

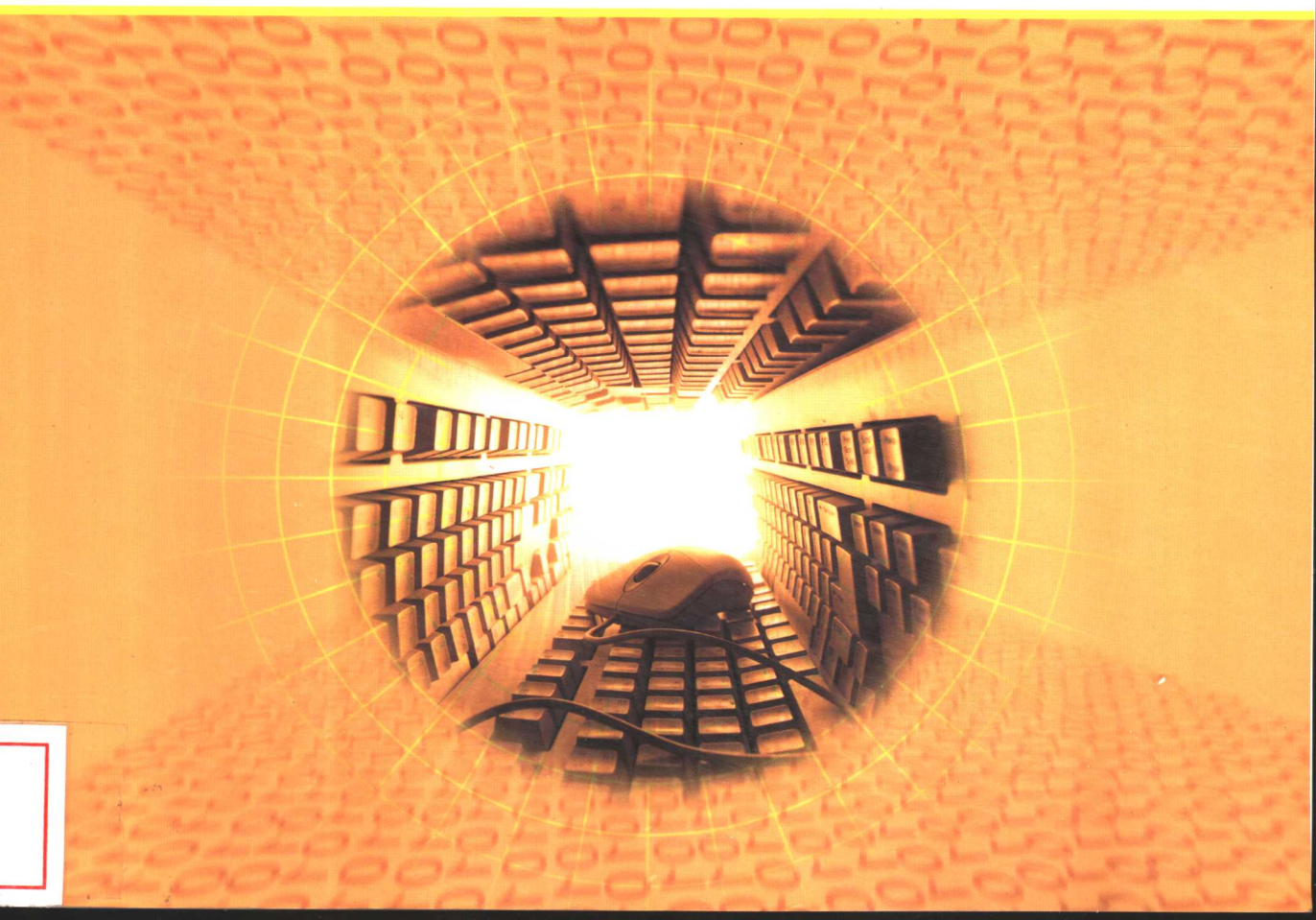
Visual Prolog

编程、环境及接口

雷英杰 邢清华 孙金萍 张 雷 编著

国防工业出版社

<http://www.ndip.cn>



Visual Prolog 编程、环境及接口

雷英杰 邢清华 编著
孙金萍 张 雷

国防工业出版社

·北京·

图书在版编目(CIP)数据

Visual Prolog 编程、环境及接口 / 雷英杰等编著.
北京:国防工业出版社,2004.1
ISBN 7-118-03282-4

I.V... II.雷... III.PROLOG 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2003)第 092229 号

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

北京奥隆印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 26 $\frac{3}{4}$ 612 千字

2004 年 1 月第 1 版 2004 年 1 月北京第 1 次印刷

印数:1-3000 册 定价:36.00 元

(本书如有印装错误,我社负责调换)

前 言

Prolog 语言是人工智能与专家系统领域最著名的逻辑程序设计语言。Visual Prolog 意指可视化逻辑程序设计语言,是基于 Prolog 语言的可视化集成开发环境,是 Prolog 开发中心(PDC)最新推出的基于 Windows 环境的智能化编程工具,其语言特性符合相应的国际标准 ISO/IEC 13211-1:1995。

Visual Prolog 是当今新一代开发智能化应用的强有力工具,它还支持基于网络的开发、数据库、多媒体与 C 语言集成等。Visual Prolog 在美国、加拿大、西欧、澳大利亚、新西兰、日本、韩国、新加坡等发达国家和地区十分流行,是国际上研究和开发智能化应用的主流工具之一。目前,我国在智能化领域的教学、研究、开发及应用正在迎来一个蓬勃发展的新时期,对这种工具软件的需求已经逐渐显现出来。国内已有不少 Visual Prolog 用户,一个 Visual Prolog 群体正在逐渐形成。预计不久的将来,在国际上已经十分流行的最新版本的可视化逻辑程序设计语言——Visual Prolog 将会在我国广泛流行开来,并将迅速成为我国研究和开发智能化应用的主流工具。

Visual Prolog 具有模式匹配、递归、回溯、对象机制、事实数据库和谓词库等强大功能。它包含构建大型应用程序所需要的一切特性:图形开发环境、编译器、连接器和调试器,支持模块化和面向对象程序设计,支持系统级编程、文件操作、字符串处理、位级运算、算术与逻辑运算,以及与其他编程语言的接口。

Visual Prolog 包含一个大型库,捆绑了范围广阔的 API 函数:包括 Windows GUI 函数族、ODBC/OCI 数据库函数族和 Internet 函数族(socket、ftp、http、cgi 等)。这个开发环境全部使用 Visual Prolog 语言写成,而且包含对话框、菜单、工具栏等若干编码专家和图形编辑器。Visual Prolog 支持 Windows 9x/Me/NT/2000/XP、OS/2、Linux 和 SCO UNIX。

Visual Prolog 非常适合于专家系统、规划和其他 AI 相关问题的求解,是智能程序设计语言中具有代表性且应用较多的一种语言。由于这种语言很适合表达人的思维和推理规则,在自然语言理解、机器定理证明、专家系统等方面得到了广泛的应用,在智能程序设计语言中占有相当重要的地位。Visual Prolog 不仅是优秀的智能化应用开发工具,而且与 SQL 数据库系统、Visual C++ 或其他 C++ 开发系统、Visual Basic、Delphi 或 Visual Age 等编程语言一样,已经成为适用于任何应用领域的强有力的通用开发工具。

智能化是当前计算机、自动化、通信、管理等信息科学技术领域中的新方法、新技术、新产品的重要发展方向与开发策略之一。信息处理的智能化与信息社会对智能的巨大需求是人工智能发展的强大动力。人工智能与专家系统曾取得过许多令人瞩目的成果,也走过不少弯路,经历过不少挫折。近几年来,随着计算机及网络技术的迅猛发展,特别是因特网的大规模普及,人工智能与专家系统的研究再度活跃起来,并正向着更为广阔的领

域发展。围绕人工智能与专家系统的研究和应用开发也迎来一个蓬勃发展的新时期。因此,引进与消化国际上已经广泛流行的功能强大和通用的智能程序设计语言、工具与环境,对于我国开发智能化应用系统十分必要。鉴于国内已有许多用户在使用 Visual Prolog,而这方面的中文资料比较缺乏,我们编写了本书,系统地介绍 Visual Prolog 语言的功能特点、编程技术、编程接口、开发环境及工具,相信对于开发智能化软件有启迪作用,也希望对我国在这一领域的教学、研究及智能化应用水平的提高起到良好的促进作用,且有益于国内同行在这一领域与国际主流保持一致。

作者非常感谢西安电子科技大学计算机学院的良好氛围和条件支持,特别要感谢王宝树教授、周利华教授、李荣才教授等的指导和鼓励,还要感谢空军工程大学计算机工程系吕辉教授、系统工程系杨建军教授、李为民教授、姜寿春教授、赵英俊教授等的支持和帮助,真诚感谢国防工业出版社的大力支持,正是由于这众多的帮助和支持才使本书得以呈献给读者。

需要特别指出的是,虽然作者竭尽所能,精心策划章节结构和内容编排,详细测试书中的每一个实例,尽可能简明而准确地表述其意,但由于水平和资料有限,书中的错误和不足之处在所难免,恳请读者不吝指正。

编 者

本书篇章结构

本书系统介绍 Visual Prolog (VP) 语言的功能特点、编程方法、开发环境及编程接口。全书分为三篇:编程指南、开发环境和编程接口。

第一篇 编程指南

详细介绍 VP 语言的功能特点及编程方法,包括第 1 章~第 9 章。

第 1 章 Visual Prolog 概述 介绍 Prolog 能做何用,VP 的功能特点,与其他编程语言的差别,以及相关的参考资料。

第 2 章 系统安装 介绍 VP 的安装过程和来自 Prolog 开发中心的技术支持。

第 3 章 编程起步 本章是熟悉整个系统大体功能的一个快速和轻松的导读。首先介绍 VP 的目标和规则等基本概念;然后从应用程序专家开始,介绍如何运行 VP 程序,如何使用代码浏览器、资源标识符浏览器、项目浏览器来学习所生成的代码。最后介绍系统最常使用的重要热键。

第 4 章 扩展“Hello World”应用程序 一步一步介绍如何使用 VP 的菜单编辑器和代码专家——对话框与窗口专家——来扩充“Hello World”应用程序,学会 VP 编程的基本技巧。

第 5 章 创建窗口 介绍如何创建一个新的源程序模块、创建一个新的窗口,如何为窗口生成缺省代码、检查来到窗口的事件,以及更多的画图操作。

第 6 章 给窗口添加控件 介绍如何使用工具栏专家、图形编辑器和菜单编辑器等工具给窗口添加、编辑和使用位图、工具栏、菜单等资源,如何画鼠标掠影、改变鼠标光标、设置鼠标热点,如何创建弹出式菜单及改变画图颜色等。

第 7 章 使用系统资源 介绍如何使用对话框包装专家来创建和使用时钟窗口(定时器)、图片窗口、树形窗口、编辑器窗口,如何处理剪贴板、控制打印机、给窗口添加更多的控件、使用列表框、创建对话框,以及使用 DDE。

第 8 章 创建帮助文件 介绍如何为应用程序创建添加编辑器功能和联机帮助等。

第 9 章 调试 Prolog 程序 介绍 VP 的调试器。

第二篇 开发环境

系统介绍 VP 的可视化开发环境(VDE),包括第 10 章~第 17 章。

第 10 章 使用 VDE 介绍 VDE 中使用的项目文件、项目窗口、信息窗口和项目模块树,以及代码浏览器、资源标识符浏览器等实用程序。还讨论了环境选项设置和全局初始化文件。

第 11 章 应用程序专家 介绍如何创建 VP 应用程序,包括项目命名、项目基目录、

多程序员支持、目标选项和 VPI 选项,以及项目中生成的文件和目录结构的描述。

第 12 章 建立、编译和链接 详细介绍 VP 的编译器选项和代码生成器选项,以及具有符号、规则和脚本功能的 Make 工具,还介绍了建立项目的有关菜单命令。

第 13 章 资源编辑 详细介绍资源命名、资源连接和资源引入,以及窗口与对话框编辑器、资源串编辑器、工具栏编辑器、菜单编辑器、图形编辑器和画图编辑工具。

第 14 章 编码专家 介绍如何利用代码生成专家来生成和编辑 VP 程序源代码,包括对话框与窗口专家、对话框包装专家及工具栏专家。

第 15 章 文本编辑器 介绍具有 VP 特色的通用文本编辑器。

第 16 章 帮助生成器 介绍如何使用这个非常方便的实用工具来生成 Windows 帮助文件。

第 17 章 调试器 介绍 VP 调试器及调试 VP 应用程序所需要的有关信息。

第三篇 编程接口

系统介绍 Visual Prolog 的可视化编程接口(VPI),VPI 是一种高级应用程序接口(API),其设计是为了利用现代操作系统和显示硬件的图形能力,为 Prolog 应用程序提供完善的用户接口。本篇内容包括第 18 章~第 24 章。

第 18 章 调用 VPI 简要介绍 VP 的编程接口和事件驱动的应用程序等概念,描述了 VP 应用程序调用 VPI 谓词的基本过程。

第 19 章 窗口与事件谓词 详细介绍窗口风格与窗口类型,窗口的创建、撤消、移动,改变窗口大小、状态,以及与窗口相关的处理谓词。还介绍与窗口有关的事件及事件处理,包括窗口的创建与撤消事件、用户请求关闭窗口事件、关闭 GUI 事件、窗口菜单事件、窗口鼠标事件和窗口键盘事件等,最后介绍了本地 GUI 事件、动态数据交换、控件通知事件、应用程序事件等的处理谓词。

第 20 章 控件操作谓词 介绍控件的操纵、控件的样式、来自控件的事件,以及各种标准控件、属主绘图控件和自定义控件等的操作谓词。

第 21 章 资源操作谓词 介绍如何在 VPI 应用程序中获取或检查由 VDE 资源编辑器创建的各种资源,首先介绍资源类型,然后详细介绍对话框、菜单、光标、图片、图元文件、剪贴板及插字符等 GUI 资源操作谓词。图片,介绍处理图片的谓词,包括图片格式转换,图片的创建、撤消、绘制、装入、保存、旋转。剪贴板,介绍如何从剪贴板得到数据,如何将数据放入剪贴板,以及测试剪贴板数据的谓词。图元文件,介绍图元文件的装入、播放、撤消、记录谓词。

第 22 章 绘图操作 详细介绍了绘图工具及其获得和设置方法,绘图谓词,包括画像素点、填充区域、绘制图标、绘制打开的图形、画填充图形,以及绘制文本、处理字体有关的操作谓词。颜色处理,介绍了用基本的红、绿、蓝三原色产生各种颜色的方法以及选择颜色的谓词。坐标系,介绍对话框的基本度量单元以及如何将一个窗口系统的坐标映射到其它窗口,还介绍了窗口的映射模式。打印,介绍了 VPI 应用程序中支持打印工作的谓词,包括开始打印、结束打印、设置页的打印方向、开始打印一新页、中止打印与获取打印设置。

第 23 章 系统级操作 主要介绍系统资源或与系统内部处理有关的操作谓词。定

时器,介绍启动、停止定时器的方法及谓词。动态数据交换,介绍客户机/服务器的角色、DDE 客户操作、DDE 服务器操作谓词,以及网络 DDE 等有关的谓词。错误处理,介绍 VPI 中的错误处理器以及如何安装新的错误处理器。本机 API 函数,介绍调用本机 GUI 函数的谓词,在 VPI 之外创建窗口的方法,VPI 窗口的本机事件,获得本机窗口句柄和设备上下文的谓词。运行外部应用程序,详细介绍从 VPI 程序中调用外部程序的谓词。执行联机帮助,介绍 VPI 中激活联机帮助的谓词。

第 24 章 VPI 程序包 介绍各种支持程序包。对话框程序包,全部由 Prolog 源程序编写而成,包括对话框的创建和初始化,对话框的事件处理、对话框论域,获取、设置对话框中控件的值,控件状态的修改等。编辑器 API,介绍编辑窗口的创建谓词、从编辑窗口获取文本谓词、增加新文本谓词以及与编辑器有关的其他各种谓词。消息窗口程序包,提供一个处理简短消息对话框的工具。工具栏程序包,介绍工具栏的创建、大小调整及移动等谓词,还介绍了在工具栏中创建单选按钮的方法。目录树程序包,介绍目录树工具包中的谓词和论域。属主绘图程序包,介绍会使属主绘图更为容易一些的例子文件。

附录 A 目录结构 包含有一个关于 VP 系统文件和子目录的描述。

附录 B 范例程序 MyProj 源代码 包含编程指南所举例题的完整的 VP 源程序代码。

内 容 简 介

本书系统介绍 Visual Prolog (VP) 语言的功能特点、编程方法、开发环境及编程接口。全书分为三篇:编程指南、开发环境和编程接口。第一篇编程指南,详细介绍 VP 编程方法,叙述如何使用应用程序专家、对话框与窗口专家、对话框包装专家和工具栏专家等代码专家,以及代码浏览器、资源标识符浏览器、项目浏览器、图形编辑器和菜单编辑器等工具进行智能化应用程序的工程开发。第二篇开发环境,详细介绍 VP 的开发环境 VDE,包括项目管理代码浏览器、资源标识符浏览器等实用程序和应用程序专家、资源编辑器等代码专家及帮助生成器、VP 调试器等。第三篇编程接口,详细介绍 VP 的编程接口 VPI。这个 VPI 是专为 Prolog 应用程序设计的高级 API,基本上不受平台限制,其内容包括处理诸如窗口、控件、菜单及事件等 GUI 元素的所有 VPI 谓词和各种支持包。

本书取材新颖,内容详实,逻辑严谨,语言通俗,理例结合,注重应用。书中包含大量经过严格测试的典型实例,便于教学、实习和自学。

本书可供高等院校计算机、自动化、信息、管理、控制与系统工程等专业本科生或研究生作为人工智能语言的教材或教学参考书,也可供从事人工智能、知识工程、智能决策等研究和应用开发的科技工作者以及其他有关专业的师生自学或参考。

目 录

第一篇

编程指南

第 1 章 Visual Prolog 概述	1
1.1 Prolog 能作何用?	2
1.2 Prolog 与其他语言有何差别?	13
1.3 Visual Prolog 的相关资料	14
1.4 联机帮助文件	16
1.5 HTML 格式的资料	17
1.6 符号约定	17
1.7 系统要求	17
1.8 如何开始学用 Visual Prolog	18
第 2 章 系统安装	19
2.1 直接从 CD-ROM 运行 Visual Prolog	19
2.2 从 Visual Prolog 以前版本升级	19
2.3 安装过程	19
2.4 从 Prolog 开发中心获得支持	21
第 3 章 编程起步	22
3.1 测试目标	22
3.2 规则	24
3.3 从应用程序专家开始	26
3.4 运行程序	27
3.5 学习所生成的代码	28
3.6 重要的热键	30
第 4 章 扩展“Hello World”应用程序	31
4.1 用菜单编辑器修改菜单	31
4.2 使用对话框与窗口专家	33
第 5 章 创建窗口	37
5.1 创建新的源模块	37
5.2 创建新窗口	39
5.3 为窗口生成缺省代码	40
5.4 检查来到窗口的事件	43
5.5 更多的画图操作	46
第 6 章 给窗口添加控件	47

6.1	创建 SWEEP.PRO 模块	47
6.2	创建新菜单(Sweep 菜单)	47
6.3	创建位图	48
6.4	创建工具栏	49
6.5	创建窗口	51
6.6	工具栏专家	52
6.7	画鼠标掠影	55
6.8	处理工具栏	57
6.9	清整处理	58
6.10	改变鼠标光标	58
6.11	设置光标热点	59
6.12	创建弹出式菜单	60
6.13	改变画图颜色	61
第 7 章	使用系统资源	62
7.1	使用定时器——时钟窗口	62
7.2	图片窗口	63
7.3	创建 Tree 窗口	64
7.4	创建编辑器窗口	66
7.5	处理剪贴板	68
7.6	打印	69
7.7	给窗口添加控件	70
7.8	使用列表框	73
7.9	目录浏览器代码	75
7.10	创建对话框	77
7.11	执行 DDE	82
第 8 章	创建帮助文件	84
8.1	对新节点创建一个链	85
8.2	格式化段中文本	85
8.3	格式化段	86
8.4	插入位图	87
8.5	测试帮助文件	87
8.6	给应用程序增加联机帮助	88
8.7	联机帮助框架	88
第 9 章	调试 Prolog 程序	91
9.1	调试器启动方式	91
9.2	浏览谓词	92
9.3	单步执行	92
9.4	设置断点	92
9.5	运行程序	93

9.6 浏览事实数据库.....	93
9.7 其他菜单项.....	93

第二篇

开发环境

VDE 概述	97
关于本篇	98
第 10 章 使用 VDE	99
10.1 菜单	99
10.2 项目文件.....	101
10.3 项目窗口.....	102
10.4 信息窗口.....	105
10.5 代码浏览器.....	105
10.6 资源标识符浏览器.....	106
10.7 项目模块树.....	107
10.8 环境选项设置.....	109
10.9 外部工具目录.....	112
第 11 章 应用程序专家	113
11.1 通用设置.....	113
11.2 目标选项.....	115
11.3 VPI 选项.....	118
11.4 其他选项.....	119
11.5 用户信息.....	123
11.6 帮助生成器.....	123
11.7 项目文件目录.....	124
11.8 新建源模块.....	125
11.9 VDE 中的多程序员支持	127
第 12 章 建立、编译和链接.....	131
12.1 项目选项.....	131
12.2 Make 工具	137
12.3 建立项目命令.....	140
第 13 章 资源编辑	143
13.1 资源命名.....	143
13.2 资源链接.....	144
13.3 引入资源.....	145
13.4 窗口与对话框编辑器.....	145
13.5 资源串编辑器.....	154

13.6	工具栏编辑器	155
13.7	菜单编辑器	159
13.8	图形编辑器	160
13.9	画图编辑工具	165
13.10	位图与图标相互转换	167
第 14 章	代码专家	169
14.1	对话框与窗口专家	170
14.2	对话框包装专家	175
14.3	工具栏专家	179
第 15 章	文本编辑器	181
15.1	启动编辑器	181
15.2	编辑器命令	182
15.3	编辑器选项	187
第 16 章	帮助生成器	189
16.1	启动帮助生成器	189
16.2	帮助生成器选项	189
16.3	创建帮助系统	191
16.4	帮助主题	192
16.5	引入帮助	203
16.6	建立帮助文件	204
16.7	帮助生成器文件菜单	205
16.8	帮助生成器查找菜单	205
第 17 章	调试器	206
17.1	概述	206
17.2	生成调试信息	207
17.3	启动调试器	207
17.4	装载程序	208
17.5	视图菜单	208
17.6	设置断点	212
17.7	机器级窗口	214
17.8	运行菜单	215
17.9	调试器配置	216
17.10	改变源文件路径	218

第三篇

编程接口

VPI 概述	220
---------------	-----

关于本篇	220
第 18 章 调用 VPI	223
18.1 编程接口	223
18.2 事件驱动的应用程序	224
18.3 调用 VPI	225
第 19 章 窗口与事件谓词	227
19.1 窗口风格	227
19.2 窗口类型	228
19.3 创建与撤消窗口	230
19.4 窗口矩形	233
19.5 改变窗口状态	237
19.6 改变窗口元素	238
19.7 查找窗口	239
19.8 窗口更新	240
19.9 事件处理	241
19.10 窗口菜单事件	247
19.11 窗口鼠标事件	248
19.12 窗口键盘事件	249
19.13 其他窗口事件	250
19.14 CONTROL_INFO 论域	253
19.15 应用程序事件	254
第 20 章 控件操作谓词	256
20.1 操纵控件	256
20.2 风格标志	257
20.3 来自控件的事件	262
20.4 创建控件	262
20.5 各种类型的控件	263
20.6 列表按钮	269
20.7 列表编辑框	269
20.8 滚动条	270
20.9 属主绘图控件	272
20.10 自定义控件	272
第 21 章 资源操作谓词	275
21.1 资源类型	275
21.2 对话框	277
21.3 菜单	283
21.4 光标	285
21.5 图片	287
21.6 窗口类	289

21.7	图元文件	291
21.8	剪贴板	292
21.9	插字符号	294
21.10	属性标志	294
第 22 章	绘图操作	298
22.1	绘图工具	298
22.2	绘图谓词	301
22.3	画文本	302
22.4	字体处理	304
22.5	颜色处理	305
22.6	坐标系	307
22.7	打印	310
第 23 章	系统级操作	313
23.1	定时器	313
23.2	动态数据交换	313
23.3	错误处理	323
23.4	本机 API 函数	324
23.5	运行外部应用程序	326
23.6	执行联机帮助	327
第 24 章	VPI 程序包	332
24.1	对话框程序包	332
24.2	编辑器 API	336
24.3	消息窗口程序包	341
24.4	工具栏程序包	349
24.5	目录树程序包	351
24.6	属主绘图程序包	353
附录 A	目录结构	372
A.1	BIN 目录	373
A.2	LIB 目录	374
A.3	VPI 目录	375
A.4	INCLUDE 目录	377
A.5	FOREIGN 目录	378
A.6	WINBIND 目录	378
A.7	SQL 接口	378
A.8	VDESRC 目录	379
A.9	TXTEXAMP 目录	379
A.10	DOC 目录	379
A.11	UPGRADE 目录	379
A.12	OOP 目录	380

A.13	WWW 目录	380
附录 B	范例程序 MyProj 源代码	381
B.1	CLOCK.PRE	381
B.2	CLOCK.PRO	381
B.3	CROSS.PRE	384
B.4	CROSS.PRO	384
B.5	DIALOGS.PRE	386
B.6	DIALOGS.PRO	387
B.7	DIRLIST.PRE	388
B.8	DIRLIST.PRO	389
B.9	EDWIN.PRE	392
B.10	EDWIN.PRO	392
B.11	MYPROJ.PRE	396
B.12	MYPROJ.PRO	396
B.13	PJCTURE.PRE	403
B.14	PICTURE.PRO	403
B.15	SWEEP.PRE	405
B.16	SWEEP.PRO	405
B.17	TREE.PRE	410
B.18	TREE.PRO	410

第一篇

编程指南

第 1 章 Visual Prolog 概述

Prolog 是全世界计算机科学家和人工智能专家多年来共同研究工作的结晶。Prolog 的第一个正式版本由法国马赛大学 Alain Colmerauer 于 20 世纪 70 年代作为一种逻辑程序设计工具研制的。其结果是诞生了一种远比当今 Pascal 和 C 这样著名的编程语言功能更加强大的语言。一个特定应用的 Prolog 程序,典型情况下只需要对应 C++ 程序的十分之一程序行。

今天,Prolog 是人工智能应用编程和专家系统开发的一个非常重要的工具,这是 Prolog 流行起来的一个十分重要的原因。更多的“用户友好性”和程序智能化的要求是使 Prolog 流行起来的另一个原因。而 Prolog 最主要的优点是非常公平地适用于任何应用领域:通过让程序员建立对象和进程之间的逻辑关系,使复杂问题更容易从本质上求解,而且产生的程序在其生命周期内更容易维护。定制知识库、专家系统、自然语言接口和智能信息管理系统等这些应用,都是当前使用 Visual Prolog 进行程序设计的领域范围。

Prolog 已经走出了人工智能实验室,PDC 的 Visual Prolog 是一个商业上富有竞争力的通用开发环境。Visual Prolog 因其容易增加程序甚至网站的智能化特性,而日益变成许多开发者选择的工具。

Prolog 是一种众所周知的说明性语言。这就是说,只要给出求解问题所需要的事实和规则,Prolog 将使用演绎推理求解编程问题。这与传统的过程性编程语言如 C、BASIC 和 Pascal 等形成了鲜明的对照。在过程性语言中,程序员必须提供一步一步的指令,准确地告诉计算机如何求解给定的问题。换句话说,程序员必须预先知道如何求解这个问题。而另一方面,Prolog 程序员只需要提供对问题的描述和求解的基本规则。此后,Prolog 系统将确定如何找到一个解。

由于这种说明性(而不是过程性)方法,众所周知的错误来源,诸如循环操作次数多一次或少一次这样的错误,一开始就被排除了。Prolog 鼓励程序员从结构良好的问题描述开始,因而实际上,Prolog 也可以被用作指定产品的规格说明工具和实现工具。

Visual Prolog 与 SQL 数据库系统、C++ 开发系统和其他语言工具如 Visual Basic、Borland 的 Delphi 或 IBM 的 Visual Age 一样,都致力于同样的目标。当今有一些组织趋向于用数据库技术来求解一切问题,但这种途径常常在开发时间和最终系统的性能两方面导致不良的结果。用 Prolog 开发的应用程序具有更优越的性能和用户友好性、更短的