

北京科海培训中心

松岗

# AutoCAD R12 窗口设计技巧 与 ADS 程序设计参考

[台] 林龙震 编著

科培 改编

清华大学出版社

**AutoCAD R12 视窗设计技巧与 ADS 程式设计参考**

**林龙震 编著**

本书由台湾松岗电脑图书资料股份有限公司出版,1993。本书中文简体字版经松岗公司授权北京科海培训中心改编和清华大学出版社合作出版,1994。未经出版者书面允许,不得用任何手段复制或抄袭本书内容。

**(京)新登字 158 号**

**AutoCAD R12 窗口设计技巧与 ADS 程序设计参考**

[台] 林龙震 编著

科 培 改编

☆

清华大学出版社出版

北京 清华园

门头沟胶印厂印刷

新华书店总店科技发行所发行

☆

开本:787×1092 1/16 印张:22.75 字数:553千字

1994年3月第1版 1994年3月第1次印刷

印数:0001~5000册

ISBN 7-302-01492-2/TP·597

定价:30.00元

## 编者序

AutoCAD R12 DCL (AutoCAD R12 Dialog Control Language) 是新版 AutoCAD 所提供的对话框制作描述语言。由于这种窗口功能在目前的成套软件上非常流行,同时也是一种趋势,所以特将此书呈现给广大读者。

本书是《AutoCAD R12 使用手册》一书的进阶读本。因为对话框的制作牵涉到 AutoLISP 与 ADS 技术;且 AutoLISP 已风行多年,人尽皆知,所以本人将 AutoLISP 技术部分放入《AutoCAD R12 使用手册》一书中;而本书除了第一部分的对话框制作技术之外,也将 ADS 的部分编撰在本书的第二部分以让有兴趣的读者有更完整的资料可查。

全书的重点将以理论配合范例相结合来指导读者进入 AutoCAD 程序设计领域,得到最佳的学习效果,由浅入深地来设计所需的 AutoCAD 功能。希望能以最简短有效的叙述来提供您需求的所有资料。无论是 AutoCAD 哪一层次的用户本书均可适用。同时,祈望广大读者对本书的缺点加以指正,谢谢。

编者

# 目 录

## 第一部分 AUTOCAD 对话框设计

<b>第一章 DCL 概论</b> .....	(1)
1.1 引 言 .....	(1)
1.2 DCL 的作用 .....	(1)
1.2.1 设计对话框 .....	(1)
1.2.2 使应用程序支持对话框 .....	(1)
1.3 对话框简介 .....	(2)
<b>第二章 对话框的组成</b> .....	(5)
2.1 引 言 .....	(5)
2.2 对话框的组成 .....	(6)
<b>第三章 对话框的属性</b> .....	(14)
3.1 控件属性 .....	(14)
3.1.1 整数 .....	(14)
3.1.2 实数 .....	(14)
3.1.3 由引号括起来的字符串 .....	(14)
3.1.4 保留字 .....	(14)
3.2 用户自定义的属性 .....	(15)
3.3 预定义属性 .....	(15)
3.4 配置与设置尺寸大小的属性 .....	(18)
3.5 功能性的属性 .....	(19)
3.6 预定义控件的属性 .....	(20)
<b>第四章 对话框控制语言(DCL)</b> .....	(27)
4.1 引言 .....	(27)
4.2 DCL 中一个简单的对话框 .....	(27)
4.3 DCL 文件结构 .....	(29)
4.3.1 base.dcl 与 acad.dcl 文件 .....	(29)
4.3.2 用户定义的 DCL 文件 .....	(29)
4.4 DCL 语法 .....	(30)
4.4.1 控件定义 .....	(30)
4.4.2 控件引用 .....	(31)
4.4.3 属性与属性值 .....	(32)
4.4.4 注释 .....	(32)
4.4.5 DCL 错误处理 .....	(32)

4.5	DCL 的技巧 .....	(33)
4.5.1	在组中控制建立控件.....	(33)
4.5.2	填充插入点(衬垫 0) .....	(34)
4.5.3	填充单元(衬垫 1) .....	(34)
4.5.4	一稀疏组中介于控件间的不必要区域.....	(34)
4.5.5	右侧的不必要空间或一对话框的按钮.....	(35)
4.5.6	一横向或纵向对话框的区域.....	(36)
4.5.7	文本组 .....	(36)
4.5.8	对话框退出按钮.....	(37)
4.6	对话框设计.....	(40)
4.6.1	相关的参考文献.....	(40)
4.6.2	一般原则.....	(40)
4.7	预定义控件以及组的规范与设计指导.....	(43)
4.7.1	按钮.....	(43)
4.7.2	组.....	(44)
4.7.3	编辑框.....	(44)
4.7.4	图象按钮以及图象控件.....	(45)
4.7.5	列表框.....	(45)
4.7.6	下拉列表.....	(45)
4.7.7	互锁按钮以及互锁列与互锁行.....	(45)
4.7.8	滑动条.....	(45)
4.7.9	文本.....	(46)
4.7.10	乒乓开关 .....	(46)
<b>第五章</b>	<b>一般规定与建议 .....</b>	<b>(47)</b>
5.1	文本的大写.....	(47)
5.2	缩写.....	(47)
5.3	配置.....	(47)
5.4	对话框的大小及配置.....	(47)
5.5	取消功能的控件.....	(48)
5.6	平铺对话框.....	(48)
5.7	隐藏对话框 .....	(48)
5.8	提供缺省值.....	(49)
5.9	键盘输入.....	(49)
5.10	颜色 .....	(49)
5.11	使用标准控制 .....	(49)
5.12	错误处理 .....	(49)
5.13	警告对话框 .....	(49)
5.14	可移植的设计 .....	(50)
5.15	汉化问题 .....	(50)

<b>第六章 用 AutoLISP 和 ADS 管理对话框</b>	<b>(51)</b>
6.1 引言	(51)
6.2 处理一对话框的函数实例	(51)
6.3 函数调用顺序	(52)
6.4 一对话框起作用时不允许使用的函数	(53)
6.5 动作程序以及回调函数	(55)
6.6 AutoLISP:动作程序	(56)
6.7 ADS:回调函数	(57)
6.8 回调原因	(58)
6.9 缺省值与 DCL 动作	(59)
6.10 平铺对话框	(60)
6.11 隐藏对话框	(60)
6.12 ADS 定义和声明	(63)
6.12.1 对话框及控件的处理	(63)
6.12.2 回调函数定义	(63)
6.12.3 状态码	(64)
6.12.4 回调原因码	(64)
6.12.5 控件模式及列表函数码	(65)
6.12.6 存放属性字符串的缓冲区	(65)
6.13 处理控件	(65)
6.13.1 初始模式及其值	(65)
6.13.2 在回调时改变模式及其值	(66)
6.13.3 设置列表框及其下拉式列表	(67)
6.13.4 处理列表值	(69)
6.13.5 建立图象	(71)
6.13.6 图象按钮输入	(73)
6.13.7 处理互锁组	(74)
6.13.8 处理滑动条	(75)
6.13.9 处理编辑框	(76)
6.13.10 特定应用程序数据	(77)
6.14 函数简介	(78)
6.15 函数目录	(81)
6.16 块定义对话框对话框实例	(90)
<b>第七章 举例</b>	<b>(91)</b>
7.1 引言	(91)
7.2 用户应该准备什么?	(91)
7.3 入门	(91)
7.4 AutoCAD 的灵魂	(98)

7.5 检验结果 .....	(104)
7.6 加入 AutoCAD 中 .....	(106)

## 第二部分 ADS 程序设计参考

<b>第一章 概 论 .....</b>	<b>(109)</b>
1.1 引 言 .....	(109)
1.2 使用 ADS 设计程序的基本条件 .....	(109)
1.3 ADS 的基本概念 .....	(110)
1.3.1 高度兼容性 .....	(110)
1.3.2 辅助编译程序和操作系统 .....	(110)
1.4 ADS 的改进及提高 .....	(111)
1.5 针对标准外部应用程序的可编程接口 .....	(112)
1.6 可编程的对话框 .....	(112)
1.7 如何从 AutoLISP 中访问 ADS 应用程序 .....	(112)
1.7.1 使用 AutoCAD 时装入一个应用程序 .....	(112)
1.7.2 数据库搜索路径 .....	(113)
1.8 列出已装入的 ADS 应用程序 .....	(114)
1.9 运行 ADS 函数 .....	(114)
1.10 卸载 ADS 应用程序 .....	(114)
1.11 在 AutoCAD 初始化时装入应用程序 .....	(115)
1.12 ADS 文件及其内容 .....	(115)
1.12.1 ADS 目标数据库 .....	(115)
1.12.2 头文件 .....	(115)
1.13 ADS 数据库函数的一般特性 .....	(116)
1.14 ADS 函数调用与 AutoLISP 调用的比较 .....	(116)
1.15 LISP 和 C 中的参数序列 .....	(116)
1.16 内存考虑 .....	(117)
1.17 应用程序文件的组织 .....	(117)
1.18 内存管理 .....	(117)
1.19 函数返回值与函数结果 .....	(118)
1.20 编译和连接一个 ADS 应用程序 .....	(118)
<b>第二章 使用 ADS 数据库 .....</b>	<b>(120)</b>
2.1 引 言 .....	(120)
2.2 ADS 应用程序的结构 .....	(120)
2.3 使用 AutoLISP 建立联系 .....	(122)
2.4 ADS 应用程序的要求码和结果码 .....	(122)
2.4.1 AutoLISP 的要求码 .....	(122)
2.4.2 应用程序的结果码 .....	(124)

2.5	外部函数 .....	(124)
2.5.1	定义外部函数 .....	(124)
2.5.2	外部函数求值 .....	(125)
2.6	错误处理 .....	(126)
2.7	应用程序之间的联系 .....	(128)
2.8	在调用函数时处理错误 .....	(131)
2.9	处理外部应用程序 .....	(131)
2.10	ADS 中定义的变量、类型和值 .....	(132)
2.10.1	一般类型和定义 .....	(132)
2.10.2	仅用于扩展图元数据中的类型 .....	(137)
2.10.3	有用的值 .....	(137)
2.11	结果缓冲区和类型码 .....	(138)
2.11.1	ADS 数据库定义的结果类型码 .....	(139)
2.11.2	DXF 组码 .....	(139)
2.12	ADS 应用程序的要求码和结果码 .....	(141)
2.12.1	要求码 .....	(141)
2.12.2	结果码 .....	(142)
2.12.3	ADS 数据库函数的结果码 .....	(142)
2.12.4	用户输入的控制位码 .....	(143)
2.13	全局变量 .....	(143)
2.13.1	应用程序的文件名 .....	(143)
2.14	序列及其他动态分配数据 .....	(144)
2.15	管理结果缓冲区 .....	(146)
2.15.1	建立序列 .....	(146)
2.15.2	AutoLISP 序列 .....	(147)
2.15.3	带 DXF 程序码的图元序列 .....	(149)
2.15.4	指令和函数运行序列 .....	(150)
<b>第三章</b>	<b>一般的公共函数 .....</b>	<b>(151)</b>
3.1	引言 .....	(151)
3.2	AutoCAD 询问和指令 .....	(151)
3.2.1	一般存取 .....	(151)
3.2.2	等待用户输入 .....	(153)
3.2.3	传送选取点给 AutoCAD 指令 .....	(153)
3.3	系统变量 .....	(153)
3.4	AutoLISP 符号 .....	(154)
3.5	文件搜索 .....	(155)
3.6	图形锁定 .....	(156)
3.7	视口描述器 .....	(157)
3.8	几何图形公共程序 .....	(158)

3.9	文本对话框公共函数 .....	(158)
3.10	获取用户输入 .....	(161)
3.10.1	用户输入函数 ads_getxxx( ) .....	(161)
3.11	控制用户输入函数的条件 .....	(163)
3.11.1	用户输入函数的输入选项 .....	(163)
3.11.2	指定关键字 .....	(165)
3.12	拖动选择集 .....	(165)
3.12.1	用户中断 .....	(166)
3.13	AutoLISP 函数的返回值 .....	(167)
3.14	转换 .....	(168)
3.14.1	字符串转换 .....	(168)
3.14.2	真实世界单位 .....	(170)
3.15	字符类型处理 .....	(171)
3.16	坐标系统变换 .....	(171)
3.17	显示控制 .....	(173)
3.17.1	交互输出 .....	(173)
3.17.2	图形和文本屏幕的控制 .....	(174)
3.17.3	低级图形和用户输入控制 .....	(175)
3.18	校正数字化仪 .....	(175)
3.19	通配符模式 .....	(177)
<b>第四章</b>	<b>选择集、图元及符号表函数 .....</b>	<b>(178)</b>
4.1	选择集和图元名 .....	(178)
4.2	处理选择集 .....	(178)
4.2.1	选择集过滤序列 .....	(180)
4.2.2	过滤序列中的通配符模式 .....	(182)
4.2.3	过滤扩展的图元数据 .....	(182)
4.2.4	相关测试 .....	(183)
4.2.5	条件过滤 .....	(184)
4.3	操作选择集 .....	(185)
4.4	转换选择集 .....	(186)
4.5	图元名称和数据函数 .....	(189)
4.5.1	图元名称函数 .....	(189)
4.5.2	图元处理及图元的用法 .....	(190)
4.6	图元前后关系和坐标转换数据 .....	(191)
4.6.1	坐标转移 .....	(191)
4.6.2	相关数据 .....	(194)
4.6.3	合成的实例 .....	(195)
4.7	图元数据函数 .....	(196)
4.8	建立复杂的图元 .....	(200)

4.9	匿名集成模块	(202)
4.10	图元数据函数和图形屏幕	(204)
4.11	扩展图元数据的注意事项	(205)
4.12	扩展图元数据的组织	(206)
4.13	登记一个应用程序	(207)
4.14	获取扩展的图元数据	(208)
4.15	管理扩展图元数据占用的内存	(209)
4.16	处理扩展图元数据	(210)
4.17	访问符号表	(210)
<b>第五章 ADS 数据库参考 (213)</b>		
5.1	引言	(213)
5.2	概述	(213)
5.2.1	建立与 AutoLISP 的接口类	(213)
5.2.2	外部函数处理类	(213)
5.2.3	外部应用程序处理类	(213)
5.2.4	错误处理类	(214)
5.2.5	内存管理类	(214)
5.2.6	AutoCAD 变量与指令类	(214)
5.2.7	几何公共函数类	(215)
5.2.8	用户输入类	(215)
5.2.9	返回外部函数值类	(216)
5.2.10	转换类	(217)
5.2.11	字符类型处理类	(217)
5.2.12	坐标系统转换类	(218)
5.2.13	数字化仪校正类	(218)
5.2.14	显示控制类	(218)
5.2.15	低级图形类	(219)
5.2.16	通配符类	(219)
5.2.17	选择集类	(219)
5.2.18	图元处理类	(220)
5.2.19	扩展的图元数据类	(220)
5.2.20	符号表类	(221)
5.3	可编程的对话框函数	(221)
5.3.1	打开及关闭 DCL 文件类	(221)
5.3.2	打开及关闭对话框类	(221)
5.3.3	初始化显示或响应函数类	(222)
5.3.4	处理控件以及属性类	(222)
5.3.5	设置列表框以及自动下拉列表类	(222)
5.3.6	建立图象类	(222)

5.3.7	特定应用程序数据类 .....	(223)
5.4	外部定义的 AutoCAD 函数类 .....	(223)
5.5	设计与 AutoCAD 指令交互作用的接口 .....	(223)
5.6	ADS 数据库函数清单 .....	(224)
5.7	外部定义的 AutoCAD 函数 .....	(290)
5.8	设计与 AutoCAD 指令交互的接口 .....	(293)
<b>第六章</b>	<b>程序设计技巧 .....</b>	<b>(298)</b>
6.1	引言 .....	(298)
6.2	程序实例所使用的技巧 .....	(298)
6.3	实例程序 fact.c .....	(299)
6.3.1	使用应用程序 .....	(299)
6.3.2	技巧 .....	(300)
6.4	实例程序 ads_perr.c .....	(300)
6.5	实例程序 arbmat.c .....	(301)
6.6	实例程序 sld2ps .....	(301)
6.7	实例程序 ptext.c .....	(302)
6.8	实例程序 colext.c .....	(303)
6.8.1	红-绿-蓝(RGB)系统 .....	(303)
6.8.2	淡青-紫红-黄色(CMY)系统 .....	(303)
6.8.3	YIQ 系统 .....	(303)
6.8.4	色调-彩度-明暗度系统(HueSaturation-Value,HSV) .....	(303)
6.8.5	色调-颜色淡薄度-彩度系统(Hue-Lightness-Staturation ,HLS .....	(303)
6.8.6	温度颜色 .....	(304)
6.8.7	颜色命名系统(CNS) .....	(304)
6.8.8	使用应用程序 .....	(306)
6.8.9	转换成 AutoCAD 颜色 .....	(306)
6.8.10	转换成 RGB .....	(307)
6.8.11	AutoCAD 和 RBG 颜色转换 .....	(308)
6.8.12	技巧 .....	(308)
6.9	实例程序 mountain.c .....	(309)
6.9.1	使用应用程序 .....	(309)
6.9.2	技巧 .....	(310)
6.10	实例程序 gravity .....	(310)
6.10.1	快速漫游星际 .....	(311)
6.10.2	使用程序 .....	(311)
6.10.3	单位 .....	(311)
6.10.4	质量图元 .....	(311)
6.10.5	模拟程序指令 .....	(312)
6.10.6	技巧 .....	(313)

6.11 实例程序 magnets.c .....	(317)
6.11.1 magnets 中的指令 .....	(318)
6.12 避免错误.....	(319)
6.12.1 传递指针以取代数据结构.....	(319)
6.12.2 释放选择集时出现的错误.....	(320)
6.12.3 用 Long(长型)整数为选择集编码时出现的错误 .....	(320)
6.12.4 释放结果缓冲区时出现的错误.....	(321)
6.12.5 以共享模式打开文件.....	(321)
<b>附录 A AutoLISP 与 ADS 函数对照表 .....</b>	<b>(322)</b>
<b>附录 B DXF 群组码 .....</b>	<b>(329)</b>
<b>附录 C 错误码.....</b>	<b>(341)</b>
<b>附录 D ASCII 码 .....</b>	<b>(346)</b>

# 第一部分 AUTOCAD 对话框设计

## 第一章 DCL 概论

### 1.1 引言

在过去若干年中,Macintosh 窗口画面曾经风靡了美洲大陆,而在中国,也有许多人喜爱在计算机上使用这种画面,这主要因为它操作简便,用户界面友好的缘故。不久前,Microsoft 公司推出的 Windows 也采用了这种用户友好的窗口设计,以方便非专业人员操作。AutoCAD 也是如此,第 9 版就开始提供所谓下拉式菜单及图标菜单的操作方式,以改善其界面,但这毕竟与窗口的含义有所不同。现在第 12 版又提出了对话框语言,以使用户自行设计对话框功能,这就是本书要介绍的 DCL。

### 1.2 DCL 的作用

#### 1.2.1 设计对话框

对话框是用 Dialog Control Language (DCL,对话控制语言)编写的 ASCII 文件(文本文件)定义的。对话框中的 DCL 描述将定义对话框以什么方式出现以及它包含的内容,例如按钮、列表、文本等。在第四章将说明有关 DCL 语法。

对话框的设计是有限制的,它的大小及各部分的布局(layout)是根据一个最小的位置信息来自动完成的。正如由字处理软件来格式化一段文本一样,布局也同样基于此一致性的原则。用户不必精确指定每个部分的大小或(X,Y)位置,所以,如同使用字处理软件一样,该自动布局可帮助用户处理设计上的不协调之处。

#### 1.2.2 使应用程序支持对话框

在某些范围内,对话框的一部分可定义它的行为。例如,可以定义能“按下”的按钮,显示出一个列表框以使用户作一些选择等。但是,一个对话框的使用方式及其行为,实际上完全由它所属的应用程序所决定。AutoCAD Development System (ADS)则提供了处理对话框的功能(包括显示对话框、响应用户的选择等)。这个可编程的对话框(Programmable Dialog Box,PDB)功能或结构将在第六章说明。

**注意** 对话框严格地限制于“交互式”这种使用方式。一个脚本文件(Script)也可启动一个对话框,但无法对它加以控制,也无法在它被打开后提供输入。对于 AutoLISP (com-

mand)函数及 ADS ads\_command ( )和 ads\_cmd ( )函数,情况亦如此。

AutoCAD 所支持的内部对话框以及用户定义的对话框,将视用户使用的计算机而有所不同。一个单独的 DCL 描述以及应用程序可在所有不同类型的计算机上定义一对话框和它的功能。从另一观点来看,对话框的实际外观完全依赖于每种计算机的图形用户接口 (Graphical User Interface, GUI)。外观上的任何差异都是由于不同计算机所使用的不同显示驱动程序所导致的,所以,用户无需改变所使用的对话框设计或应用程序。例如,图 1-1-1 示出在两种不同类型的计算机上使用相同 DCL 语法的相同对话框。

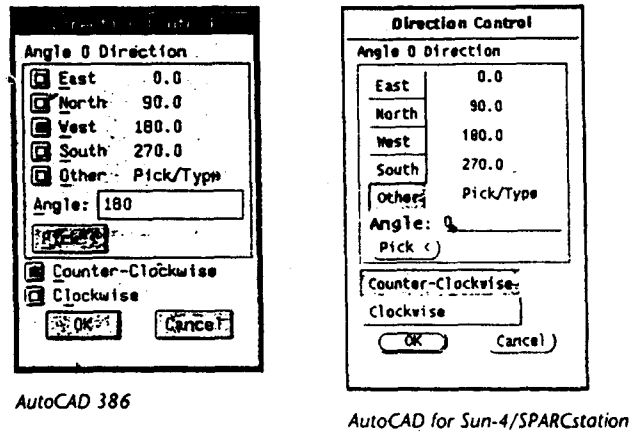


图 1-1-1 显示在不同类型计算机上的对话框

用户为什么希望在应用程序中使用对话框呢?原因之一就是为了维持对主机系统的观感,尤其是让应用程序运行于频繁使用窗口和对话框的系统中时(例如 Windows, the Macintosh 或 OPEN LOOK 等软件)。正是由于这些系统,使用户对拥有图形用户接口的应用软件的需求不断增加。另一个原因是,使用对话框比要响应连续提示文本的方式来得更容易、更自然,也更快一些。

设计一个对话框时,也需要考虑到用户在输入数据时顺序将会发生变化。这样会增加设计难度,而比起一般的设计似更不具“线性”性质,但可反映出用户的工作方式,所以在经过一些练习后,它也可变成较直观的操作方式。

若在开始设计之前,同时也对对话框和应用程序作详细的计划,则可节省时间并省去不少麻烦。若用户从未有过设计 GUI 应用程序或对话框的经验,请参考 1.4.6 节。

### 1.3 对话框简介

读者可能对 AutoCAD 对话框的使用方式已十分熟悉,它们的使用完全是自说明性的。图 1-1-2 示出了由标注部分所组成的一个标准 AutoCAD 对话框。在对话框的建立和设计过程中,对话框由控件(tiles)组成。

以下分别说明图 1-1-2 中各不同成分的含义和使用方式。

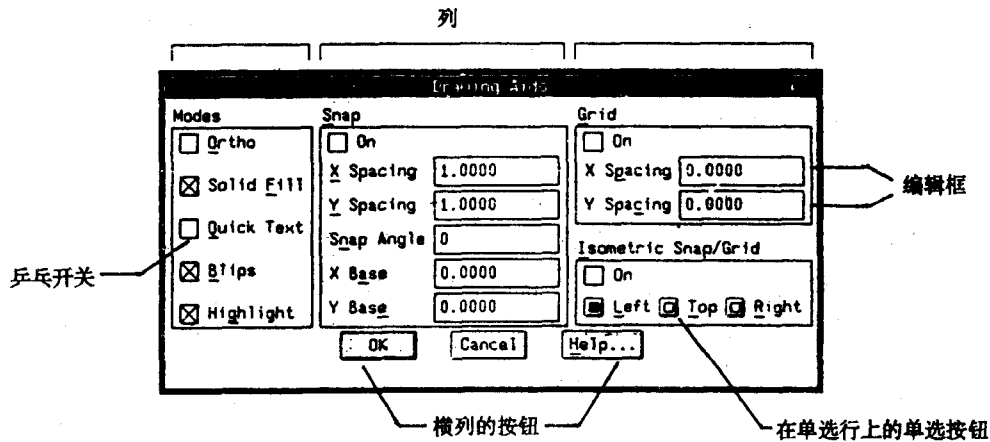


图 1-1-2 一个典型的标准对话框

有些计算机可让用户用键盘控制对话框,在此我们假定用户的计算机属于这种类型。

有些按钮对所有对话框来说是一条命令。例如,Tab 键(制表键)一般可从一个控件移到另一个控件;而按下空格键(Spacebar)则可选通或选断乒乓开关(taggle)。有两种用户熟悉的按钮,一个是接受键(通常是回车键),用以接受对话框,另一个是取消键(在某些类型的计算机上是 Esc 键,其他的则是 Ctrl+C 键,也可能两者都是),用以将它取消。一般来说,接受一个对话框即表示要保持用户的输入并以某种方式使用它们的值,取消一个对话框表示要放弃用户所输入的任何值。

当 AutoCAD 第一次显示一个对话框时,其中一个控件将含有初始的键盘输入光标,直到用户将光标移到另一个控件上时才会影响到这个初始控件。欲在控件之间移动,请按下 Tab 键,以移到另一个域上;也可按下按钮标号的首字符。例如,在上面的 DrawingAids 对话框范例中,可通过按下 L、T 或 R 等键,以在 Isometric Snap/Grid 控件中选择一个互锁按钮。有些计算机可能还要求同时按下 Alt 键,亦即必须按下 Alt+L、Alt+T 或 Alt+R 才可以移到互锁按钮上。除非选取一个按钮,否则可以改变光标,以在控件之间移动,但请勿作任何选择。

欲要选择一个所需的控件,必须按下接受键。对某些控件来说,双击鼠标左按钮相当于按下接受键。

在建立了一个自定义的对话框后,可以指定它的标号,控制哪个控件将含有初始光标,是否每个控件都可用 Tab 键到达,以及一个控件当前是否应被赋予功能(被解除功能的控件在显示时呈灰色)。

欲从一个 AutoCAD 画面中激活一个可编程的对话框以取代一个图形画面,仅需用一个 AutoLISP 函数或外部 ADS 函数来处理对话框即可。最简便的方法就是将该函数定义为一条 AutoCAD 命令(C:xxx)。例如,下述片段来自缺省的 acad.mnu 文件,显示文件下拉式画面的开头:

```
*** POP1
[Pile]
```

```
[New...]^ C^ C_new  
[Open...]^ C^ C_open  
[Save...]^ C^ C_qsave  
[Save As...]^ C^ C_saveas  
[Recover...]^ C^ C_recover  
[...]
```

在此范例中,每个选项都发出一条可处理对话框的命令(事实上,除了 New 选项外,所有选项都通过 getfiled 函数来激活标准的 AutoCAD 文件对话框)。

大部分支持 AutoCAD 的计算机对于它们所显示的对话框都有各自设置颜色的方式;对于那些不能设置对话框颜色的计算机(例如 DOS 386),若希望改变对话框的颜色,则可使用 AutoCAD 命令(DLGCOLOR)来改变它。

## 第二章 对话框的组成

### 2.1 引言

一个对话框是由位于其中的控件以及互锁按钮、乒乓开关或编辑框等部分所组成的。其基本控件类型(如按钮、编辑框、列表框、图象等)都是由 PDB 功能预先定义的,本章将说明它们的特点和使用方法。可以用包含(或不包含)的封闭对话框(或边界)将数个控件组合成更复杂的控件。这些组合本身也可被视为单一的控件,所以对话框可以被递归地组织成一个层次系统或树状结构,而位于树状结构中最顶端的就是对话框本身。

一个控件或其子组合的设计、外观及其行为都由 DCL 中控件的属性(attributes)定义。例如,对话框本身和大部分预定义的控件类型都有一个文本标号(label)属性,而一个对话框的标号即是位于其顶端的控件,一个按钮的标号即是位于按钮中的文本,依此类推。

DCL 也可以定义一与特定对话框无关的新控件(原型, proto-type)以及控件组(子组合, subassemblies)。可以使用改变预定义控件相同的方法来根据需要改变原型的属性。另一方面,子组合仅使用在不能改变其属性的“隔离层架”(off-the-shelf)下。

图 1-2-1 示出了一个对话框的树状结构实例。该树状结构的分枝都是已预定义好的控件,而该树状层次的顶端或其根部总是一个对话框。对话框说明用户建立在反映此树状结构的 DCL,同时一个对话框将其自身以一对话框的形式显示出来。

