

图形 / 图像处理丛书

最佳 Visual LISP 及 VBA for AutoCAD2000

程序 123 例

张光斌 编著

(附练习光盘)

用实例学 在实践中练

- ◆ AutoCAD开发环境及其最新发展
- ◆ Visual LIP 实例训练
- ◆ VBA 实例训练
- ◆ 问题解答

北京航空航天大学出版社

最 佳

Visual LISP 及 VBA for AutoCAD 2000

程序 123 例

(附练习光盘)

张光斌 编著

北京航空航天大学出版社

内 容 简 介

本书从编程技巧和实际应用角度出发,介绍 AutoCAD 开发环境及其最新发展,力求避免大篇幅论理,而是使用大量示例与说明,使读者按照示例练习,在实践中掌握 Visual LISP 和 VBA for AutoCAD 的功能。书中程序均在 AutoCAD 2000 中运行通过,绝大部分程序对早期版本也适用,并附有光盘一片。书中解答了常见的问题,附录对 Visual LISP 和 VBA for AutoCAD 进行了对比,并列出了 AutoCAD 2000 的主要文件类型和快捷键。本书可提供中高级编程人员使用。

图书在版编目(CIP)数据

最佳 Visual LISP 及 VBA for AutoCAD 2000 程序 123 例/

张广斌编著. —北京:北京航空航天大学出版社, 1999.9

ISBN 7-81012-924-4

I. 最… II. 张… III. 计算机辅助设计-应用程序-程序设计
II. IV. TP391.72

中国版本图书馆 CIP 数据核字 (1999) 第 33037 号

最佳 Visual LISP 及 VBA for AutoCAD 2000 程序 123 例

张光斌 编著

责任编辑 许传安

责任校对 陈 坤

北京航空航天大学出版社出版发行

(北京市学院路 37 号, 邮编 100083, 发行部电话 82317024, 发行部传真 82328026)

<http://www.buaapress.cn.net>

E-mail: pressell@publica.bj.cninfo.net

北京宏文印刷厂印刷 各地书店经售

787×1092 1/16 印张: 20.25 字数: 518 千字

1999 年 9 月第 1 版 1999 年 9 月第一次印刷 印数: 4000 册

ISBN 7-81012-924-4/TP·366 定价: 45.00 元

前 言

从菜单和工具栏直到每一个图形对象,都可以精确地定制 AutoCAD 2000 并进行二次开发,工具包括 Visual LISP, Microsoft Visual Basic for Applications (VBA) 和 ObjectARX。为了取得最大限度的灵活性, AutoCAD 2000 允许在图形文件中嵌入多个 VBA 程序项。如果是 AutoLISP 的老用户,一定会喜欢在 AutoCAD 2000 内部集成的、具有全新界面的 Visual LISP。

Visual LISP 是一个完整的开发语言,可用于在 AutoCAD R14 和 AutoCAD 2000 下开发应用程序,实现用户化的操作。它运用了 Autodesk 公司的 ObjectARX 面向对象技术和 ActiveX 对象容器技术,易于开发和操作,大大扩展了 AutoLISP 的编程能力,使用其编写的软件比用 AutoLISP 开发的更具智能化,运行速度更快。

VBA 在集成开发环境中提供了高质量的用户化编程能力,能够使得 AutoCAD 数据与其它 VBA 应用程序直接共享,如 Microsoft Office 97 软件。最重要的是 VBA 的加入,扩展了 AutoCAD 集成用户化工具的集成能力。它集成了 AutoLISP, Visual LISP 和 ObjectARX API 等工具。这样,就为访问 AutoCAD 2000 软件的多种技术框架提供了新的选择和新的开发机会——可以按所需求的工作方式应用程序用户化,也可以从战略上考虑开发基于 AutoCAD 的应用程序。

作者长期从事机械、飞行器结构产品的设计、开发及相关的教学工作,较长时间使用 AutoCAD 进行过产品设计,对该软件的特点、功能及应用有较深入的理解和体会。本书不仅向读者介绍了 AutoCAD 开发环境及其最新发展,还通过大量实例给出了应用 Visual LISP 和 VBA 的经验和体会,具有重要参考价值。

本书第一章介绍 AutoCAD 2000 的开发环境及技术框架等新特性;第二章和第三章给出 123 个 Visual LISP 和 VBA 实例及说明;第四章对疑难问题进行了解答;附录对 Visual LISP 和 VBA 进行了对比,并列出了 AutoCAD 2000 的主要文件类型和快捷键。

本书是在作者的《AutoCAD LT for Windows 培训教程》、《AutoCAD R14 for Windows 实例教程》和《晋级 AutoCAD 2000 及实例训练》等书(均由北京航空航天大学出版社出版)的基础上,为满足中高级 AutoCAD 用户的需求而推出的。书中程序附有光盘一片,其目录的编排是按本书的节(2.1, 2.2……3.1, 3.2……)而划分的,所有程序均在 AutoCAD 2000 中运行通过,绝大部分程序对早期版本也适用但,并未经过严格的测试。

本书由张光斌编写,在编写过程中得到了刘 斌、卢豫麟、欧阳志平、杜 杰等同志的关心与支持,并提出宝贵意见,杨 慧同志也给予了大力的帮助,在此表示诚挚的谢意。

由于本书的编写时间仓促和限于编者水平,书中错误与不妥之处在所难免,恳请读者批评指正。

编 者

1999 年 6 月

目 录

第一章 AutoCAD 开发环境及其最新发展.....	1
1.1 AutoCAD 早期版本的开发环境.....	1
1.1.1 AutoLISP	1
1.1.2 ADS.....	2
1.1.3 ARX	3
1.2 内部集成 Visual LISP.....	4
1.3 VBA 的应用.....	6
1.4 ObjectARX 编程接口功能扩充	7
1.5 Java 的支持	8
第2章 Visual LISP 实例训练.....	10
2.1 2D 绘图	10
2.1.1 绘制角平分线.....	10
2.1.2 绘制螺栓.....	12
2.1.3 根据范例绘图.....	18
2.1.4 绘制多个圆的中心线.....	19
2.1.5 绘制两圆交叉公切线.....	22
2.1.6 局部放大之一.....	23
2.1.7 局部放大之二.....	28
2.1.8 绘制带对角线的长方形.....	30
2.1.9 椭圆转换为弧段.....	31
2.1.10 线段端点连接.....	35
2.1.11 绘制齿轮.....	38
2.1.12 绘制小草.....	40
2.1.13 绘制网格.....	42
2.1.14 绘制工字型梁.....	43
2.1.15 绘制“L”型剖面梁.....	46
2.1.16 绘制“无限长”水平线和垂直线.....	48
2.1.17 两线交叉跨越示意.....	49
2.1.18 绘制平行线.....	52
2.1.19 绘制管子剖面.....	53

2.1.20	绘制平板.....	55
2.1.21	参数化设计阶梯轴.....	57
2.1.22	圆周开槽.....	65
2.1.23	绘制任意角的星形.....	67
2.1.24	绘制与三点相切的圆.....	69
2.2	3D 造型.....	70
2.2.1	绘制 2D 和 3D 螺旋线.....	70
2.2.2	绘制 3DFace.....	74
2.2.3	绘制连续弯折的 3D 管线.....	75
2.3	图形编辑.....	80
2.3.1	沿指定角度阵列对象.....	80
2.3.2	在交点处断开线.....	82
2.3.3	改变文本内容.....	83
2.3.4	显示并修改单个实体的属性.....	84
2.3.5	擦除线段.....	88
2.3.6	复制对象到指定层.....	89
2.3.7	沿指定方向多重复制对象.....	90
2.3.8	复制并旋转对象.....	92
2.3.9	断开对象.....	95
2.3.10	双重偏移.....	96
2.3.11	改进的倒圆角命令.....	97
2.3.12	连接直线.....	99
2.3.13	绘制十字交叉路口.....	100
2.3.14	改变线或弧的线宽.....	102
2.4	文本标注.....	103
2.4.1	面积标注.....	103
2.4.2	转换大小写.....	104
2.4.3	Mtext 转换成 Text.....	106
2.4.4	文本添加下划线.....	108
2.4.5	绘制带阴影框的文本.....	110
2.5	尺寸标注.....	111
2.5.1	标注直线的方向和长度.....	111
2.5.2	测量并标注弧长.....	113
2.5.3	“气泡”标注.....	115
2.5.4	点的坐标标注.....	121
2.5.5	曲线型旁注线.....	121
2.5.6	绘制特殊的旁注线.....	124
2.5.7	标 签.....	125
2.6	图 块.....	136

2.6.1	创建并插入匿名块.....	136
2.6.2	改变属性文本高度.....	137
2.6.3	拆开不定比例的块.....	138
2.6.4	插入块或外部块.....	140
2.6.5	块内容列表.....	144
2.6.6	组合与解散实体.....	145
2.6.7	再插入同样块.....	147
2.7	图 层.....	150
2.7.1	将实体颜色改变为随层.....	150
2.7.2	实体改变到目标层.....	151
2.7.3	设置拾取对象所在图层为当前层.....	152
2.7.4	复制到当前层.....	153
2.7.5	解冻层.....	154
2.7.6	删除层上的实体.....	156
2.7.7	隔离层.....	160
2.7.8	冻结层.....	161
2.7.9	新建层.....	162
2.7.10	移动层上的实体.....	163
2.7.11	保存图层设置.....	163
2.8	实用工具.....	168
2.8.1	两线夹角测量.....	168
2.8.2	矩形面积测量.....	170
2.8.3	单位转换.....	171
2.8.4	改进的 LIST 命令.....	175
2.8.5	列出绘图环境.....	187
2.8.6	体积测量.....	189
2.8.7	实时放大.....	189
2.9	趣味话题.....	191
2.9.1	自动计数.....	191
2.9.2	分形图案.....	193
2.9.3	数字递增复制.....	196
2.9.4	画龙和曲线图案.....	198
2.9.5	变 脸.....	202
2.9.6	绘制松果.....	208
2.9.7	质数计算器.....	211
2.9.8	改进的多边形.....	217
2.9.9	自动产生对话框.....	220

第3章	VBA 实例训练.....	230
-----	---------------	-----

3.1 2D 绘图与编辑	230
3.1.1 创建点对象	230
3.1.2 绘制多义线	230
3.1.3 绘制一组放射线	231
3.1.4 绘制一组圆	232
3.1.5 绘制剖面线	233
3.1.6 改变实体颜色	234
3.1.7 实时改变线型比例	235
3.1.8 改变多义线的宽度	237
3.2 3D 造型与编辑	239
3.2.1 创建实心填充区域	239
3.2.2 创建面域	239
3.2.3 创建 3D 网格	240
3.2.4 3D 工字梁	241
3.2.5 3D 旋转	249
3.2.6 编辑 3D 实体	251
3.3 文件管理	252
3.3.1 新建绘图	252
3.3.2 DWG 文件管理器	253
3.3.3 DVB 文件加载器	261
3.4 文 本	268
3.4.1 创建文本	268
3.4.2 创建多行文本	268
3.4.3 文本、数据的输入与绘图	269
3.4.4 文本查找与替换	273
3.4.5 改变文本的高度	277
3.5 输入输出	279
3.5.1 控制输入	279
3.5.2 打印输出模型空间的绘图	280
3.5.3 打印输出图纸空间的绘图	281
3.6 视图控制	282
3.6.1 拆分视口	282
3.6.2 控制视口	282
3.6.3 定义用户坐标系统	283
3.7 图 层	285
3.7.1 新建图层	285
3.7.2 搜索图层	285
3.7.3 图层管理标准化	286
3.7.4 改变图层	290

3.8 实用工具	292
3.8.1 检索图形数据库中的第一个实体	292
3.8.2 计算距离	293
3.8.3 计算面积	294
3.8.4 射线查询	295
第4章 问题解答	297
附 录	304
附录 A Visual LISP 和 ActiveX/VBA 函数对比	304
附录 B AutoCAD 2000 主要文件类型	310
附录 C AutoCAD 2000 的快捷键	311

第一章 AutoCAD 开发环境及其最新发展

1.1 AutoCAD 早期版本的开发环境

AutoCAD 于 1986 年 V2.18 版时提供了 AutoLISP 程序设计的支持,从此了可以在一个通用的 CAD 平台上添加自己所需的特殊功能的能力。1987 年 AutoCAD V2.6 版推出, AutoLISP 得到一些改善。这也正是流行 AutoCAD 及在 AutoCAD 环境下开发编程的开始。1988 年推出的 AutoCAD R10 使得 AutoLISP 程序可以在扩展内存中运行。这对于编写稍大一点规模的应用程序提供了可能。AutoCAD 第一代开发工具开发出的第一代应用软件开始流行。

1990 年推出的 AutoCAD R11 第一次在 PC 版的 AutoCAD 上提供了 C 语言开发环境 (ADS)的支持。这使得在 AutoCAD 上开发大规模的、综合性的应用程序成了可能。AutoCAD 下的应用程序可以利用标准 C 语言提供的能力,编制出第一代工具无法实现的应用。真正在国内普遍流行用第二代开发工具开发 AutoCAD,是在 1992 年推出的 AutoCAD R12 的基础上进行的。

1994 年推出的 AutoCAD R12,第一次提供了面向对象的 C++开发环境的支持 (ARX),这就是 AutoCAD 第三代开发工具。AutoCAD 本身在内核上也发生了本质的变化,尽管最终用户并不会感觉到,但 AutoCAD 确实走向了一个开放式的面向对象的 CAD 平台,为今后的进一步发展奠定了技术基础。很遗憾的是 R13 上的 ARX 是个新生事物,本身不完善,在不同的小版本上的 ARX 程序不兼容,直至 R13C4 版才较为稳定。ARXSDK 发布的初期所要求的开发环境也较为苛刻(要求 MSVC2.1~2.2),现在已经可以利用 MSVC 4.x 作为开发环境。1997 年发布的 AutoCAD R14 是第三代开发工具应用的舞台,一方面 AutoCAD R14 对 ARX 的支持更加可靠和稳定;另一方面 ARX 经过一定时间的摸索,二次开发者逐步具备开发 ARX 应用程序的能力。

1.1.1 AutoLISP

AutoLISP 是 AutoCAD 内嵌的一种解释语言。AutoCAD 从 V2.18 版开始,一直支持这种语言。AutoLISP 的大部分功能是直接针对 AutoCAD 的。它是一种很好的交互式语言,很适合于 CAD 这类交互性很强的应用。

AutoLISP 有如下优点:

1. 语言规则十分简单, 易学易用

AutoLISP 来源于 LISP 语言, 以一种通用的规则——表 (LIST), 来描述过程和数据。由于这种规则极其简单, 一个对计算机了解不太深入的工程技术人员, 仅需短期的学习, 就可以熟练掌握这门语言。LSIP 语言的简洁性比其他许多计算机语言所无法相比的。“万能”的表代替了复杂的过程, 可描述任何数据结构。

2. 直接针对 AutoCAD, 易于交互

AutoLISP 提供的大多数函数都是直接针对 AutoCAD 的操作, 用它编写访问 AutoCAD 图形数据库 (DWG) 的操作。

3. 解释执行, 立竿见影

程序员既可以一次运行一段 AutoLISP 代码, 也可以在 AutoCAD 的提示符: Command: 下逐条键入 AutoLISP 指令, 立即得到结果。

随着计算机技术的发展, CAD 编程更加复杂, 代码越来越庞大, AutoLISP 的缺点亦愈来愈明显:

- 功能单一, 综合处理能力差;
- 解释执行, 程序运行速度慢;
- 缺乏很好的保护机制, 软件质量不易保证;
- LISP 用表来描述一切, 并不能很好的反映现实世界和过程, 跟人的思维方式也不太一致。

AutoLISP 的这些特点, 仅适合于有能力的终端用户做些自己的开发任务。

1.1.2 ADS

ADS (AutoCAD C 语言开发环境) 是 AutoCAD R11 开始支持的一种基于 C 语言的灵活的开发环境。

R11 上的 ADS 程序的编译环境是不流行的 C 编译器、HighC 和 WatcomC 以及 ZotechC 的。R12 for DOS 虽然支持流行的实模式的 Borland C / C++ 编译器 and Microsoft C / C++ 编译器, 但 640KB 的局限性太大。R12 for DOS 上的主要编译器仍是 HighC 和 WatcomC。在 R12 for Windows 版上, Borland 公司和 Microsoft 公司的编译器已经可以很好的发挥作用了, 不过对运行效率和规模较大的应用程序来说, Watcom 公司的编译器是更好的选择。

ADS 程序的许多特点正好同 AutoLISP 相反, 它比 AutoLISP 程序优越的地方在于:

- 具备错综复杂的综合大规模处理能力;
- 编译成机器码后执行的, 速度快;
- 编译时可以检查出程序设计语言的逻辑错误;
- 程序源码的可读性好于 LISP。

与 AutoLISP 相比, ADS 的不便之处在于:

- C 语言比 LISP 语言要复杂得多, 难以在短时间内掌握, 熟练应用需要更长的周期;
- ADS 程序的隐藏错误往往导致 AutoCAD, 乃至操作系统的崩溃;
- 需要编译才能运行, 不易见到代码的效果;

- 同样功能 C 语言的 ADS 程序一般比 AutoLISP 程序的源代码要长不少，比较繁冗。

用 ADS C++进行编程有如下优势：

- 为全部面向对象的编程，使得程序层次分明，清晰可读，从而提高软件的可维护性；
- 经过良好封装 ADS C++的程序代码比 ADS C 程序简洁得多，许多时候比 AutoLIPS 更简洁；
- ADS C++的程序能够减少程序出现错误的概率。经 C++封装后的选择集和实体不再需要调用 `ADS μ ssfree` 和 `ADS μ relrb` 进行内存释放，可以由对象的析构函数负责内存的释放。

1.1.3 ARX

ARX (AutoCAD 运行时间扩展) 是 AutoCAD R13 之后推出的一个全新的面向对象的开发环境，也是 AutoCAD 第一次直接提供面向对象的二次开发工具。ADS C/C++使得用户可以在 AutoCAD 上开发大规模综合应用软件。然而计算机技术的发展不仅要求 CAD 的交互性和综合性，同时对自动化和智能化也提出了更高的要求。可以说 AutoLISP 着眼于应用程序的交互性，ADS C/C++着眼于应用程序的综合性，而 ARX 则着眼于应用程序的智能性。

ARX 程序有如下特征：

1. 能力强

ARX 程序具有 ADS 程序和 AutoLISP 程序所不能具备的访问和控制 AutoCAD 的能力。ARX 程序可以监控和处理 AutoCAD 的各种事件，可以定义 AutoCAD 命令，包括可以透明执行的命令。这些命令可以像 AutoCAD 自身的命令一样被 AutoLISP 和 ADS 调用。可以定义 AutoCAD 数据库对象，包括可见的实体和不可见的实体。这些对象的行为由程序规定，包括对象在各个视窗中的显示和对各种事件的响应。ARX 定义的实体往往可以用较少的数据，就可以实现较复杂的显示，有利于减少磁盘文件的大小。ARX-COM 程序是新一代开放式 ActiveX 程序，可以被其他 Windows 应用程序访问。

2. 速度快

这里指的是访问 AutoCAD 数据库的速度。ARX 应用程序本身是 AutoCAD 的一部分，即被 `ACAD.EXE` 调用的动态链接库 (DLL)，拥有同 AutoCAD 一样的内存地址空间，直接访问 AutoCAD 的各种内存对象。过去 AutoLISP 和 ADS 都是通过函数间接地访问 AutoCAD。以 ADS 读取 AutoCAD 数据库为例，通过 ADS 库函数把 AutoCAD 数据库对象变换成 `resbuf` 链表，然后访问 `resbuf` 链表，需要变换数据结构和遍历链表两部分时间的消耗。这样在访问 AutoCAD 数据库的操作上，ARX 应用程序比 AutoLISP 和 ADS 的速度要快得多。

3. 面向对象的编程

ARX 提供了 6 个主要类库来访问和控制 AutoCAD 及处理 CAD 常见的事物，它们是：

- `AcRxAutoCAD` 运行扩展库；
- `AcEdAutoCAD` 命令扩展库；
- `AcDbAutoCAD` 数据库访问库；
- `AcGe` 通用几何库。
- `AcGiAutoCAD` 图形界面库；

- AcBr 边界描述库;

然而, ARX 应用程序比 ADS 程序具备更大的危险性和破坏性。ARX 程序设计比 ADS 要复杂得多。ARX 程序设计的错误, 经常导致 AutoCAD 崩溃, 甚至操作系统的崩溃。程序员需要有很高的素质, 才能编制好 ARX 应用程序。此外, ARX 应用程序是一个动态链接库, 调试时需频繁启动 AutoCAD (DOS 下), 额外的时间消耗比 Windows 下的 ADS 要多。这些都说明, 开发 ARX 应用程序, 需要更多的投入。

AutoCAD 2000 提供了完整、高性能且面向对象的 CAD 程序开发环境, 从菜单和工具栏直到每一个图形对象, 都可以精确地定制, 为用户和开发者提供了多种选择。这些选择包括基于 C++ 的 ObjectARX、基于 ActiveX Automation (AutoCAD 的 COM 接口) 的 VBA 和 VisualLISP。所有这些开发接口都能与 AutoCAD 的 ActiveX Automation, 或 COM 接口对象协同工作。AutoCAD 的开放体系结构是以组件为基础的结构, ActiveX Automation 正是 AutoCAD 的组件之一。ActiveX Automation 使得 AutoCAD 的应用程序可以用任何编程语言表达, 例如, C++ (ObjectARX) 可以编写较高性能的专业组件, 而 VB 和 Delphi 则用于快速开发, Java 语言也不例外。因此, 与大多数 Windows 应用程序相同, AutoCAD 的 ActiveX Automation 是使 Java 成为 AutoCAD 2000 所支持的开发语言的关键技术。

1.2 内部集成 Visual LISP

Visual LISP 是 AutoLISP 的换代产品。它与 AutoLISP 完全兼容, 并提供它所有的功能, 同时它能访问新的多文档设计环境。COM/ActiveX 用户界面、事件响应器等。Visual LISP 同时提供了新的编程环境。该环境提供括号匹配、跟踪调试、源代码和语法检查等工具, 方便了创建和调试 LISP 程序, 实现用户化和编写应用程序。

用户和开发者可以充分利用完全集成在 AutoCAD 内部的 LISP 开发环境。对于十分熟悉的 AutoLISP 语言, 利用可视化的编程环境可以使学习难度下降。在线帮助提供了样例程序和教程。Visual LISP 教程以 PDF 文件格式提供, 可以方便地打印生成硬拷贝文件。作为一个完整的用户化开发环境, Visual LISP 可以迅速而方便地建立自己的高效解决方案 (图 1.2.1), 对 AutoLISP 源程序提供自动格式化功能。在 VisualLISP 中的自动格式化选项可以编程控制 (图 1.2.2)。

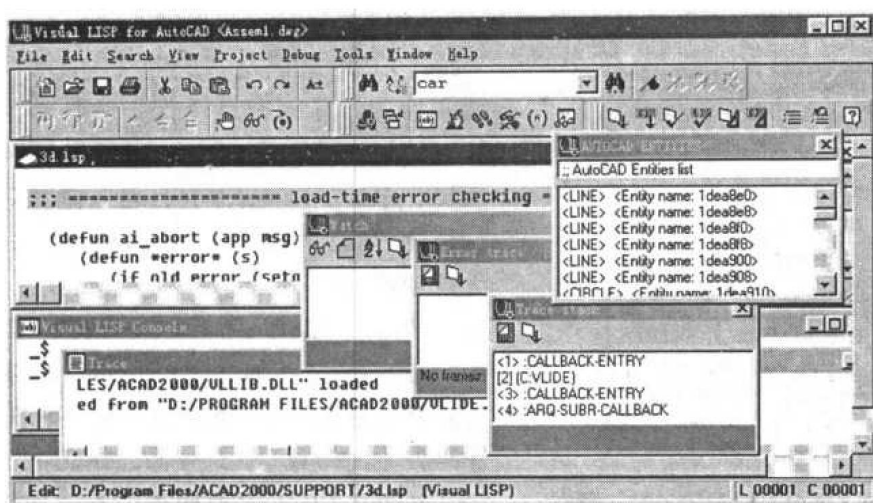


图 1.2.1 Visual LISP for AutoCAD 编辑器界面

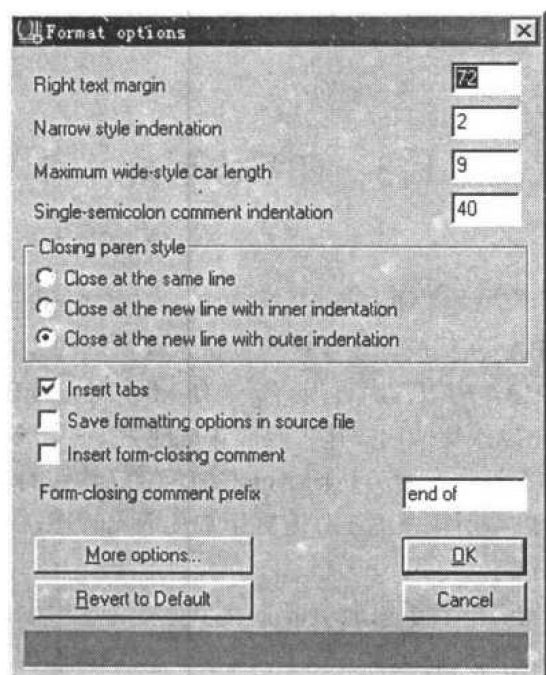


图 1.2.2 【Format options】(格式选项) 对话框

集成在 AutoCAD 内部的 Visual LISP 开发环境带有完整的排错工具。Visual LISP 编程语言中具备的带颜色标志的语法检查系统可以提供实时的语法检查反馈，有助于发现打字输入时出现的错误。括号匹配功能将对检查程序中不平衡的括号提供帮助。Visual LISP 中包括动态显示功能，可以亮显正在运行的语句，帮助分析程序运行状况和程序排错等等。在 LISP 程序段中可以设置断点，加上动态显示功能可以监控某一程序段的运行。Visual LISP 对标准 AutoLISP 符号提供了保护功能，在需要时显示警告信息并且在符号缺失时提供中断程序运行的能力。

【Apropos】(适配)功能可以对特定函数显示符号表，缩短程序启动时间(图 1.2.3)。

使用 Visual LISP，程序开发人员可以编译 LISP 程序，加快程序的运行速度。编译后的



图 1.23 【Apropos options】(适配选项) 对话框

Visual LISP 代码是二进制代码，使用者无法阅读和修改，从而有助于保护软件算法和知识产权。使用编译后的 LISP 应用程序，可以降低对应用程序的支持需求，并且可以更容易地使 CAD 标准化。因为编译的程序可以防止使用者有意或无意地修改。

Visual LISP 中包含 LISP 与 ActiveX Automation 之间的接口，可以访问实体对象的方法和属性。对象反应器程序将在相关 AutoCAD 事件发生时运行。

Visual LISP 对反应器的支持包括编辑器反应器、连接器反应器和数据库反应器，可以使用操作系统级的文件访问函数。Visual LISP 具备有扩展的表处理工具库。Visual LISP 变量支持跨越图形。

1.3 VBA 的应用

AutoCAD 2000 包含了 VBA (Microsoft Visual Basic for Applications) 5.0 应用程序开发应用语言 (图 1.3.1)。VBA 在集成开发环境中提供了高质量的用户化编程能力。它能够使得 AutoCAD 数据与其它 VBA 应用程序直接共享，如 Microsoft Office 97 软件。最重要的是 VBA 的加入，扩展了 AutoCAD 集成用户化工具的集成能力。它集成了 AutoLISP, Visual LISP 和 ObjectARX API 等工具。这样，就为访问 AutoCAD 2000 软件的多种技术框架提供了新的选择和新的开放机会，可以按所需求的工作方式用户化应用程序，也可以从战略上考虑开发基于 AutoCAD 的应用程序。

在 AutoCAD 中为开发者配备了更加丰富的 ActiveX 对象用于自定义和编程。用户和开发者现在可以通过更多的事件来触发应用程序的运行。在 AutoCAD 中可以构造和装入多个 VBA 工程。可以把 VBA 工程嵌入到 DWG 文件中，由此可以将图纸所需要的应用程序与图纸一起传递。

使用 VBA，可以通过拖拽图标的方法，简单而可视化地建立用户界面。既可以使用 AutoCAD 工具栏，也可以使用标准 VBA 界面工具创建新的工具栏。

可以从大量的程序例子中拷贝粘贴已有的程序或通过各种在线帮助，它的智能感可以指导在每一步作出适当选择来编辑程序。颜色代码文本可以按代码段识别所执行的特定任务，所以当代码改变时能立即知道应该如何去做，甚至可以将“仅作参考”这样的信息插入到代码中作为以后的参考点。

专业 VBA 程序员会发现，AutoCAD 2000 软件中的 ActiveX Automation 对象模型十分有用。Autodesk 开放了每一个 AutoCAD 对象模型。这些对象模型显示了每一个对象的方法

和属性，并能够按着用户的愿望去编程或用户化 AutoCAD 2000 软件。

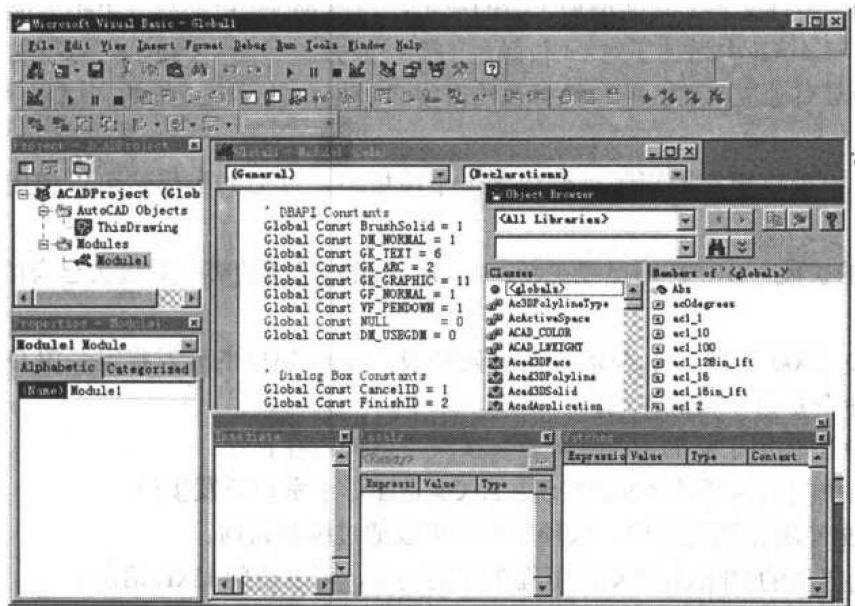


图 1.3.1 Microsoft Visual Basic 编辑器界面

无论编程技术水平如何，使用 VBA 编程都是简单而迅速的。使用 VBA 能够节省编程时间的一个例子是在一个图形文件中可以嵌入并运行多个 VBA 程序。

1.4 ObjectARX 编程接口功能扩充

开发者可以从 ObjectARX AcDb 的基本类中导出 AutoCAD 的所有对象。

用户化的【Osnap】（对象捕捉）模式使用新的 API 来定义、注册和处理用户化的 Osnap 模式。新的 Osnap API 协议定义了如何返回自定义的 Osnap 对齐线和 Osnap 点。

输入点监视器使应用程序可以注册回调程序，用于为输入点或光标追踪事件监控输入点计算履历，为追踪输入点获取过程提供更多的通报回调过程。

布局功能提供了创建和控制多重布局的开发接口。

局部打开功能提供了文件局部打开过程的可编程控制，适用于大图形的管理。

可完成 AutoCAD R12 实体的对象化过程和实现过去在 AcDbEntity 或 AcDbCurve 中声明的全部虚拟成员。

提供外部参照编辑 API，使应用程序可以选择外部参照或深入到图块参照之中，编辑其中的图元并保存修改。

【Saveas】（另存为）过程可以通过编程决定默认文件格式。

三维实体编辑功能提供的新的方法可支持三维实体编辑命令中的操作。

启动向导的自定义功能增加了支持新向导定义和设置系统注册表，使得系统启动时指向自己的启动向导。

图纸属性功能提供了访问 DWG 图纸文字描述信息的能力。

【Mtext】(多行文本)功能增加了新的成员,使得 Mtext 更加具有可扩展能力。

在 AcDb 和 AcGi 中增加了对 24 位颜色的支持能力。

各视口中具有独立 UCS 的特性开放给开发者使用,使面向三维的应用程序具有更大的灵活性。

AcArray 模板将所有的矩阵类合成到一个模板中,过去的类仍然可以使用但不推荐使用。

AcBr 库的扩展, BREP 库已经更新; AcGe 库增添了新的类,对过去已有的类增加了新的功能。

AutoCAD 2000 不再支持 ADS 编程开发环境。AutoCAD 的开发将完全由 ObjectARX 来取代, ADS 命令是可以重进入的。

AutoCAD 中使用了新的内存管理器,实现更好的内存管理。

对图形系统的访问通过 AcGs 实现。AcGs 是图形子系统开发接口。

为 DXF 文件添加预览图形。该预览图形可以通过编程访问。

通过特别提供的 ObjectARX 应用程序开发接口,可以访问 URL 信息。

通过 AcDBEntity 支持显示和打印过程中新增的线宽信息。

多文档接口 (MDI)——ObjectARX MDI 应用程序开发接口提供对应用程序对象、文档管理对象、文档锁定与解锁和文档事件通报的支持。

MFC 标签形式对话框扩展——动态链接库 AcUi 是 MFC 的扩展。它用于扩展 MFC 中与界面相关的 MFC 基础类库。

通过 AcDb 布局对象,可以编程访问布局功能,在一个 DWG 文件中创建多个布局。AcDb 视口对象支持非矩形视口剪切。

可以在应用程序中确定 OLE 对象是否被选中。

在【Options】(选项)对话框中的所有参数都可以通过类 AcAppProfileManager 中的方法来访问。

新的 ActiveX API 中支持设计中心、数据库连接、打印输出、Internet 访问、对象特性管理器 (OPM)、选项参数设定、MFC 工具栏和菜单扩展。

【ISM】(光栅图形支持模块)功能也有了增强。

应用程序可以支持 Intellimouse 特性并直接截获鼠标事件。

1.5 Java 的支持

AutoCAD 支持 Java 编程,是继 AutoLISP, ADS, ARX, ObjectARX, VBA 和 VisualLISP 之后, AutoCAD 拥有的又一种新开发工具。

通过微软的 Visual J++集成开发环境,开发者能够使用 Java 语言操作 AutoCAD 的内部数据。在某些情况下,甚至可以把从网上下载的 Java 脚本代码加入到已有的 AutoCAD 应用程序中,从而大大拓宽了应用程序的可用性。