	33
2.5.6 避免某些编程捷径	34
2.5.6.1 对串的巧妙使用	34

2.5.7 语言特定代码.....	35
2.6 使事情对转换人员更加容易一些.....	36
2.7 测试考虑.....	37
2.8 小结.....	39
第3章 字符集编码	41
3.1 代码页模型.....	41
3.2 Windows 中的双字节字符集	42
3.2.1 DBCS 编程基础	43
3.2.1.1 显示操作.....	45
3.2.2 允许和不允许 DBCS 的程序比较	46
3.2.3 将核心代码库对 DBCS 开放	48
3.2.3.1 运行时优化.....	48
3.2.3.2 双重编译.....	50
3.2.3.3 宏和插入函数.....	50
3.2.4 如何在一个 DBCS 字符串中后移	51
3.3 UNICODE	52
3.3.1 Unicode 标准	53
3.3.1.1 Unicode 对某些应用系统的重要意义	56
3.3.1.2 Unicode 简化编码过程	56
3.3.2 在 Unicode 标准中的汉字统一.....	58
3.4 在混合环境中的兼容性问题.....	60
3.4.1 Unicode 和 Win32 API	61
3.4.2 Win32s 和 Windows 95 中的 Unicode	62
3.4.3 Windows NT 上的 Unicode	65
3.4.4 处理 Unicode 数据的 Win32 API 调用	68
3.4.5 Windows NT 上对 Unicode 开放的图板	70
3.4.6 Visual C++2 中的 Unicode	72
3.5 UNICODE 数据转换.....	75
3.5.1 移植基于 Unicode 的程序.....	82
3.6 小结.....	84
第4章 为本地化准备用户界面	86
4.1 Windows 资源文件	88
4.1.1 资源编译.....	89
4.1.2 本地化资源文件.....	91
4.1.2.1 一个好的本地化工具的特征.....	91
4.1.3 本地化对话框.....	93

4.1.4 串的存储	95
4.1.4.1 字符串表的注意事项	97
4.1.5 在 RCDATA 资源中存储常数	97
4.2 多语言资源	99
4.2.1 运行时改变用户界面	103
4.2.1.1 消息框	104
4.2.1.2 多语言 DLL	104
4.3 版本戳	109
4.4 小结	112
第 5 章 支持本地传统	114
5.1 格式化的日期和时间串	114
5.2 日历格式	118
5.3 货币和数字格式	119
5.4 地区标识	121
5.5 缺省用户和系统地区	123
5.6 数据处理	125
5.6.1 为提高效率而构建表格	125
5.6.2 使用排序键对数据进行排序和搜索	128
5.6.3 在 Windows 95 和 Windows NT 上运行一个单一的 二进制程序	130
5.6.4 注意具有重大影响的微小细节	131
5.7 小结	132
第 6 章 Microsoft Windows 中的多语言输入输出	133
6.1 多语言支持	134
6.1.1 在键盘布局之间进行切换	134
6.1.1.1 语言/版面配对	134
6.1.2 改变 Windows 95 上的输入语言	136
6.2 字体字符集属性	138
6.2.1 支持使用大字体显示多种语言正文	139
6.2.1.1 字体、Locale 和字符集标签	139
6.3 字体枚举和选择	142
6.3.1 EnumFontFamiliesEx	142
6.3.2 ChooseFont	143
6.4 Rich Text 和输出多语言文字	144
6.5 多语言文字布局	145
6.5.1 分析和预处理文字	145

6.6 小结	149
第7章 处理远东写作系统	150
7.1 用于本地化的工具	150
7.2 远东写作系统	151
7.2.1 中文	153
7.2.2 日文	153
7.2.3 朝鲜文	154
7.3 远东输入方法编辑器	154
7.3.1 日文 Windows 的标准 IME	155
7.3.1.1 输入日文字符	156
7.3.2 朝鲜文 Windows NT 3.5 的标准 IME	158
7.3.2.1 输入朝鲜文字符	158
7.3.3 中文输入方法	159
7.3.3.1 语音符号输入方法	159
7.3.3.2 仓颉(Chang Jei)输入方法	160
7.3.3.3 大易、数组和内部编码输入方法	160
7.3.3.4 激活 IME	160
7.3.3.5 输入中文字符	161
7.4 IME 系统的工作原理	163
7.5 对 Windows NT 3.5 上运行的应用系统的 IME 支持	165
7.5.1 增加响应 IME 消息的代码	165
7.5.2 增加对 IME 函数的调用	166
7.6 对在 Windows 95 上运行的应用系统的三种级别 IME 支持	167
7.6.1 无 IME 支持	168
7.6.2 部分 IME 支持	168
7.6.3 完全 IME 支持	170
7.7 用户化 Windows 95 上的 IME 支持	172
7.7.1 设置输入上下文	172
7.7.2 修改用户界面	173
7.8 垂直书写和打印	174
7.9 日本硬件	176
7.10 字体	176
7.10.1 兼容性问题	178
7.10.1.1 在日文 Windows 和美国 Windows 上运行一个单一的 二进制程序	180
7.10.1.2 在 Windows NT 3.5 和 Windows 95 上运行单一的二 进制程序	181

7.11	表意字符的排序规则	181
7.12	亚洲语言的换行规则	182
7.12.1	日文中的正文换行	182
7.12.2	中文中的正文换行	184
7.12.3	朝鲜文中的正文换行	186
7.12.4	泰国语中的正文换行	187
7.13	小结	188
附录 A	词汇	189
附录 B	一般拉丁变音符号和连字符	207
附录 C	选取的国际标点符号	210
附录 D	所选取语言的排序	211
附录 E	Microsoft Windows 中支持的代码页面	226
附录 F	地区专有的代码页面信息	228
附录 G	DBCS/Unicode 映射表	231
	代码页 932	232
	代码页 936	256
	代码页 949	283
	代码页 950	345
附录 H	代码页面	389
	1250 WINDOWS LATIN 2(中欧)	390
	1251 WINDOWS CYRILLIC(斯拉夫语)	397
	1252 WINDOWS LATIN 1(ANSI)	401
	1253 WINDOWS GREEK(希腊)	405
	1254 WINDOWS LATIN 5(土耳其语)	409
	1255 WINDOWS HEBREW	413
	1256 WINDOWS ARABIC	417
	1257 WINDOWS BALTIC RIM	421
	437 MS-DOS LATIN US	425
	708 MS-DOS ARABIC ASMO	426
	737 MS-DOS GREEK	427
	775 MS-DOS BALTIC RIM	428
	850 MS-DOS LATIN 1	429
	851 MS-DOS GREEK 1	430
	852 MS-DOS LATIN 2	431
	855 MS-DOS CYRILLIC	432
	857 MS-DOS TURKISH	433
	860 MS-DOS PORTUQUESE	434

861 MS-DOS ICELANDIC	435
862 MS-DOS HEBREW	536
863 MS-DOS FRENCH CANADA	437
865 MS-DOS NORDIC	438
866 MS-DOS CYRILLIC CIS 1	439
869 MS-DOS GREEK 2	440
10000 MACINTOSH ROMAN	441
10006 MACINTOSH GREEK	442
10007 MACINTOSH CYRILLIC	443
10029 MACINTOSH CENTRAL EUROPE	444
10079 MACINTOSH ICELANDIC	445
10081 MACINTOSH TURKISH	446
附录 I Microsoft Win32 有关国家和语言的信息	447
附录 J 支持 Win32 NLSAPI 的平台	455
附录 K Microsoft Windows 中支持的地区	457
附录 L 多语言 API 函数和结构	462
附录 M 字体签名位域赋值	464
附录 N Windows 95 输入法管理器 API 函数	481
附录 O Windows 95 输入法管理器 API 消息和参数	484
附录 P Microsoft Windows 的本地化版本	488
附录 Q 国际键盘布局	490
附录 R 国际键盘布局上的死键组合	549
附录 S 度量衡	557
附录 T 国际货币、日期和时间格式	559
附录 U 纸张和信封类型及规格	606
附录 V 国际地址格式	612
附录 W Microsoft 分公司	618
作者简介	625

对国际化的理解

开发全球化软件是一种不断平衡的行为。开发者及其管理人员常常完全忽略了创建一个高质量其它语言版本的工作和细节。如果你是一个开发者,你应该确实让你的管理者理解开发软件将涉及到什么?熟悉可能遇到的各种问题将有助于你在生产周期开始时作出决定,节约你的时间和金钱,但不后悔。

创建全球化软件的过程包括两个方面:“国际化”,它包括通用编程和设计;“本地化”,它包括为一特定市场而对产品进行的转换和本地化。使用由 Microsoft Win32 API 所提供的国际语言,是一个国际化的步骤,而调节对话框的大小和标准化术语则是本地化的步骤。开发人员书写代码,他们主要关注于国际化方面。但由于代码和特征的设计将影响产品的转换和客户化,因此开发者还必须理解基本的本地化概念。

本书的前两章提供了对 Microsoft Windows 上软件国际化的一些很有帮助的介绍。在这两章中讨论创建全球化应用系统的一般概念和技术,定义了在这本书的其它部分将使用的术语,并为以后章节涉及有关使用基于 Windows 的工具以及 Win32 API 函数的特殊技术细节提供了一种构架。

1.1 地 区

词汇

- **地区:** 与语言、国家和文化传统有关的用户环境特征。地区确定诸如排序顺序、键盘盘面、日期、时间、数字和货币格式之类的约定。在 Windows 中,地区提供的有关文化习俗的信息常常多于有关语言的信息。
- **地区敏感:** 依赖于不同的地区而表现出不同的动作或返回不同的数据。例如,Win32 排序函数将根据传给每个函数的地区参数而返回不同的结果。
- **国际标准:** 由一个政府或一个工业标准化组织所定义的一种语言规则、度量、培训指南或有关技术的约定。例如,字符集、键盘版面,以及诸如标点符号之类的文化传统。Windows 使用许多国际标准化组织(ISO)的命名规则。
- **语系:** 用于书写一种或几种语言的字符系统。字符表示被分离出来的声音、音节、或单词元素,并且根据一组通用的规则,如缺省的书写方向等而形成正文。
- **分隔符:** 用于分隔列表中的各项,标志数字中的千位,或代表一个小数点的符号。关于分隔符,不同的地区遵循不同的规则。

在地理上,一个地区是一个地点。在软件术语中,一个地区是与一个地点有关的一组信息。Windows 上的地区信息包括口语中的名字和标识符,书面语言中的书法,以及其它

文化传统(见图 1-1)。对地区敏感的标准包括键盘版面、缺省页大小和信封大小,常用打印机、常用监视器分辨率、字符集或字符编码的范围、正文的方向性(从左到右,或从右到左,水平的,或垂直的),以及输入方式。第 4 章和第 5 章将详细地讨论地区信息的有关方面。

地区	英语	法语	日语	阿拉伯联合酋长国 (UAE)阿拉伯语
国家	美国	法国	日本	阿拉伯联合酋长国
语言	英语	法语	日语	阿拉伯语
书法	拉丁文	拉丁文	片假名、平假名	阿拉伯文
正文方向	从左到右	从左到右	水平从左到右,或垂直从右到左	从右到左
Windows 定义代码页	1252	1252	932	1256
货币符号	\$	F	¥	.د.ا
长日期格式	January 15, 1995	15 January 1995	1995年 1月 15日	15. January 1995
短日期格式	1/15/95	15/01/1995	95/01/15	95/01/15
时间格式	1:00 PM	13:00	13:00	1:00
日历	阳历	阳历	阳历 或 阴历	阳历
缺省页大小	美国信纸 8.5in×11in	A4 (210×297mm)	A4 (210×297mm)	A4 (210×297mm)
小数点分隔符	.	,	.	,
列表分隔符	,	;	,	;
千位分隔符	,	空格	,	,

图 1-1 某些地区的部分缺省信息

Windows 支持大量的地区,以语言和语言子集来标志。语言子集通常对应于一个国家。认识到这一点的一种方法是考虑词组“在 Y 国家的 X 语言”。人们说写某种语言的方法可能在不同国家之间不会有太大的差别(如在澳大利亚、德国和瑞士所说的德语),但文化传统和国家标准却常常不同。

Microsoft Windows 95 支持 75 种地区,而 Microsoft Windows NT 3.5 只支持 50 种。Windows 支持 4 种中文地区(中华人民共和国、新加坡、中国香港特别行政区、台湾地区),6 种英语地区(澳大利亚、英国、加拿大、爱尔兰、瑞典、美国),以及 5 种法语地区(比

利时、加拿大、法国[标准的]、卢森堡、瑞士)。另外,Windows 95 还支持 16 种阿拉伯语地区(阿尔及利亚、巴林、埃及、伊拉克、约旦、科威特、黎巴嫩、利比亚、摩洛哥、阿曼、卡塔尔、沙特阿拉伯、叙利亚、突尼斯、阿拉伯联合酋长国、也门)。

1.2 国际化

词汇

- 国际化或全球化: 开发一个程序内核的过程。其功能设计和代码设计不是基于某种语言或地区上所作的假设,而其源代码库简化了一个程序不同语言版本的创建过程。
- 变音符: (1)置于一个拉丁字符上、下,或其中的任何一个标志,常用于表示与未标志状态时语音上的改变。(2)附在或覆盖一个前置的基本字符的字符。大多数的变音符是不占空间的字符,因而不会增加基本字符的宽度。
- 着重字符: 带有一个变音符的字符。
- 代码页: 字符的一个有序集合,其中每个字符有一个数字下标(代码点值)与之相对应。这个术语通常用于由 Microsoft Windows 3.1 或 MS-DOS 所定义的代码页环境中,也可以被称作为一个“字符集”。
- NLSAPI: 国际语言支持 API 的首字母缩略词。这套系统可以在包含国际语言支持(基于语言和文化传统的信息)的 32 位 Windows 上运行。

“国际性”、“国际意识”、“国际化”、“全球化”以及“世界性”都是一些用来描述设计能在多种语言环境中运行的程序的词汇。为了创建良好运行的转换软件,你将不可避免地对与某种语言有关的代码和功能进行再加工,而在开发之始创建一个可作为产品所有语言版本基础的国际化内核,将使你的工作更加有效。这将涉及对用户界面、功能集,以及对产品大多数可能的语言版本足够通用有效的一个代码库的设计。当然,一些用户化也是必要的,但若国际化版本所需的修改越少,产品的发布将越快。

国际化的目的是在一个产品的不同语言版本之间为用户提供一种一致的视觉,感觉和功能(术语一致性对产品包尤其重要)。用户希望本地化软件支持由产品的本土版本所支持的基本功能集,并且希望能达到同样的质量级别。他们还希望不同语言版本之间能方便地交互。因而将所有版本装入并有正确解释的单一文件格式是必要的*。例如,在比利时银行的职员可能使用你程序的法语版本来装入在你程序的荷兰版本中创建的一个文件。如果一种语言版本中的文件对其它语言版本是不可读的,或者如果在转换中有大量的错误,那么你的产品将得到一个不好的名声。而且也可以理解,当软件公司忽略了什么是用户所需要的主要功能时,国际用户将感到非常的失望。

创建 Windows 的小组努力地创建一种在任何一种语言中都可识别的用户界面,你可以从图 1-2a 和图 1-2b 中所示 Windows NT 3.5 的程序管理器的不同版本中看到,在英

* 出于对代码页的考虑,基于 Windows 95 的应用系统的文件可移植性并不总是很灵活。在日语 Windows 95 上运行的应用系统很容易装入英语文件,但在英语 Windows 95 上运行的应用系统则需要大量额外的工作才能装入并正确地显示一个日语文件。更多信息请参见第 3 章。

语和日语版本中的菜单和图符是一致的。如果你不说日语但是熟悉英语版本,你也可能找到使用操作系统的方法。

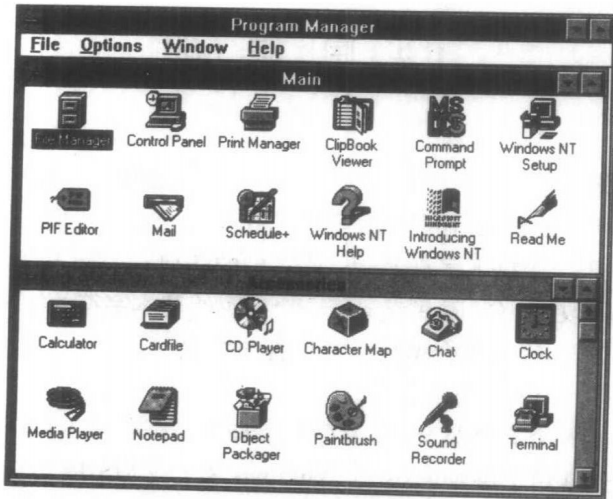


图 1-2a Windows NT 3.5 美国版本中的程序管理器

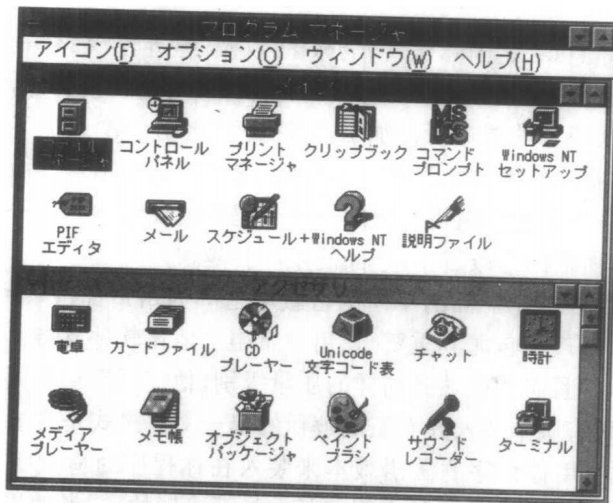


图 1-2b Windows NT 3.5 日语版本中的程序管理器

视觉上的一致性是一个产品国际化程度的一个度量标准,同样对国际性转换的支持也是一种标准。其中最基本的是对输入和显示国际性字符的支持。在过去,英文程序通常将合法字符限制为 ASCII 字符集。尤其是随着 Unicode 的出现(参见第 3 章),这已经不再是可接受的了。至少,欧洲用户应该可以在英语软件中键入着重字符。随着在 Windows 95 中可用的对混合语言数据的支持(参见第 6 章)和在 Windows NT 中可用 Unicode 的支持,现在已经有可能创建这样的软件,它能够显示在你程序的几乎任何一种语言版本中

创建的文件。

其它的国际性转换包括字符串排序和格式化日期、时间、数字及货币的规则。Windows 操作系统拥有大量的国际信息,你可以通过 Win32 NLSAPI 来访问(参见第 5 章)。这组函数将帮助你创建国际化核心代码,允许各种各样的用户输入完全符合文化习惯的数据。

用户无法直接看到的一部分国际化包括编码、测试、调试和翻译技术。如果你对产生翻译软件的需求不熟悉,那么你将对要解决的大量细节感到惊讶。这本书描述了你会遇到的各种情况,但不可能预见到所有的情况。开发者的工作是保证对一个程序的各种语言版本来说,无论需要什么改变,都能快速而方便地得以实现,同时不破坏任何功能。这需要合理地组织程序,从而在代码设计中预见如何实现一种语言版本,避免在转换软件中可能出现的错误编码。第 2 章详细地讲述了这些技术。

1.2.1 对 Windows 的国际化

词汇

- 单字节字符集(SBCS):一种字符编码,其中每个字符以一个字节来表示。单字节字符集限制为 256 个字符。
- 多字节字符集(MBCS):一种混合宽度字符集,其中一些字符由多于 1 个字节组成的字符集,双字节字符集(DBCS)是多字节字符集的一种特殊类型,包括一些由 1 个字节组成的字符和一些由 2 个字节组成的字符。
- 双向(BiDi)正文:可以从左到右也可以从右到左读的字符混合。例如,大多数阿拉伯和希伯来字符,是从右读到左的,而在阿拉伯或希伯来正文中的数字和引用的西方术语则是从左读到右的。
- 拉丁语:有古罗马继承下来的字符集,其中还包括以后添加的 26 个字符,用于书写在非洲、美洲、亚洲部分地区、欧洲和大洋洲所使用的语言。Windows 3.1 LATIN 1 字符集包括西欧语言和使用同一字母表的语言。LATIN 2 字符集包括中欧和东欧语言。
- 传统汉字:在新加坡和中国香港特别行政区、台湾地区使用的汉字字符集,与几千年前汉字的原始形式一致。
- 简体汉字:中华人民共和国使用的汉字字符集,包括作为传统汉字字符简化版的几千个表意字符。
- 软件开发包(SDK):一组用于创建 Windows 操作应用软件的工具和库。
- 设备驱动器包(DDK):一组为创建作用于硬件设备的基于 Windows 软件的工具和库。

邻近国家的软件工程需求常常非常类似。事实上,三种国际地理组就可以覆盖几乎所有的 Windows 应用市场:中东;远东;欧洲、俄国以及美洲。在你开始一个基于 Windows 的新产品设计,或者对一个现有产品确定其国际市场时,请检查一下在图 1-3 中所列目录的开发事项。

语言版本	语言字符集类型		语系	正文方向	SDK/DDK 版本	备注
欧洲	西欧, 中欧, 东欧, 希腊, 俄国, 土耳其, 印度尼西亚	单字节	拉丁语 希腊语 西里尔(现代俄语的本源)	从左到右	英文 Windows SDK 和 DDK	
中东	阿拉伯, 希伯来 拉丁语	单字节	阿拉伯语, 希伯来 拉丁语	双向	Windows SDK 和 DDK 补充	
远东	传统汉字, 简体汉字, 日语, 韩国语	多字节	假名, hangul, 表意字符	水平 和垂直	单独的 Windows SDK 和 DDK	输入方法
泰国	泰国	单字节	泰国语	从左到右	Windows SDK 和 DDK 补充	正文版面

图 1-3 基于开发问题的 Windows 的国际化目录

如果你进行了合理的计划,那么一旦你拥有了某个目录中一种或多种语言的核心功能和代码库,你可以确定在该目录中开发另一种语言版本的代价与可能的获利相比显得很小。一个好的例子是远东*目录(双字节语言),为 Windows 的中文、日文和韩国版本书写的系统共享同样的开发事项。一旦你拥有了一个日文版本的应用以后,那么开创一个丰富的韩国市场的应用所必需的开发步骤将极小。

Windows 95 开发组针对图 1-3 中所列的目录,创建了几种独立的代码库,在此基础上创建操作系统的本地化版本。这些目录中的每种语言在 Windows 95 语言版本之间,没有代码上的区别,只改变了用户界面的语言。中东、远东和泰国代码库是欧洲代码库的超集(为 Windows 的单字节版本所写的系统可以在 Windows 的双向或双字节版本上运行,反之则不然)。事实上,为了在 Windows 的一个欧洲版本中增加中东功能,你只需要额外安装几个库即可。Windows 3.x 对西欧语言、中欧和东欧语言、希腊语、土耳其语、中东语言、远东语言和泰国语有不同的代码库。Windows 未来版本的目的是对所有语言使用一种单一的代码库。

你的产品某些语言版本可能只要求在包装和一些小的软件构件上作些改动,诸如拼写检查器或示例文件等等,就能在不只一个地区进行推广。英文产品是一个主要的例子。只要一个程序的设计没有文化倾向,并且其功能也被适当地加以了国际化,那么它可以在

* 虽然“远东”已经不如以前那么常用了,但当指需要双字节字符集的 Windows 语言版本时,Microsoft 仍然使用这个术语。

澳大利亚、英国、加拿大、中国香港特别行政区、印度、爱尔兰、新西兰、南非、美国和许多其它国家销售。在设计阶段记住这一点将有助于你生产一个在全世界发行的程序。

西班牙语是另一种广泛应用的语言。几乎无需改变，一个西班牙语言程序就可以在整个拉丁美洲，西班牙和美国发行。虽然本地方言在许多说西班牙语的国家中都存在，但是计算机用户一般接受使用核心词汇转换的软件。Windows 支持许多在多个地区中使用的语言，包括：

- | | |
|--------|--------|
| ■ 阿拉伯语 | ■ 意大利语 |
| ■ 汉语 | ■ 挪威语 |
| ■ 荷兰语 | ■ 葡萄牙语 |
| ■ 英语 | ■ 罗马语 |
| ■ 法语 | ■ 俄语 |
| ■ 德语 | ■ 西班牙语 |

附录 P 列出了 Windows 的所有本地化版本。

1.3 本地化

词汇

- 本地化：为特定的国际市场对一个程序进行调节的过程，包括转换用户界面，调节对话框、用户化功能(如果必要的话)，以及测试结果以保证程序仍然工作。
- 本地化级别：创建各种语言版本所必需的转换和用户化的工作量。由风险和赢利平衡确定的级别范围，覆盖了从什么也不用转换到发行一个使用用户化的特性完全转换了的产品。
- 开放：修改程序代码来处理双向和双字节语言，如阿拉伯语和日语的输入、显示和编辑。
- 版面：显示正文的顺序和间隙。

国际化核心代码对将在本土以外发行的软件来说是一种基本要求。不过，根据目标市场和应用类型，你花费在本地化上的工作量和开销可以不一样(参见图 1-4)。一些市场，如东欧国家和中华人民共和国，目前来说要相对小一些，因此在这些市场上发行一个产品被认为是对一种未来的投资。而其它市场，如法国和德国，已经发展得相当好，打入其中需要大规模地投资和致力于相当高的标准。还有一些市场，如韩国和阿拉伯语言国家，虽然小但是发展迅速。有可能通过快速地发行一个部分本地化的应用系统并紧接着发行一个更加充分地进行了本地化的版本或其下一个版本，从而可以在这些小市场上抢占一席之地。例如，Microsoft 一般在发行其美国产品后几天内就会发行一个对中欧和东欧字符集开放的 Windows 新操作系统。本地化版本通常在随后的六个月内完成。

对于 Microsoft，由分部市场管理人员来决定 Windows 的每种语言版本的本地化级别。开发人员和转换人员为市场管理人员提供对每种本地化级别所需要的时间和金钱的估算。然后由分部和产品组来决定潜在的赢利是否值得这些开销。附录 P 列出了 Win-

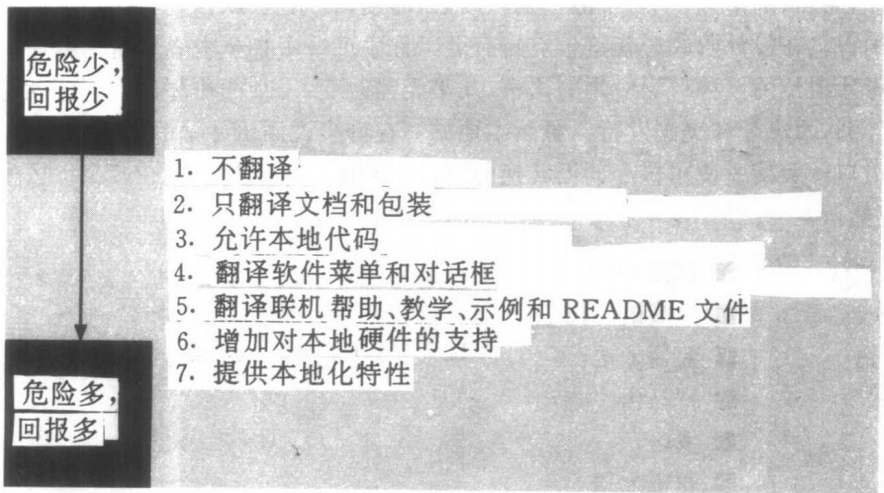


图 1-4 本地化级别

Windows 的本地化版本和其发行日期,这是 Microsoft 到目前为止所作出的决定。

在图 1-4 中列出的前两个级别,不转换和只转换文档和包装,几乎不需要什么开发——最多要求开发人员为转换人员解释一些技术问题。下一个级别,开放代码,不包括转换开销。术语“开放”主要适用于最初为西欧语言所设计的软件。开放软件允许用户创建以自己的语言形成的文档,即使用户界面并未被本地化,这类程序的中东和远东语言版本的代码必须被修改以便处理中东或远东语言正文的输入、排版;显示和打印(涉及的有关细节请参见第 3,6 和 7 章)。不允许用户输入或编辑正文的屏幕保护,多媒体标题和其他程序不需要被开放。

如果你产品的一个英文版本已经存在,那么考虑使用它来探测在一个完全本地化的版本可以使用之前用户将购买一个英文产品的市场。下面的列表为确定一种英文用户界面的可接受市场提供了一些指南。

- 小市场(例如,中欧和东欧以及印度尼西亚);
- 没有竞争者的产品市场;
- 许多人说英语的市场(例如,印度、以色列、韩国、荷兰);
- 目标用户说英语的市场(科技、医学和技术团体)。

如果你想在只有转换后产品才能获得成功的市场尽快地发行产品或者不想支付完全转换的开销,那么可以进行部分转换。作为一般的规则,涉及的转换越多,那么需要开发人员回答的问题也就越多,必须解决的问题也越多。

在竞争更加激烈的市场上,你需要修改功能特性,而且在某些情况下,还需要增加对本地所售硬件的支持。

1.3.1 本地化过程

词汇

- 可用性测试:通过一系列测试,观察用户完成一组指定的任务。可用性测试的目的是

确定新程序特征的直观程度和使用上的方便程度。

- Beta 测试：将预发布软件分发给用户和潜在的用户，以获得用户的反馈和错误报告。

创建本地化软件过程需要在产品小组各部分人员之间进行大量的交流。图 1-5 反映了一些基本的交流路线。从上到下，表示了国际性产品开发的艰苦历程。在核心开发阶段，开发组为负责正文转换和对话框大小调节的本地化组提供文件，并在必要时处理返回来进行编译的文件。本地化的可执行代码然后被传递给测试组，测试组将功能性问题报告给开发组，将用户界面上的问题报告给本地化组。由这三个组共同努力来解决问题，使这种循环继续。

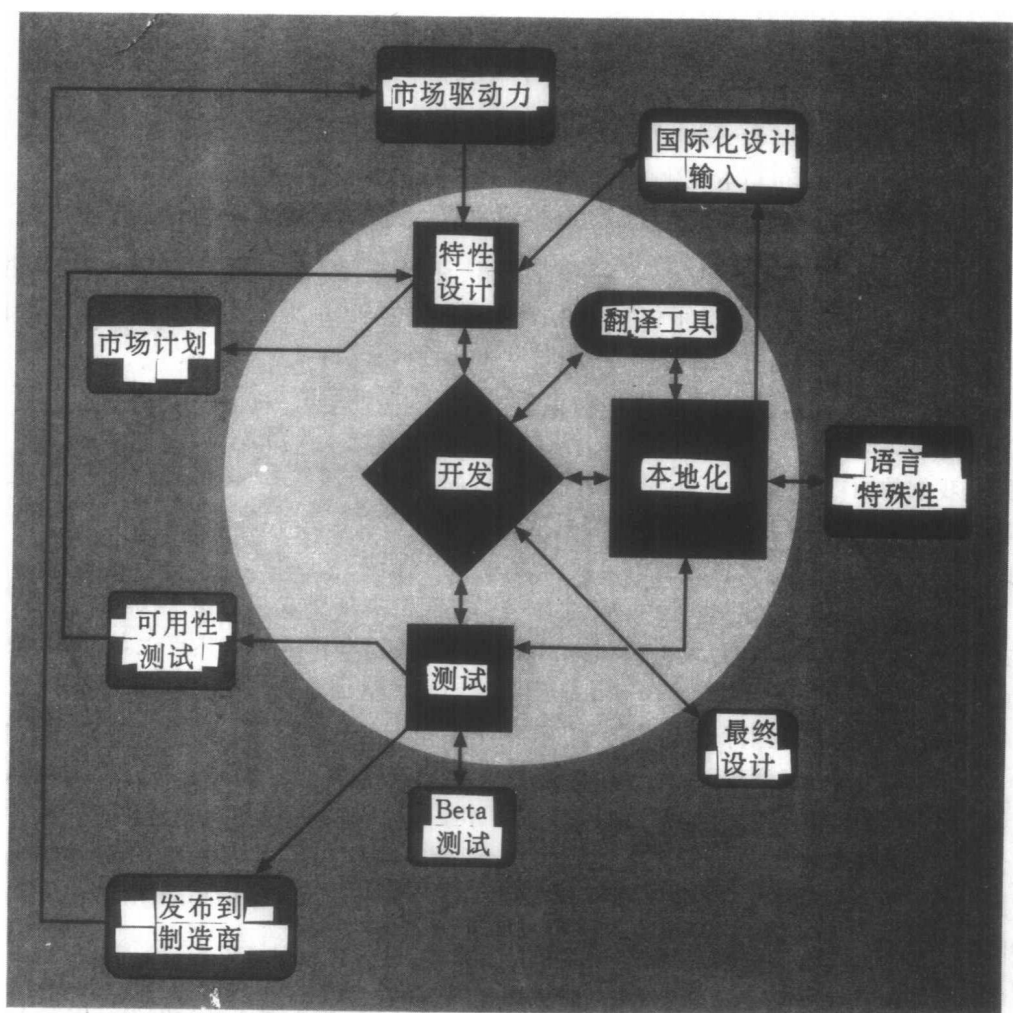


图 1-5 本地化过程

最省心的一类本地化过程是在最初的特性设计和代码设计阶段产品组就考虑了国际性的问题。市场力量、可用性测试、开发局限性也会影响产品的设计。转换和测试代码与