

高等院校计算机辅助设计教材 (2)



AutoCAD 2002 开发教程

唐亮 张帆 卫朝富 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

高等院校计算机辅助设计教材 (2)



AutoCAD 2002 开发教程

唐亮 张帆 卫朝富 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

这是一本学习 AutoCAD 2002 定制与开发的高等院校 CAD 教程。本书系统和完整地介绍了 AutoCAD 2002 定制的方法、AutoLISP 语言基础知识和 Visual LISP 的开发环境,并通过实例详细讲解如何定制模板、菜单系统、线型和填充文件,如何对 AutoCAD 进行二次开发。

全书分为三篇,共 16 章。第一篇(从第 1 章到第 3 章)主要讲述基本的定制方法,包括对 AutoCAD 2002 的优化、定义模板和菜单系统等相关内容。第二篇(从第 4 章到第 7 章)讲述较为深层的定制内容,包括线型和填充文件的定义、语言支持及编程接口等内容。第三篇(从第 8 章到第 16 章)讲述 Auto LISP 和 VisualLISP 的相关内容,包括 AutoCAD 的语言开发环境,如何使用 AutoLISP 做程序开发,如何维护 VisualLISP 应用程序以及如何设计和管理对话框。

本书主要以结合实例的方式,由浅入深地进行系统的阐述,内容丰富而全面,对 AutoCAD 定制与开发的初学者,这是一本具有较强指导性和实用性的工具书。

本书适合于大专院校 CAD 专业师生、AutoCAD 2002 定制与开发的初学者,也可作为社会 AutoCAD 定制与开发培训班的教材。

本书的技术咨询邮箱地址为 Jcxt@sina.com。

- 系 列 书 名 : 高等院校计算机辅助设计教材(2)
书 名 : AutoCAD 2002开发教程
文 本 著 者 : 唐亮 张帆 卫朝富
责 任 编 辑 : 朱培华
出 版、发 行 者 : 北京希望电子出版社
地 址 : 北京市海淀区知春路63号卫星大厦三层 100080
网 址: www.bhp.com.cn E-mail: lwm@bhp.com.cn
电 话 : 010-62520290,62521724,62528991,62630301,62524940,62521921,82610344
(发行) 010-62613322-215(门市) 010-82675588-501,82675588-201(编辑部)
- 经 销 : 各地新华书店、软件连锁店
排 版 : 希望图书输出中心 马 君
文 本 印 刷 者 : 北京双青印刷厂
开 本 / 规 格 : 787 毫米×1092 毫米 1/16 21.25 印张 486 千字
版 次 / 印 次 : 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷
印 数 : 1-5000 册
本 版 号 : ISBN 7-900101-80-2
定 价 : 32.00 元

说明: 凡我社产品如有残缺,可执相关凭证与本社调换。

前 言

计算机辅助设计技术发展十分迅速，在建筑、机械、电子、土木等领域得到广泛应用。AutoCAD 2002 是 Autodesk 公司推出的 AutoCAD 最新版本，它在继承以前版本优点的基础上，新增了很多新功能，特别在多文档处理、设计中心、三维造型、网络支持、开发工具的完善等方面有了很大改进。

随着对 AutoCAD 运用的深入，越来越多的用户发现：利用 AutoCAD 提供的交互功能已经不能满足设计的要求，在解决某些专业问题时更是如此。AutoCAD 作为一种通用的制图系统，具有开放的体系结构，用户能根据需要来定制和扩展它的许多内容。定制是 AutoCAD 的最大的优点之一，用户无需正规培训，只需很少的程序设计知识就可以学会定制 AutoCAD。定制 AutoCAD 以满足用户需求是提高工作效率的最佳方式之一。此外在 AutoCAD 众多的开发工具中，AutoLISP 是使用最为简单的、应用最为广泛、用户最多的一种开发工具，它的显著特点是简单易学、可以直接使用，因此一直深受广大 AutoCAD 开发人员的喜爱。

本书主要讲述如何对 AutoCAD 2002 自定义和使用 AutoLISP 进行应用程序的开发。全书分为三篇，共 16 章，第一篇（从第 1 章到第 3 章）主要讲述基本的定制方法，包括对 AutoCAD 2002 的优化，定义模板和菜单系统等相关内容。第二篇（从第 4 章到第 7 章）讲述较为深层的定制内容，包括线型和填充文件的定义、语言支持及编程接口等内容。第三篇（从第 8 章到第 16 章）讲述 AutoLISP 和 Visual LISP 的相关内容，包括 AutoCAD 的语言开发环境，如何使用 AutoLISP 做程序开发，如何维护 Visual LISP 应用程序以及如何设计和管理对话框。

本书主要以结合实例的方式，由浅入深地进行系统的阐述，内容丰富而全面，是 AutoCAD 定制与开发入门读者参考学习的好工具。

本书向读者提供大量的典型实例。通过本书的学习，相信读者的开发水平能有较大程度的提高。

本书由唐亮、张帆和卫朝富执笔编写。胡明、王华杰、石磊、张维、吴轶秦、韩璐、王镭、张华东、李晓、范之誉、王宏、李琦、王瑾、吴君华、付鑫育、李龙、钱少伟、刘荣强、李伟光、朱峰、许大中、魏勇、萧玉、丁桦、李林、邵华刚、朱莉、肖育新、戴礼华、钟行兆、马军、李志盛、柳欢兵、关政、徐方方、钟华、王昊平、郭彩萍、程相军、李志远、肖帆、祝文君、叶才智、邵冰等同志在整理材料方面给予了作者很大的帮助。在此，作者向他们表示深深的谢意。

本书限于作者水平，难免在内容选材和叙述上有不当之处。竭诚欢迎广大读者对本书提出批评和建议。

编 者

2002 年 5 月

目 录

第一篇 初级定制

第 1 章 定制基础	2
1.1 概述	3
1.2 基本需求	4
AutoCAD 环境	4
1.3 系统参数文件	8
1.3.1 定义外部命令	8
1.3.2 命令别名	10
1.4 自定义联机文档	11
1.4.1 Windows 帮助文件	11
1.4.2 HTML 文件	13
1.5 AutoCAD 的最优化	15
1.5.1 使用命令行开关	15
1.5.2 使用环境变量	17
1.5.3 系统故障后的恢复	18
1.6 小结	20
1.7 思考与练习	20
第 2 章 定制图形模板	21
2.1 建立图形模板	22
2.2 标准的图形模板	22
2.2.1 使用对话框	24
2.2.2 使用 Units 命令	24
2.3 根据打印尺寸和绘图比例定制图形	25
2.4 小结	28
2.5 思考与练习	28
第 3 章 定制菜单系统	30
3.1 使用菜单文件	31
3.1.1 菜单文件类型	31
3.1.2 位图资源 DLL	32
3.1.3 加载菜单文件	32
3.1.4 基本和局部菜单	33
3.2 菜单文件结构	34
3.3 菜单项语法	35
3.3.1 名称标记	35
3.3.2 标签	35

3.3.3 菜单宏	36
3.4 菜单组	44
3.5 按钮和辅助菜单	44
3.5.1 创建按钮和辅助菜单	44
3.5.2 实例说明: 按钮和辅助 菜单的交换	45
3.5.3 反斜杠的特殊用法	47
3.6 下拉菜单和快捷菜单	47
3.6.1 实例说明: 下拉菜单的创建	47
3.6.2 创建快捷菜单	49
3.6.3 下拉菜单栏标题	49
3.6.4 级联子菜单	50
3.6.5 菜单项标签的显示控制	50
3.6.6 引用下拉菜单和快捷菜单	52
3.6.7 下拉菜单的交换和插入	53
3.7 工具栏	55
3.7.1 创建工具栏	55
3.7.2 引用工具栏	57
3.8 图像控件菜单	57
3.8.1 图像控件菜单项	57
3.8.2 显示图像控件菜单	58
3.8.3 实例说明: 样板图像控件菜单	58
3.9 实例说明: 面向菜单的帮助	60
3.10 加速键	60
3.11 小结	61
3.12 思考与练习	62

第二篇 中级定制

第 4 章 定制线型和填充文件	64
4.1 定义线型文件	65
4.1.1 定义线型	65
4.1.2 简单线型	65
4.1.3 复杂线型	66
4.1.4 加载线型	69
4.2 创建填充图案	70
4.2.1 AutoCAD 图案文件 acad.pat	70

4.2.2	如何构造填充图案.....	70
4.2.3	虚线图案	71
4.3	实例: 奇怪的杯.....	73
4.4	小结.....	74
4.5	思考与练习.....	74
第5章	形、字体和 PostScript 支持	76
5.1	使用形文件.....	77
5.2	编译形或字体文件.....	77
5.3	创建形定义文件.....	78
5.3.1	形说明	78
5.3.2	字体说明	84
5.3.3	大字体说明.....	85
5.3.4	Unicode 字体说明	88
5.3.5	高级形定义技术.....	89
5.4	PostScript 支持	90
5.4.1	AutoCAD PostScript 的 支持文件	90
5.4.2	高级 PostScript 处理	94
5.4.3	输入 PostScript 图像	95
5.5	小结.....	96
5.6	思考与练习.....	96
第6章	DIESEL 表达式语言的应用	97
6.1	状态栏配置.....	98
6.1.1	MODEMACRO 变量	98
6.1.2	定义 MODEMACRO	99
6.1.3	用 AutoLISP 设置 MODEMACRO 定义	100
6.2	DIESEL 表达式在菜单中的应用.....	102
6.3	DIESEL 表达式在 AutoLISP 中的应用	104
6.4	错误信息.....	105
6.5	小结.....	105
6.6	思考与练习.....	105
第7章	编程接口	106
7.1	命令脚本.....	107
7.1.1	加载 AutoCAD 时调用脚本.....	107
7.1.2	创建幻灯片演示.....	110
7.2	ActiveX Automation	111
7.2.1	从命令行中启动 Automation	

应用程序.....	111	
7.2.2	实例: 从菜单中启动 应用程序.....	111
7.3	AutoCAD VBA	113
7.3.1	用 AutoCAD VBA 进行开发.....	114
7.3.2	使用 AutoCAD VBA 应用程序.....	114
7.3.3	自动加载和执行 VBA 工程	115
7.4	ObjectARX.....	116
7.4.1	使用 ObjectARX 应用程序..	116
7.4.2	自动加载 ObjectARX 应用程序.....	117
7.5	小结.....	117
7.6	思考与练习.....	117

第三篇 AutoCAD 的二次开发

第8章	AutoCAD 的语言开发环境	120
8.1	语言开发环境概述	121
8.2	AutoLISP 语言.....	121
8.2.1	AutoLISP 程序的结构特点...	122
8.2.2	AutoLISP 的内存分配.....	123
8.2.3	AutoLISP 程序的执行过程.....	124
8.3	Visual LISP	125
8.4	ObjectARX 语言.....	126
8.5	小结.....	127
8.6	思考与练习.....	127
第9章	AutoLISP 基础	128
9.1	AutoLISP 的表达式和数据类型.....	129
9.1.1	AutoLISP 的表达式.....	129
9.1.2	AutoLISP 的数据类型.....	130
9.2	加载和运行 AutoLISP 程序.....	136
9.3	AutoLISP 变量.....	137
9.3.1	显示变量的值.....	138
9.3.2	编号处理.....	138
9.3.3	字符串的处理.....	139
9.4	小结.....	140

9.5 思考与练习.....	140	10.9 其它函数.....	171
第 10 章 AutoLISP 语言的相关函数.....	141	10.9.1 command 函数.....	171
10.1 AutoLISP 的赋值与运算函数.....	142	10.9.2 menucmd 函数.....	172
10.1.1 赋值函数.....	142	10.9.3 getenv 函数.....	172
10.1.2 数学运算.....	143	10.9.4 setvar 函数.....	172
10.1.3 三角函数.....	145	10.9.5 osnap 函数.....	172
10.1.4 逻辑运算.....	146	10.9.6 type 函数.....	172
10.2 程序分支与循环函数.....	150	10.9.7 redraw 函数.....	173
10.2.1 条件函数.....	151	10.9.8 *error* 函数.....	173
10.2.2 分支函数.....	151	10.10 小结.....	174
10.2.3 重复函数.....	151	10.11 思考与练习.....	174
10.2.4 循环函数.....	151	第 11 章 AutoLISP 对 AutoCAD 对象的控制	176
10.2.5 求多个表达式值的函数..	152	11.1 选择集.....	177
10.3 字符串与类型转换函数.....	152	11.2 选择集函数.....	178
10.3.1 字符串函数.....	152	11.2.1 创建选择集.....	178
10.3.2 类型转换函数.....	153	11.2.2 选择集过滤表.....	179
10.4 表处理函数.....	154	11.2.3 过滤器表中的通配模式..	180
10.4.1 表元素提取函数.....	154	11.2.4 过滤扩展数据.....	180
10.4.2 表的测量、连接与 替换函数.....	155	11.2.5 过滤器测试的逻辑分组..	181
10.4.3 表元素搜索函数.....	156	11.2.6 关系测试.....	182
10.5 基本输出函数.....	158	11.2.7 选择集操作.....	183
10.5.1 消息的显示.....	158	11.2.8 在 AutoLISP 和 ObjectARX 应用程序之间传递选择集..	184
10.5.2 AutoLISP 的控制字符....	159	11.3 对象的处理.....	184
10.6 AutoLISP 自定义函数.....	160	11.3.1 实体名称函数.....	184
10.6.1 defun 函数.....	160	11.3.2 实体数据函数.....	189
10.6.2 用 defun 函数定义 AutoCAD 的新函数.....	161	11.3.3 实体数据函数和图形屏幕..	195
10.6.3 lambda 函数.....	161	11.3.4 老式多段线和优化多段线..	196
10.6.4 C: 函数.....	162	11.3.5 处理非图形对象.....	196
10.6.5 添加命令.....	162	11.4 扩展数据 xdata.....	197
10.7 交互数据输入函数和相关的 计算函数.....	163	11.4.1 组织扩展数据.....	198
10.7.1 getxxx 函数.....	163	11.4.2 注册应用.....	200
10.7.2 表求值函数.....	167	11.4.3 获取扩展数据.....	200
10.8 与文件有关的函数.....	169	11.4.4 在实体上附着扩展数据....	201
10.8.1 AutoLISP 应用程序 加载函数.....	169	11.4.5 管理扩展数据使用的内存..	202
10.8.2 程序函数.....	169	11.4.6 扩展数据中的句柄.....	202
		11.5 访问符号表和词典.....	203
		11.5.1 符号表.....	203
		11.5.2 词典条目.....	204

11.6	小结.....	205	13.2.1	设置断点中断程序执行.....	236
11.7	思考与练习.....	205	13.2.2	单步调试程序.....	237
第 12 章	使用 Visual LISP 开发程序	206	13.2.3	监视表达式的求值结果.....	238
12.1	控制台的使用.....	207	13.2.4	程序的继续执行 和自动执行.....	240
12.1.1	理解控制台的工作.....	207	13.3	使用 Visual LISP 调试功能.....	240
12.1.2	对多个图形使用 控制台窗口.....	209	13.3.1	开始调试任务.....	241
12.1.3	使用控制台快捷菜单.....	210	13.3.2	理解中断循环.....	241
12.1.4	记录控制台窗口动作.....	210	13.3.3	使用断点.....	241
12.2	文本编辑器的使用.....	211	13.4	使用 Visual LISP 数据查看工具 ...	243
12.2.1	编辑文件.....	211	13.4.1	使用“Watch”窗口.....	244
12.2.2	使用文本编辑器快捷菜单.....	212	13.4.2	理解“Trace Stack”窗口 ...	245
12.2.3	使用键盘快捷键.....	212	13.4.3	查看 AutoCAD 图形实体 ...	249
12.2.4	移动和复制文本.....	214	13.5	小结.....	252
12.2.5	查找文本.....	215	13.6	思考与练习.....	252
12.2.6	文本书签.....	217	第 14 章	编译与维护 Visual LISP 应用程序 253	
12.3	使用控制台和编辑器的 编码辅助工具.....	218	14.1	程序的编译和连接.....	254
12.3.1	Visual LISP 代码的 语法着色.....	218	14.1.1	使用编译器.....	254
12.3.2	自动匹配的使用.....	219	14.1.2	通过文件编译程序.....	254
12.3.3	Visual LISP 的完词功能.....	221	14.1.3	编译实例学习.....	256
12.4	用 Visual LISP 设置代码格式.....	224	14.1.4	加载和运行已编译程序.....	257
12.4.1	Visual LISP 的代码格式.....	224	14.1.5	链接函数调用.....	258
12.4.2	应用格式设置选项.....	227	14.2	创建应用程序模块.....	258
12.4.3	应用 Visual LISP 注释样式.....	228	14.2.1	创建新的应用程序.....	258
12.4.4	代码格式编排程序 所受的限制.....	229	14.2.2	修改应用程序选项.....	263
12.5	检查语法错误.....	229	14.2.3	重新编译应用程序.....	264
12.5.1	括号匹配检测.....	229	14.2.4	更新应用程序.....	264
12.5.2	利用代码着色检测 语法错误.....	231	14.3	管理多个 LISP 文件.....	264
12.5.3	使用检查命令检查 语法错误.....	231	14.3.1	理解 Visual LISP 工程.....	265
12.6	小结.....	232	14.3.2	Visual LISP 的文件类型 ...	265
12.7	思考与练习.....	232	14.4	定义工程.....	266
第 13 章	应用程序的调试	234	14.4.1	指定工程属性.....	266
13.1	Visual LISP 调试功能.....	235	14.4.2	利用“Project”窗口 操作工程文件.....	269
13.2	通过实例学习程序调试.....	235	14.5	使用已存在的工程.....	270
			14.5.1	打开工程.....	270
			14.5.2	在工程源文件中 查找字符串.....	271
			14.5.3	在 Visual LISP 应用	

程序中包含工程.....	272	15.5.4 单词字母的大写.....	291
14.6 优化应用的代码.....	273	15.5.5 对话框的嵌套和隐藏.....	292
14.6.1 定义编译选项.....	273	15.5.6 键盘输入的处理.....	292
14.6.2 选择编译模式.....	274	15.5.7 错误处理.....	292
14.6.3 选择链接模式.....	275	15.6 实例：对话框的设计.....	293
14.6.4 理解安全优化.....	275	15.7 小结.....	296
14.7 小结.....	277	15.8 思考与练习.....	296
14.8 思考与练习.....	277	第 16 章 管理对话框	297
第 15 章 设计对话框	278	16.1 AutoLISP 程序控制对话框.....	298
15.1 对话框部件.....	279	16.2 动作表达式与回调.....	300
15.2 用 DCL 定义对话框.....	280	16.2.1 动作表达式.....	300
15.2.1 base.dcl 和 acad.dcl 文件.....	280	16.2.2 回调原因.....	301
15.2.2 引用 DCL 文件.....	280	16.2.3 缺省值与 DCL 动作.....	301
15.2.3 DCL 的语法.....	281	16.3 处理控件.....	302
15.3 用 VisualLISP 显示对话框.....	283	16.3.1 初始化控件模式和值.....	302
15.3.1 预览错误处理.....	285	16.3.2 在回调时改变控件 模式和值.....	302
15.3.2 对 DCL 文件的语义核查..	285	16.3.3 处理单选控件组.....	303
15.4 调整对话框的布局.....	285	16.3.4 处理滑块.....	304
15.4.1 在控件组中分配控件.....	286	16.3.5 处理编辑框.....	304
15.4.2 调整控件间距.....	287	16.4 嵌套和隐藏对话框.....	305
15.4.3 调整右端和底部的空间..	287	16.5 列表框和弹出式列表.....	308
15.4.4 调整加框行和 列周围的空间.....	288	16.5.1 列表操作.....	308
15.4.5 自定义退出按钮文本.....	288	16.5.2 处理列表元素.....	309
15.5 设计指导.....	290	16.5.3 特定应用数据.....	310
15.5.1 美观和功效.....	290	16.6 DCL 错误处理.....	310
15.5.2 用户控制.....	290	16.7 实例：用 AutoLISP 管理对话框..	311
15.5.3 容错处理.....	291	16.8 小结.....	328

第一篇

初级定制



第 1 章 定制基础

第 2 章 定制图形模板

第 3 章 定制菜单系统

本篇讲述对 AutoCAD 进行初级定制的知识与方法，主要包括定制的基本概念，以及基本的 AutoCAD 自定义功能。初次接触定制功能的读者可以通过阅读本篇内容，掌握如何在 AutoCAD 中输入操作系统命令，创建常用命令的别名，优化 AutoCAD 系统以及定义模板和菜单系统。

通过本篇的学习，用户能够根据需要创建图形模板，定义标准的图形模板以及根据视窗和图纸空间来定制图形，还可以自定义菜单，或者向菜单中添加所需的命令和宏，以达到优化工作环境的目的。

第1章

定制基础



- 概述
- 基本需求
- 系统参数文件
- 自定义联机文档
- AutoCAD 2002 的最优化





本章讲述基本的 AutoCAD 自定义功能。初次接触定制功能的读者可以通过学习本章，掌握如何在 AutoCAD 中输入操作系统命令，如何创建常用命令的别名，如何为自己的应用和程序开发联机帮助，以及如何优化 AutoCAD 系统。

1.1 概 述

AutoCAD 是开放式结构的通用绘图系统，用户可以根据需要扩展和调整 AutoCAD 的功能。

一般可以通过以下方法定制 AutoCAD:

- 在 AutoCAD 中运行外部程序。例如，只要在程序参数文件 acad.pgp 中加入适当的外部命令，就可以在 AutoCAD 内部复制磁盘或删除文件。对于经常使用的 AutoCAD 命令，可以创建它们的简单缩写，即别名。例如，输入 b 就可以调用 BHATCH 命令。别名也是在程序参数文件 acad.pgp 中定义的。
- 为应用程序和内部过程制作帮助系统。使用 AutoLISP、VBA 和 ObjectARX 程序，可以从菜单、工具栏获得帮助。通过创建帮助文件，可以自定义 AutoCAD 命令的帮助，也可以为自己的应用程序定义帮助。
- 创建自定义点划线型、填充图案、形和文字样式。用户可以创建符合企业标准和工作需要的填充图案、形和字体。
- 创建自定义菜单和工具栏，以修改 AutoCAD 用户界面。菜单在很大程度上决定了 AutoCAD 界面的外观。将最常用的命令放在菜单上最容易选取的位置，并将命令分组，合并到单个菜单项中，工作效率会显著提高。通过菜单自定义功能，用户可以根据实际需要调整 AutoCAD 界面。菜单文件定义了选择定点设备按钮后执行的操作。同样，下拉菜单、屏幕、数字化仪和图像控件菜单的功能和外观，以及工具栏和加速键，均在菜单文件定义。
- 自定义状态行。可以使用 DIESEL 字符串表达式语言和 MODEMACRO 系统变量在状态行中显示各种信息，如日期和时间、系统变量设置或其他可以通过 AutoLISP 提取的信息。
- 创建自定义模板。在作图的过程中可以通过模板把普通的参数定义到网页向导，或是定义告示牌模板的内容。
- 编写脚本自动完成重复性任务。脚本是包含 AutoCAD 命令的 ASCII 文本文件，其运行方式类似于批处理文件。例如，如果需要以某种方式打印一组图形，则可以编写脚本来依次打开每个图形，打开或关闭其中不同的图层并发出 PLOT 命令，然后让它在夜间无人看管时自动运行。还可以用脚本和幻灯片创建展示会中那种自动连续播放的演示。幻灯片是不可编辑的 AutoCAD 绘图区域“快照”。在图片菜单和对话框中，幻灯片还用于概括图形信息。。
- 在命令提示或 AutoLISP 及 ObjectARX 程序中重定义或禁用选定的 AutoCAD 命令。例如，重新定义某些 AutoCAD 命令以提供辅助信息和提示；或者创建图形管理系统，在其中重新定义 QUIT 命令，在退出编辑任务之前将某些信息写入日志文件。



1.2 基本需求

大多数自定义功能都需要编辑或创建 ASCII 文本文件，因此，应该能熟练使用一种文本编辑器（例如 Microsoft Windows 记事本）或能将文件保存为 ASCII 格式的字处理器（例如 Microsoft Word），还应该注意在修改 AutoCAD 支持文件（例如菜单文件）之前进行备份。

AutoCAD 环境

AutoCAD 的缺省目录结构将程序和支持文件按逻辑关系分组，这种结构可按照需要修改。但有些应用程序需要在特定位置查找某些文件，进行修改时应避免与这些应用程序的要求冲突。如果未指定驱动器和目录，AutoCAD 只在库搜索路径中查找文件。

1. 库搜索路径

AutoCAD 按照库路径中指定的次序搜索支持文件，如下所示：

- 当前目录（这通常由快捷图标的“开始”设置决定）。
- 包含当前图形文件的目录。
- OPTIONS 中指定的搜索路径中所列出的目录。
- 包含 AutoCAD 程序文件的目录。

上述各目录可能相同，这取决于当前环境。

如果某个文件不在搜索路径中，则必须指定其路径名和文件名，AutoCAD 才能找到该文件。例如，如果要将在不在库搜索路径中的图形 part5.dwg 插入到当前图形中，则必须如下指定其完整的路径名：

```
command: insert
Enter block name or [?]: /files2/olddwgs/part5
```

如果该图形确实位于指定的位置，AutoCAD 将像通常一样提示完成 INSERT 命令。

2. 目录结构

文件操作和操作系统目录结构可以通过一定的方法来简化。AutoCAD 使用树状结构的目录和子目录。

建议读者将自己定义的文件（例如 AutoLISP 应用程序和菜单文件）与 AutoCAD 的程序和支持文件分开存放，这样更容易查找可能的冲突，并可以单独升级某个应用程序，而不影响其他应用程序。

下面将用图 1.1 所示的目录结构样例来说明组织目录和文件的方法。

可以为自定义的 AutoLISP 和 Visual Basic 应用程序、菜单文件和其他第三方应用程序创建一个目录（例如样例中的/AcadApps）。如果需要为相互独立的工作文件维护多个图形文件目录，可以创建一个主目录（例如/AcadJobs），然后在其下为每项工作创建一个子目录。

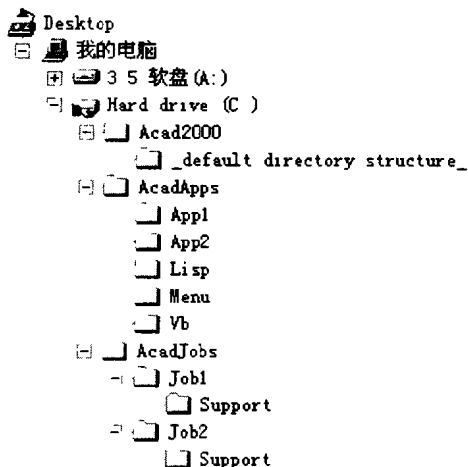


图 1.1 AutoCAD 树状结构样例

3. 多种配置

AutoCAD 为使用定点设备和打印设备进行配置时,所提供的配置信息记录在配置文件中。配置文件 acad2002.cfg 缺省放在包含 AutoCAD 程序文件的目录中,但用户可以指定替换的路径和/或文件名。

用户一般只需要一个配置,但有时也会需要多个配置。例如,用户的大多数工作使用鼠标即可完成,但偶尔也需要使用数字化仪。在这种情况下,应该将系统设置为处理多个配置,而不是在每次更换设备时重新进行配置。

许多 AutoCAD 系统变量的值和“选项”对话框中的配置选项都保存在配置文件中。如果需要使用这些变量和操作参数的不同设置,可将它们相应的值保存在不同的配置文件中。

必须将 AutoCAD 设置为使用不同的配置文件,才能利用多配置的优点。可用/c 命令行开关指定替换的配置文件。

4. 多个图形文件目录

维护多个图形文件目录不仅是一种约定,而且经常是工作中所必须的。将图形和其他相关文件保存在独立的目录中可以简化基本的文件维护工作。

可将自己的图形子目录都放在/AcadJobs 目录中。图形子目录仍然可以包含其他子目录,读者可在这些子目录中保存与特定的图形类型或工作相关的支持文件。/AcadJobs/Job1/Support 可包含/AcadJobs/Job1 中图形文件专用的块和 AutoLISP 文件。在支持路径中指定 Support 不带路径前缀),将把当前目录的 Support 子目录添加到支持路径中。注意:用“选项”对话框指定目录时,AutoCAD 将使用绝对路径。要使用上述的相对路径命名惯例,必须在命令行中用/s 开关指定支持路径。

在启动 AutoCAD 时,如果所需的图形文件目录是当前目录,则该目录中的所有文件和子目录都可以直接访问。用户可以为每项工作创建程序图标或“开始”菜单项,并指定正确的工作目录。



可使用批处理程序来代替图标或菜单。用批处理程序可自动创建新的工作目录。下列批处理程序先验证是否存在指定的目录，再将该目录设为当前目录，然后运行 AutoCAD。

```
@echo off
C:
if exist \AcadJobs\Jobs\%1 goto RUNACAD
echo.
echo *** Creating \AcadJobs\Jobs\%1
echo *** Press Ctrl+C to cancel.
echo.
pause
mkdir \AcadJobs\Jobs\%1
:RUNACAD
cd \AcadJobs\Jobs\%1
start C:\AutoCAD\acad.exe
```

用记事本或其他 ASCII 文本编辑器将此批处理程序保存为文件 acad.bat。用户须要按实际情况更改目录和驱动器的名称。将此文件放在系统搜索路径指定的目录（例如 C:\winnt）中，然后就可以用“开始”菜单上的“运行”命令，或者在资源管理器中双击，来运行此批处理程序。如果将该文件保存为 acad.bat，那么在使用时需要按照如下语法：

acad jobname

其中，jobname 是要置为当前的工作目录的名称。

5. 命令搜索过程

AutoCAD 要对所输入命令进行一系列有效性检查。一个命令可以是内部命令、系统变量、acad.pgp 文件中定义的外部命令或命令别名（在下一节中描述），或是用户定义的 AutoLISP 命令，也可以由 ObjectARX 应用程序或设备驱动程序命令来定义。另外，命令可以在命令行中键入或者从适当的菜单中选择，也可以从脚本文件中或者由 AutoLISP 或 ObjectARX 应用程序来输入。

下表描述 AutoCAD 验证命令名称的搜索次序。

表 1.1 AutoCAD 的命令搜索过程

- | | |
|---|---|
| 1 | 如果输入空响应（按 SPACEBAR 键或 ENTER 键），则 AutoCAD 调用上次执行的命令。HELP 是缺省命令 |
| 2 | AutoCAD 先在内置命令表中查找该命令名。如果该命令在内置命令表中，并且没有前缀句号 (.)，AutoCAD 将接着在未定义命令表中查找该命令。如果该命令未被定义，则继续搜索；否则将执行该命令，除非有其他原因使命令不能执行（例如，不允许透明地运行或在“透视”模式下运行命令） |
| 3 | AutoCAD 检查该命令名称是否由设备驱动程序定义，然后检查命令名是否由显示驱动程序定义 |



(续表)

- 4 AutoCAD 在程序参数文件(acad.pgp)定义的外部命令中查找该命令名称。如果该命令名称对应一个已经定义的外部命令, 则运行该命令, 并结束搜索
- 5 AutoCAD 在由 AutoLISP 或 ObjectARX 应用程序定义的命令列表中查找该命令名称。自动加载的命令在此时加载(关于自动加载命令的详细信息, 请参见自动加载命令)
- 6 AutoCAD 在系统变量列表中查找该命令名称。如果找到该名称, AutoCAD 将输入作为变量名, 执行 SETVAR 命令
- 7 如果该命令名称对应程序参数文件中定义的命令别名, AutoCAD 用该命令的全名在内置命令列表中重新搜索
- 8 如果前面所有步骤都失败了, 则搜索终止, 并显示非法命令的警告信息

6. 可自定义的支持文件

AutoCAD 用支持文件保存菜单定义、加载 AutoLISP 和 ObjectARX 应用程序以及描述文字字体等。许多支持文件是文本文件, 可以用文本编辑器进行修改。

下表列出可编辑的 AutoCAD 支持文件。这些支持文件按照扩展名的字母顺序列出。请记住在对这些文件进行修改之前先制作备份。

表 1.2 可自定义的支持文件

文件	说明
*.ahp	AutoCAD 帮助文件。相关的帮助索引文件的扩展名为.hdx
asi.ini	数据库连接的转换映射文件
*.dcl	AutoCAD 对话框控制语言(DCL)程序文件
*.lin	AutoCAD 线型定义文件
acad.lin	标准 AutoCAD 线型库文件
acadiso.lin	标准 AutoCAD ISO 线型库文件
*.lsp	AutoLISP 程序文件
acad.lsp	每次启动 AutoCAD 时加载的用户自定义 AutoLISP 例行程序
acaddoc.lsp	每次开始一个图形时加载的用户自定义 AutoLISP 例行程序
*.mnl	多线库文件
*.mnl	AutoCAD 菜单使用的 AutoLISP 例行程序。MNL 文件必须和它所支持的 MNU 文件同名
acad.mnl	标准 AutoCAD 菜单使用的 AutoLISP 例行程序
*.mns	AutoCAD 生成的菜单源文件。包含定义 AutoCAD 菜单的命令字符串和宏语法
acad.mns	标准 AutoCAD 菜单的源文件
*.mnu	AutoCAD 菜单源文件。包含定义 AutoCAD 菜单的命令字符串和宏语法
acad.mnu	标准 AutoCAD 菜单的源文件
*.pat	AutoCAD 填充图案定义文件
acad.pat	标准 AutoCAD 填充图案库文件



(续表)

文件	说明
acadiso.pat	标准 AutoCAD ISO 填充图案库文件
acad.pgp	AutoCAD 程序参数文件。包含外部命令和命令别名的定义
fontmap.ps	AutoCAD 字体映射文件。由 PSIN 使用，是 AutoCAD PostScript_ 解释器能识别的全部字体的目录（或字体映射）
acad.psf	AutoCAD PostScript 支持文件，PSOUT 和 PSFILL 命令的主要支持文件
acad.rx	列出启动 AutoCAD 时加载的 ObjectARX 应用程序
*.scr	AutoCAD 脚本文件。脚本文件包含一组 AutoCAD 命令，其运行方式与批处理文件相似
*.shp	AutoCAD 形/字体定义文件。编译后形/字体文件的扩展名为.shx
acad.unt	AutoCAD 单位定义文件。包含进行单位换算所需的数据

1.3 系统参数文件

AutoCAD 系统参数文件 acad.pgp 是 ASCII 文本文件，用于保存命令定义，可将此文件当作自定义 AutoCAD 命令的列表。当用户输入不常用的命令时，AutoCAD 在 acad.pgp 中查找该命令。此文件分成两个部分：第一部分定义外部命令，第二部分定义命令别名。此文件可以用分号 (;) 引入说明文字。

1.3.1 定义外部命令

在 AutoCAD 运行时，用户可调用其他程序或实用工具，例如：

- Windows 系统命令和实用工具，例如 start、type、dir 或 copy。
- 应用程序，例如文本编辑器或字处理器。
- 数据库管理程序、电子表格和通讯程序。
- 用户提供的程序，例如批处理文件或 Visual Basic 应用程序。

定义外部命令时，需要指定在 AutoCAD 命令提示中使用的命令名称以及传递给操作系统的可执行命令字符串。外部命令部分中的每行都包括以逗号分隔的五个字段，如下所示：

```
command,[executable],flags[,*]prompt[,return_code]
```

command: 在命令提示中输入的命令。如果此名称是 AutoCAD 内部命令名称，则该命令将被忽略。这个名称不区分大小写。

Executable: 输入命令名时传递给操作系统的固定字符串，可以是能在操作系统提示中执行的任何命令，可包含开关或参数。该字符串是否区分大小写由运行的应用程序决定。

Flags: 必要的位编码参数，按任意组合将这些整数值相加以得到所需的结果。

- 0 启动应用程序并等待结束。
- 1 不等应用程序结束。
- 2 最小化运行应用程序。