

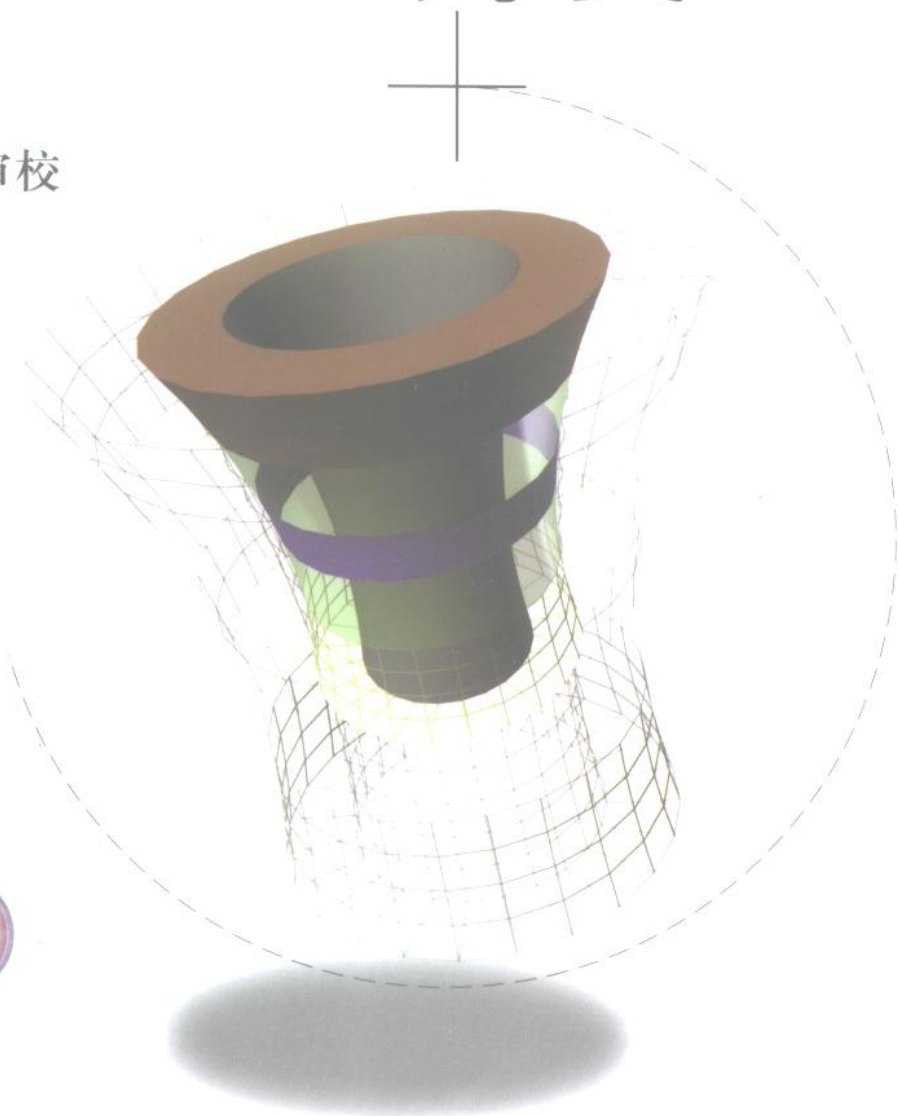
AutoCAD 2000



AutoCAD 2000
实用技术丛书

Visual LISP 开发

阎聚群 编
抖斗书屋 审校



72.9633
832

AutoCAD 2000 Visual LISP 开发

阎聚群 编

抖斗书屋 审校



华中理工大学出版社

中国 · 武汉

.. 2001106

图书在版编目(CIP)数据

Auto CAD 2000 Visual LISP 开发/阎聚群 编
武汉:华中理工大学出版社, 2000年1月
ISBN 7-5609-2130-2

I. A...

I. 阎...

Ⅲ. Auto CAD 2000-Visual LISP-基本知识

N. TP319

JS115/14

本书封面贴有华中理工大学出版社激光防伪标志,无标志者不得销售。

版权所有 盗印必究

Auto CAD 2000 Visual LISP 开发

阎聚群 编

责任编辑:周筠 任卫群

封面设计:潘 群

责任校对:戴文遐

监 印:张正林

出版发行:华中理工大学出版社

武昌喻家山 邮编:430074 电话:(027)87542624

经销:新华书店湖北发行所

录排:北京中科辅龙公司抖斗书屋

印刷:荆州市今印集团有限责任公司

开本:787×1092 1/16

印张:20.75

字数:470 000

版次:2000年1月第1版

印次:2000年1月第1次印刷

印数:1—5 000

ISBN 7-5609-2130-2/TP·359

定价:32.00元

(本书若有印装质量问题,请向出版社发行科调换)

001200X

内 容 简 介

本书面向 AutoCAD 的用户，从实用的角度出发，介绍了 Visual LISP 的使用方法、AutoLISP 的主要函数和 AutoCAD2000 的新增变量。为了使具有极少量 AutoLISP 编程背景的用户也可以使用本书，本书从最初步的操作开始讲起，一直讲到一个完整的 LISP 程序的调试和运行过程，并对每一步操作尽可能地使用屏幕图进行详细说明。

本书分为上、下两篇：Visual LISP 的使用；AutoLISP 基础和函数列表。

上篇 Visual LISP 的使用共 6 章，主要讲解了 Visual LISP 的界面、各种开发工具、程序的调试、应用程序的创建和维护、ActiveX 应用等。

下篇 AutoLISP 基础和函数列表共 3 章，主要对 AutoLISP 的语言基础、AutoLISP 的常用函数和 Auto CAD2000 中新增的一些函数进行了详细的讲解。用户即便以前没有 Auto LISP 语言的编程经历，也可以通过对本书后几章的学习对 Auto LISP 编程形成初步的概念，并进行初步的程序编写和调试运行。

本书结构清晰，风格严谨，既可以作为 Visual LISP 及 AutoLISP 的学习用书，亦可以作为平时使用的速查手册。

前 言

1 Visual LISP 与 Auto LISP

AutoLISP 语言是用来扩展和定制 AutoCAD 功能的编程语言。它以 LISP 编程语言为基础。LISP 语言是 50 年代后期产生、用来做人工智能的编程语言，现在仍是很多人工智能应用的基础。

AutoLISP 是在 AutoCAD2.1 版中作为一个应用程序接口(API)引入的。而选择 LISP 语言作为 AutoCAD 的 API 是因为它适合 AutoCAD 设计过程的无组织性，因为设计问题有许多不同解法。

Visual LISP (VLISP)是用来加速设计 Auto LISP 程序的软件工具。VLISP 是一个集成的开发环境，它可以使 AutoLISP 程序中的源代码编辑、程序测试与调试更加容易。同时 VLISP 还提供了一个传送工具，可以独立地传送用 AutoLISP 编写的程序。

过去用 AutoLISP 编写 AutoCAD 的应用程序，意味着用字处理工具编写代码，然后把这些代码装入 AutoCAD，然后再运行它。而调试程序更是意味着必须从整体上来考虑每一个变量和每一段代码。而且最要命的是必须算出每一段程序做出的结果和每一个变量的变化。如果仍然不能确定错误原因，就得重新修改代码，重新调试程序，比如加上一些断点、一些提示信息，等等。最后终于通过了调试，但还得把这些断点、提示信息等统统去掉。真是麻烦的要命。可是在没有 VLISP 的时候，每一个 AutoLISP 的程序员都是这么做的，现在大家就知道程序员的苦了。

虽然如此，AutoLISP 还是给我们带来了无比的振奋，因为它能让我们自己定制富于个性的界面和实用化的界面。同时 AutoLISP 是内嵌在 AutoCAD 中的，既可以在字处理软件中编写程序，同时还可以在 AutoCAD 的命令行中直接键入 AutoLISP 程序。比起另外的一些图形处理软件来，可以说是非常简单了。

AutoLISP 是以 ASCII 码存储的、以“.LSP”为后缀的文本文件。这些文件有一个头描述，它能描写程序和它的用处。这个头描述也可以包括注释，比如作者的信息和一些合法的描述。这个描述是以“；”开头的，它能在字处理软件中看到。所有这些都能看出它的简便性。

在 AutoLISP 应用与程序的发展周期中，AutoLISP 程序员能执行一些原 AutoCAD 中不存在的操作。一些操作，比如文本编辑，用另外一些程序也能实现。而另外的一些操作，比如源代码级的调试，就只能使用 VLISP 了。在 VLISP 中，可以在这个单一的环境中执行大部分的必要的操作。比如文本编辑、程序调试和与 AutoCAD 交互等等。

但是，AutoCAD 能够直接读出 AutoLISP 代码，根本不需要编译，而且 VLISP 提供了一个集成开发环境，可以选择在 AutoCAD 模式下输入代码。这样，就能直接看出结果，使得 AutoLISP 的编程更加简单化，而与用户的编程经验没有什么关系。但是如果用户已经熟练掌握了 AutoLISP，你就能够更快地掌握 VLISP 的使用。即便用户不大喜欢编制 AutoLISP

程序,也能够快速地编制出 LISP 程序来。因为 AutoCAD 提供了大量常用的程序,在此基础上用户可以适当加以修改,编制自己需要的程序。如果用户清楚地知道如何装载和使用哪些程序,那么无疑,用户的工作效率将大大提高。

2 一点说明

作为一个用户,必须熟知 Visual LISP 文件的格式,否则将寸步难行。为此,用户必须认真阅读 AutoCAD 的自述文件中有关 Auto LISP 和 Visual LISP 的注意事项。这些注意事项在 Auto CAD2000 的帮助菜单下都可以找得到。特别是以前做过 Auto CAD 较低版本的应用软件开发,就更应该详细查阅一下现行版本与以前版本的不同之处,这样就不会由于版本不同而造成软件开发的失误,从而在一定程度上提高工作的效率。

Auto LISP 的函数参考资料简明扼要地描述并提供了每一个 Auto LISP 函数的例子。如果想查阅一下某一个函数的语法约定或者想弄清楚某个函数的返回值,可以翻一翻 Auto LISP 函数参考资料。

由于编写时间较为仓促,难免有错误及不足之处,敬请用户批评指正,并希望用户能与笔者进行交流,共同探讨 AutoCAD 2000 的各项技术。

本书由中科辅龙计算机技术有限公司策划,阎聚群编写,参加编写工作的还有张志利、余刚、刘冲、钱坤、李文江、单雨、覃旭文、庞劭勋等。全书由王艳燕统稿,徐平校排。另外,郭美山、石利文、杨桂莲、郑红、魏红、刘小华等做了部分审校工作。在此一并表示感谢。

由于时间仓促、作者水平有限,本书错漏之处在所难免,欢迎广大用户批评指正。

史惠康

1999 年秋于中科院计算所

目 录

上篇 Visual LISP 的使用

第一章 从 Visual LISP 界面开始	(2)
1.1 启动 Visual LISP	(2)
1.2 关于 Visual LISP 的界面	(4)
1.3 浏览 Visual LISP 的菜单	(8)
1.4 关于 Visual LISP 的控制窗口	(9)
1.5 关于 Visual LISP 的文本编辑器	(10)
1.6 装载并运行一个程序	(11)
1.7 退出 Visual LISP 系统	(13)
第二章 Visual LISP 的各种开发工具	(14)
2.1 开发软件所要注意的几个问题	(14)
2.2 使用控制窗口	(14)
2.2.1 重复先前的操作	(14)
2.2.2 中断现行的操作	(15)
2.2.3 控制窗口的快捷菜单	(17)
2.2.4 控制窗口的日志文件	(18)
2.3 使用文本编辑器	(19)
2.3.1 编辑一个文件	(20)
2.3.2 使用文本编辑器的快捷菜单	(22)
2.3.3 使用键盘快捷菜单	(23)
2.3.4 移动、粘贴文本	(24)
2.3.5 查找指定文本	(25)
2.3.6 书签的使用	(27)
2.4 使用 VLISP 的代码帮助	(29)
2.4.1 Visual LISP 的彩色代码	(29)
2.4.2 使用 Apropos 功能	(30)
2.4.3 Visual LISP 自动完善不完整单词	(39)
2.4.4 Auto LISP 的函数帮助功能	(42)
2.5 使用 VLISP 代码格式化工具	(42)

2.5.1	Visual LISP 格式化选项简介	(43)
2.5.4	保存恢复格式化选项	(50)
2.5.5	格式化的约束	(51)
2.5.6	格式化的快捷操作	(52)
2.6	检查代码语法错误	(53)
2.6.1	检查括号的平衡	(53)
2.6.2	使用彩色代码检查语法错误	(55)
2.6.3	使用核对命令寻找语法错误	(56)
第三章	程序的调试	(58)
3.1	Visual LISP 调试功能简介	(58)
3.2	通过例子来学习程序调试	(59)
3.2.1	在程序中设置断点	(59)
3.2.2	按步执行程序	(60)
3.2.3	监测一个表达式的值	(60)
3.2.4	继续程序的执行	(61)
3.2.5	实时执行模式(Animate Mode)	(61)
3.3	使用 Visual LISP 的调试选项	(63)
3.3.1	开始调试过程	(64)
3.3.2	关于中断循环	(64)
3.3.3	使用断点	(66)
3.4	使用 Visual LISP 的数据检查工具	(69)
3.4.1	使用观察窗	(70)
3.4.2	关于堆栈跟踪窗口	(72)
3.4.3	使用符号服务(Symbol Service)对话框	(79)
3.4.4	使用检查(Inspect)窗口	(79)
3.4.5	浏览 AutoCAD 所画的实体	(80)
第四章	创建应用程序	(84)
4.1	编译链接程序	(84)
4.1.1	使用编译器	(84)
4.1.2	编译一个文件	(84)
4.1.3	预排编译例子	(86)
4.1.4	装载并运行一个编译程序	(87)
4.2	建立程序模块	(87)
4.2.1	创建一个新的应用程序	(87)
4.2.2	装载并运行一个 VLISP 应用程序	(91)
4.2.3	更改应用程序的选项	(91)
4.2.4	重新建立一个应用程序	(93)
4.2.5	更新一个应用程序	(93)

4.3 设计多文件环境.....	(93)
4.3.1 关于作用域空间(Namespace)	(94)
4.3.2 运行在应用程序的作用域空间	(95)
4.3.3 在不同的作用域空间共享数据	(98)
4.3.4 在 MDI 环境中纠错	(99)
第五章 维护 Visual LISP 应用程序	(101)
5.1 管理多个 LISP 文件	(101)
5.2 定义一个项目	(101)
5.2.1 指定项目的各项特征	(102)
5.2.2 使用项目窗口(Project Window)	(104)
5.3 调用已存在的项目	(108)
5.3.1 打开一个项目	(108)
5.3.2 在项目的源文件中查找一个字符串	(109)
5.3.3 在一个应用程序中包含一个项目	(110)
5.4 优化程序代码	(112)
5.4.1 设置项目编译选项	(112)
5.4.2 选择项目编译模式	(113)
5.4.3 选择项目链接模式	(114)
第六章 ActiveX 应用	(115)
6.1 在 AutoLISP 中使用 ActiveX 对象	(115)
6.2 AutoCAD 对象模型	(115)
6.2.1 对象属性	(116)
6.2.2 对象方法	(116)
6.2.3 对象集	(116)
6.3 访问 AutoCAD 对象	(116)
6.3.1 用 inspect 工具查看对象属性	(116)
6.3.2 应用对象的文本格式	(119)
6.3.3 过程讲解	(119)
6.3.4 程序调用应注意的事项	(120)
6.4 应用 ActiveX 技术调用 Visual LISP 函数	(121)
6.4.1 确定所需要的 Visual LISP 函数	(122)
6.4.2 确定怎样调用函数	(123)
6.4.3 将 AutoLISP 数据类型转换为 ActiveX 数据类型	(124)
6.4.4 显示和更新对象的属性	(128)
6.4.5 用 ActiveX 方法返回参量	(130)
6.4.6 对象属性方法的列表	(131)
6.4.7 处理集合对象	(133)
6.4.8 释放对象与内存	(136)

6.4.9 转换对象参考	(136)
6.4.10 处理 ActiveX 方法返回的错误	(137)
6.5 使用 ActiveX 与其他应用程序通信	(139)

下篇 AutoLISP 基础和函数列表

第七章 AutoLISP 语言基础	(142)
7.1 AutoLISP 的表达式	(142)
7.2 AutoLISP 的数据类型	(143)
7.2.1 整型	(143)
7.2.2 实型	(144)
7.2.3 字符型	(145)
7.2.4 表	(145)
7.2.5 选择集	(145)
7.2.6 实体名称	(146)
7.2.7 VLA 对象	(146)
7.2.8 文件描述符	(146)
7.2.9 符号和变量	(147)
7.3 AutoLISP 的程序文件	(147)
7.3.1 格式化 AutoLISP 代码	(147)
7.3.2 注释行	(148)
7.4 AutoLISP 的变量	(149)
7.4.1 显示变量的值	(150)
7.4.2 nil 变量	(150)
7.4.3 预设变量	(150)
7.5 数字处理功能	(151)
7.6 字符串处理功能	(152)
7.7 基本输出函数	(153)
7.8 表处理功能	(154)
7.8.1 使用表表示坐标	(155)
7.8.2 点对	(156)
7.9 符号和函数处理功能	(157)
7.9.1 使用 defun 定义一个函数	(157)
7.9.2 C: XXX 形式的函数	(158)
7.9.3 函数中的局部变量	(160)
7.9.4 带参量的函数	(161)

7.10 出错处理功能	(162)
第八章 AutoLISP 常见函数列表	(164)
8.1 运算函数	(164)
8.1.1 + 函数	(164)
8.1.2 - 函数	(164)
8.1.3 * 函数	(165)
8.1.4 / 函数	(166)
8.1.5 = 函数	(166)
8.1.6 /= 函数	(167)
8.1.7 < 函数	(167)
8.1.8 <= 函数	(168)
8.1.9 > 函数	(168)
8.1.10 >= 函数	(169)
8.1.11 ~ 函数	(169)
8.1.12 1+ 函数	(170)
8.1.13 1- 函数	(170)
8.2 A	(171)
8.2.1 Abs 函数	(171)
8.2.2 acad_colordlg 函数	(171)
8.2.3 acad_strlsort 函数	(172)
8.2.4 action_tile 函数	(172)
8.2.5 add_list 函数	(173)
8.2.6 alert 函数	(173)
8.2.7 alloc 函数	(174)
8.2.8 and 函数	(175)
8.2.9 angle 函数	(175)
8.2.10 angtof 函数	(176)
8.2.11 angtos 函数	(177)
8.2.12 append 函数	(177)
8.2.13 apply 函数	(178)
8.2.14 arx 函数	(179)
8.2.15 arxload 函数	(179)
8.2.16 arxunload 函数	(180)
8.2.17 ascii 函数	(180)
8.2.18 assoc 函数	(181)
8.2.19 atan 函数	(181)
8.2.20 atof 函数	(182)
8.2.21 atoi 函数	(182)

8.2.22	atom 函数	(182)
8.2.23	atoms-family 函数	(183)
8.2.24	autoarxload 函数	(184)
8.2.25	autoload 函数	(184)
8.3	B	(184)
8.4	C	(185)
8.4.1	Car 函数	(185)
8.4.2	Cdr 函数	(186)
8.4.3	Chr 函数	(186)
8.4.4	Close 函数	(187)
8.4.5	Command 函数	(187)
8.4.6	Cond 函数	(188)
8.4.7	Cons 函数	(188)
8.4.8	Cos 函数	(189)
8.4.9	Cvunit 函数	(189)
8.5	D	(190)
8.5.1	dimx_tile 函数	(190)
8.5.2	dimy_tile 函数	(190)
8.5.3	Distance 函数	(190)
8.5.4	Distof 函数	(191)
8.5.5	done_dialog 函数	(192)
8.6	E	(192)
8.6.1	end_image 函数	(192)
8.6.2	end_list 函数	(192)
8.6.3	Entdel 函数	(192)
8.6.4	Entget 函数	(193)
8.6.5	Entlast 函数	(193)
8.6.6	Entmake 函数	(194)
8.6.7	Entmod 函数	(194)
8.6.8	Entnext 函数	(195)
8.6.9	Entsel 函数	(195)
8.6.10	Entupd 函数	(196)
8.6.11	*error* 函数	(196)
8.6.12	Eval 函数	(197)
8.6.13	Exit 函数	(197)
8.6.14	Exp 函数	(197)
8.6.15	Expand 函数	(198)
8.6.16	Expt 函数	(198)

8.7	F	(199)
8.7.1	fill_image 函数	(199)
8.7.2	Findfile 函数	(199)
8.7.3	Fix 函数	(200)
8.7.4	Float 函数	(200)
8.7.5	Foreach 函数	(201)
8.8	G	(201)
8.8.1	Gcd 函数	(201)
8.8.2	get_attr 函数	(202)
8.8.3	get_tile 函数	(202)
8.8.4	Getangle 函数	(202)
8.8.5	Getcorner 函数	(203)
8.8.6	Getdist 函数	(203)
8.8.7	Getenv 函数	(204)
8.8.8	Getfiled 函数	(204)
8.8.9	Getint 函数	(205)
8.8.10	Getkword 函数	(206)
8.8.11	Getorient 函数	(206)
8.8.12	Getpoint 函数	(206)
8.8.13	Getreal 函数	(207)
8.8.14	GetString 函数	(207)
8.8.15	Getvar 函数	(207)
8.8.16	Graphscr 函数	(208)
8.8.17	Grclear 函数	(208)
8.8.18	Grdraw 函数	(208)
8.8.19	Grvecs 函数	(209)
8.8.20	Getcfg 函数	(209)
8.9	H	(210)
8.9.1	Handent 函数	(210)
8.9.2	Help 函数	(210)
8.10	I	(211)
8.10.1	If 函数	(211)
8.10.2	Initdia 函数	(211)
8.10.3	Inters 函数	(212)
8.10.4	Itoa 函数	(213)
8.11	L	(213)
8.11.1	Lambda 函数	(213)
8.11.2	Last 函数	(214)

8.11.3	Layoutlist 函数	(214)
8.11.4	Length 函数	(215)
8.11.5	List 函数	(215)
8.11.6	Listp 函数	(216)
8.11.7	Load 函数	(216)
8.11.8	load_dialog 函数	(217)
8.11.9	Log 函数	(217)
8.11.10	Logand 函数	(217)
8.11.11	Logior 函数	(218)
8.11.12	Lsh 函数	(218)
8.12	M	(219)
8.12.1	Mapcar 函数	(219)
8.12.2	Max 函数	(220)
8.12.3	Mem 函数	(220)
8.12.4	Member 函数	(221)
8.12.5	Menucmd 函数	(221)
8.12.6	Menugroup 函数	(222)
8.12.7	Min 函数	(223)
8.12.8	Minusp 函数	(223)
8.12.9	mode_tile 函数	(224)
8.13	N	(224)
8.13.1	new_dialog 函数	(224)
8.13.2	Not 函数	(225)
8.13.3	Nth 函数	(225)
8.13.4	Null 函数	(226)
8.13.5	Numberp 函数	(226)
8.14	O	(226)
8.14.1	Open 函数	(226)
8.14.2	Or 函数	(227)
8.15	P	(228)
8.15.1	Polar 函数	(228)
8.15.2	prinl 函数	(228)
8.15.3	Princ 函数	(229)
8.15.4	Print 函数	(229)
8.15.5	Progn 函数	(230)
8.15.6	Prompt 函数	(230)
8.16	Q	(230)
8.16.1	Quit 函数	(230)

0017012

8.16.2	Quote 函数	(231)
8.17	R	(231)
8.17.1	Read 函数	(231)
8.17.2	read-char 函数	(232)
8.17.3	read-line 函数	(233)
8.17.4	Redraw 函数	(233)
8.17.5	Rem 函数	(234)
8.17.6	Repeat 函数	(234)
8.17.7	Reverse 函数	(235)
8.18	S	(235)
8.18.1	Set 函数	(235)
8.18.2	set_tile 函数	(235)
8.18.3	Setcfg 函数	(236)
8.18.4	Setenv 函数	(236)
8.18.5	Setfunhelp 函数	(237)
8.18.6	Setq 函数	(238)
8.18.7	Setvar 函数	(238)
8.18.8	Setview 函数	(239)
8.18.9	Sin 函数	(239)
8.18.10	Slide_image 函数	(239)
8.18.11	Snvalid 函数	(240)
8.18.12	Sqrt 函数	(241)
8.18.13	Ssadd 函数	(242)
8.18.14	Ssdel 函数	(242)
8.18.15	Ssget 函数	(243)
8.18.16	Sslength 函数	(244)
8.18.17	Ssmemb 函数	(244)
8.18.18	Ssname 函数	(245)
8.18.19	Ssnamex 函数	(245)
8.18.20	Startapp 函数	(246)
8.18.21	start_dialog 函数	(247)
8.18.22	start_image 函数	(247)
8.18.23	start_list 函数	(247)
8.18.24	Strcase 函数	(248)
8.18.25	Strcat 函数	(248)
8.18.26	Strlen 函数	(249)
8.18.27	Subst 函数	(249)
8.18.28	Substr 函数	(250)

8.19	T	(250)
8.19.1	Tablet 函数	(250)
8.19.2	Tblnext 函数	(251)
8.19.3	Tblobjname 函数	(251)
8.19.4	Tblsearch 函数	(252)
8.19.5	term_dialog 函数	(252)
8.19.6	Terpri 函数	(253)
8.19.7	textbox 函数	(253)
8.19.8	Textpage 函数	(253)
8.19.9	Textscr 函数	(253)
8.19.10	Trace 函数	(254)
8.19.11	Trans 函数	(254)
8.19.12	Type 函数	(255)
8.20	U	(256)
8.20.1	unload_dialog 函数	(256)
8.20.2	untrace 函数	(257)
8.21	V	(257)
8.21.1	vector_image 函数	(257)
8.21.2	Ver 函数	(258)
8.21.3	vl-acad-defun 函数	(258)
8.21.4	vl-acad-undefun 函数	(258)
8.21.5	vl-cmdf 函数	(259)
8.21.6	vl-consp 函数	(259)
8.21.7	vl-directory-files 函数	(260)
8.21.8	vl-every 函数	(261)
8.21.9	vl-exit-with-error 函数	(261)
8.21.10	vl-exit-with-value 函数	(262)
8.21.11	vl-file-copy 函数	(262)
8.21.12	vl-file-delete 函数	(263)
8.21.13	vl-file-directory-p 函数	(263)
8.21.14	vl-file-rename 函数	(264)
8.21.15	vl-file-size 函数	(264)
8.21.16	vl-filename-base 函数	(264)
8.21.17	vl-filename-directory 函数	(265)
8.21.18	vl-filename-extension 函数	(265)
8.21.19	vl-get-resource 函数	(266)
8.21.20	vl-list* 函数	(266)
8.21.21	vl-list-length 函数	(267)

8.21.22	vl-load-all 函数	(267)
8.21.23	vl-member-if 函数	(268)
8.21.24	vl-member-if-not 函数	(268)
8.21.25	vl-position 函数	(269)
8.21.26	vl-prin1-to-string 函数	(269)
8.21.27	vl-princ-to-string 函数	(270)
8.21.28	vl-remove 函数	(270)
8.21.29	vl-remove-if 函数	(271)
8.21.30	vl-remove-if-not 函数	(271)
8.21.31	vl-some 函数	(271)
8.21.32	vl-sort 函数	(272)
8.21.33	vl-sort-I 函数	(272)
8.21.34	vl-symbol-name 函数	(273)
8.21.35	vl-symbol-value 函数	(273)
8.21.36	vl-symbolp 函数	(274)
8.21.37	vl-unload-vlx 函数	(274)
8.21.38	vl-vbaload 函数	(275)
8.21.39	vl-vbarun 函数	(275)
8.21.40	vlax-add-cmd 函数	(275)
8.21.41	vlax-curve-getArea 函数	(276)
8.21.42	vlax-curve-getDistAtParam 函数	(277)
8.21.43	vlax-curve-getDistAtPoint 函数	(277)
8.21.44	vlax-curve-getEndParam 函数	(278)
8.21.45	vlax-curve-getEndPoint 函数	(278)
8.21.46	vlax-curve-getParamAtDist 函数	(279)
8.21.47	vlax-curve-getParamAtPoint 函数	(279)
8.21.48	vlax-curve-getPointAtDist 函数	(279)
8.21.49	vlax-curve-getPointAtParam 函数	(280)
8.21.50	vlax-curve-getStartParam 函数	(280)
8.21.51	vlax-curve-getStartPoint 函数	(281)
8.21.52	vlax-curve-isClosed 函数	(281)
8.21.53	vlax-curve-isPeriodic 函数	(281)
8.21.54	vlax-curve-isPlanar 函数	(282)
8.21.55	vlax-curve-getClosestPointTo 函数	(282)
8.21.56	vlax-ename->vla-object 函数	(283)
8.21.57	vlax-erased-p 函数	(283)
8.21.58	vlax-get-property 函数	(284)
8.21.59	vlax-invoke-method 函数	(284)