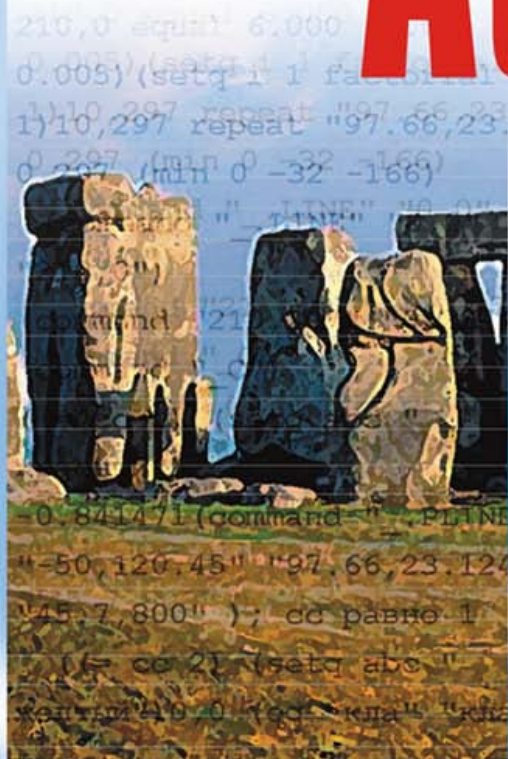


Николай Полещук,
Петр Лоскутов



AutoLISP и Visual LISP в среде **AutoCAD**



**Наиболее
полное
руководство**

+ CD



В ПОДЛИННИКЕ®

УДК 681.3.068
ББК 32.973.26-018.1
П49

Полещук Н. Н., Лоскутов П. В.

П49 AutoLISP и Visual LISP в среде AutoCAD. — СПб.: БХВ-Петербург, 2006. — 960 с.: ил.

ISBN 5-94157-738-9

Книга является руководством разработчика LSP-, FAS- и VLX-приложений, функционирующих в среде графического процессора AutoCAD версий 2005 и 2006. Содержит полную информацию по языкам AutoLISP и Visual LISP. Освещает вопросы COM-связи с внешними приложениями, программной адаптации интерфейса, создания справочных систем. Впервые на русском языке полностью описаны и пояснены на примерах методы и свойства объектной модели AutoCAD, используемой в технологии ActiveX. Исходные тексты рассмотренных программ размещены на сопроводительном компакт-диске.

Для программистов и разработчиков приложений в среде AutoCAD

УДК 681.3.068
ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Наталья Таркова</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Вильга Савельева</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Игоря Цырульникова</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 22.03.06.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 77,4.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Отпечатано с готовых диапозитивов
в ОАО "Техническая книга"

190005, Санкт-Петербург, Измайловский пр., 29

ISBN 5-94157-738-9

© Полещук Н. Н., Лоскутов П. В., 2006
© Оформление, издательство "БХВ-Петербург", 2006

Оглавление

Предисловие	1
Глава 1. Языки AutoLISP и Visual LISP	5
1.1. Интерпретатор LISP	5
1.1.1. Вычисляемые выражения	6
1.1.2. Функция <i>setq</i> и точность отображения	8
1.1.3. Комментарии	9
1.2. Символы	9
1.2.1. Типы данных	10
1.2.2. Предопределенные символы	12
1.2.3. Переменные	12
1.2.4. Функция <i>load</i>	13
1.2.5. Функция <i>vl-load-all</i>	14
1.3. Выполнение команд AutoCAD	14
1.3.1. Функция <i>command</i>	14
1.3.2. Функция <i>vl-cmdf</i>	16
1.4. Арифметические функции	17
1.4.1. Функция +	17
1.4.2. Функция -	18
1.4.3. Функция *	18
1.4.4. Функция /	19
1.4.5. Функция <i>I+</i>	19
1.4.6. Функция <i>I-</i>	20
1.4.7. Функция ~	20
1.4.8. Функция <i>abs</i>	20
1.5. Логические функции	21
1.5.1. Функция =	21
1.5.2. Функция /=	22
1.5.3. Функция <	22
1.5.4. Функция <=	23

1.5.5. Функция $>$	23
1.5.6. Функция \geq	24
1.5.7. Функция <i>and</i>	24
1.5.8. Функция <i>or</i>	24
1.5.9. Функция <i>not</i>	25
1.5.10. Функция <i>null</i>	25
1.5.11. Функция <i>if</i>	26
1.5.12. Функция <i>progn</i>	26
1.5.13. Функция <i>cond</i>	27
1.5.14. Функция <i>minusp</i>	29
1.5.15. Функция <i>zerop</i>	30
1.5.16. Функция <i>numberp</i>	30
1.5.17. Функция <i>listp</i>	30
1.5.18. Функция <i>vl-consp</i>	31
1.5.19. Функция <i>atom</i>	31
1.5.20. Функция <i>vl-symbolp</i>	32
1.5.21. Функция <i>boundp</i>	32
1.5.22. Функция <i>eq</i>	32
1.5.23. Функция <i>equal</i>	33
1.5.24. Функция <i>while</i>	33
1.5.25. Функция <i>repeat</i>	35
1.6. Функции вычислений	35
1.6.1. Функция <i>fix</i>	35
1.6.2. Функция <i>rem</i>	36
1.6.3. Функция <i>max</i>	36
1.6.4. Функция <i>min</i>	37
1.6.5. Функция <i>gcd</i>	37
1.6.6. Функция <i>angle</i>	37
1.6.7. Функция <i>distance</i>	38
1.6.8. Функция <i>sqrt</i>	38
1.6.9. Функция <i>sin</i>	38
1.6.10. Функция <i>cos</i>	39
1.6.11. Функция <i>atan</i>	39
1.6.12. Функция <i>exp</i>	39
1.6.13. Функция <i>expt</i>	40
1.6.14. Функция <i>log</i>	40
1.6.15. Функция <i>logand</i>	40
1.6.16. Функция <i>logior</i>	41
1.6.17. Функция <i>lsh</i>	41
1.6.18. Функция <i>boole</i>	42
1.6.19. Функция <i>inters</i>	43
1.6.20. Функция <i>polar</i>	44
1.6.21. Функция <i>textbox</i>	44
1.6.22. Функция <i>vports</i>	44
1.6.23. Функция <i>vl-symbol-value</i>	47
1.6.24. Функция <i>set</i>	47

1.7. Функции преобразования данных.....	47
1.7.1. Функция <i>type</i>	47
1.7.2. Функция <i>float</i>	48
1.7.3. Функция <i>itoa</i>	49
1.7.4. Функция <i>rtos</i>	49
1.7.5. Функция <i>atoi</i>	50
1.7.6. Функция <i>atof</i>	51
1.7.7. Функция <i>angtof</i>	51
1.7.8. Функция <i>angtos</i>	52
1.7.9. Функция <i>distof</i>	52
1.7.10. Функция <i>trans</i>	53
1.7.11. Функция <i>cvunit</i>	55
1.7.12. Сообщения о неправильном типе	55
1.8. Функции обработки строк	56
1.8.1. Функция <i>chr</i>	56
1.8.2. Функция <i>ascii</i>	57
1.8.3. Функция <i>read</i>	57
1.8.4. Функция <i>strcase</i>	58
1.8.5. Функция <i>strcat</i>	59
1.8.6. Функция <i>strlen</i>	59
1.8.7. Функция <i>substr</i>	59
1.8.8. Функция <i>wcmatch</i>	60
1.8.9. Функция <i>vl-string->list</i>	61
1.8.10. Функция <i>vl-list->string</i>	62
1.8.11. Функция <i>vl-string-elt</i>	62
1.8.12. Функция <i>vl-string-left-trim</i>	62
1.8.13. Функция <i>vl-string-right-trim</i>	63
1.8.14. Функция <i>vl-string-trim</i>	63
1.8.15. Функция <i>vl-string-mismatch</i>	63
1.8.16. Функция <i>vl-string-position</i>	64
1.8.17. Функция <i>vl-string-search</i>	64
1.8.18. Функция <i>vl-string-subst</i>	65
1.8.19. Функция <i>vl-string-translate</i>	65
1.8.20. Функция <i>vl-symbol-name</i>	66
1.9. Функции обработки списков и точечных пар.....	66
1.9.1. Функция <i>list</i>	66
1.9.2. Функция <i>append</i>	67
1.9.3. Функция <i>nth</i>	67
1.9.4. Функция <i>reverse</i>	68
1.9.5. Функция <i>car</i>	68
1.9.6. Функция <i>cdr</i>	69
1.9.7. Функции <i>caar</i> , <i>cadr</i> и аналогичные им.....	69
1.9.8. Функция <i>cons</i>	70
1.9.9. Функция <i>vl-list*</i>	71
1.9.10. Функция <i>member</i>	71

1.9.11. Функция <i>assoc</i>	72
1.9.12. Функция <i>apply</i>	72
1.9.13. Функция <i>mapcar</i>	73
1.9.14. Функция <i>vl-every</i>	73
1.9.15. Функция <i>foreach</i>	74
1.9.16. Функция <i>eval</i>	74
1.9.17. Функция <i>quote</i>	75
1.9.18. Функция <i>acad_strlsort</i>	75
1.9.19. Функция <i>last</i>	75
1.9.20. Функция <i>subst</i>	76
1.9.21. Функция <i>length</i>	76
1.9.22. Функция <i>vl-list-length</i>	76
1.9.23. Функция <i>vl-member-if</i>	77
1.9.24. Функция <i>vl-member-if-not</i>	77
1.9.25. Функция <i>vl-position</i>	78
1.9.26. Функция <i>vl-remove</i>	78
1.9.27. Функция <i>vl-remove-if</i>	78
1.9.28. Функция <i>vl-remove-if-not</i>	79
1.9.29. Функция <i>vl-some</i>	79
1.9.30. Функция <i>vl-sort</i>	80
1.9.31. Функция <i>vl-sort-i</i>	81
1.9.32. Функция <i>vl-catch-all-apply</i> и обработка ошибок	81
1.9.33. Функция <i>vl-catch-all-error-message</i>	82
1.9.34. Функция <i>vl-catch-all-error-p</i>	82
1.10. Функции ввода данных и указания объектов	83
1.10.1. Функция <i>initget</i>	83
1.10.2. Функция <i>getkword</i>	85
1.10.3. Функция <i>getint</i>	86
1.10.4. Функция <i>getreal</i>	87
1.10.5. Функция <i>getdist</i>	87
1.10.6. Функция <i>getangle</i>	88
1.10.7. Функция <i>getorient</i>	89
1.10.8. Функция <i>getpoint</i>	89
1.10.9. Функция <i>getcorner</i>	89
1.10.10. Функция <i>getstring</i>	90
1.10.11. Функция <i>getvar</i>	90
1.10.12. Функция <i>setvar</i>	91
1.10.13. Функция <i>getenv</i>	91
1.10.14. Функция <i>setenv</i>	92
1.10.15. Функция <i>getcfg</i>	93
1.10.16. Функция <i>setcfg</i>	93
1.10.17. Функция <i>getcname</i>	93
1.10.18. Функция <i>getfiled</i>	94
1.10.19. Функция <i>acad_colorldg</i>	96
1.10.20. Функция <i>acad_truecolorldg</i>	97
1.10.21. Функция <i>acad_truecolorcli</i>	99

1.11. Функции печати, вывода сообщений и доступа к файлам.....	99
1.11.1. Функция <i>findfile</i>	99
1.11.2. Функция <i>open</i>	100
1.11.3. Функция <i>close</i>	100
1.11.4. Функция <i>read-line</i>	100
1.11.5. Функция <i>write-line</i>	101
1.11.6. Функция <i>princ</i>	101
1.11.7. Функция <i>prinl</i>	102
1.11.8. Функция <i>print</i>	102
1.11.9. Функция <i>read-char</i>	102
1.11.10. Функция <i>write-char</i>	103
1.11.11. Функция <i>prompt</i>	103
1.11.12. Функция <i>alert</i>	103
1.11.13. Функция <i>terpri</i>	104
1.11.14. Функция <i>vl-directory-files</i>	104
1.11.15. Функция <i>vl-file-copy</i>	105
1.11.16. Функция <i>vl-file-delete</i>	106
1.11.17. Функция <i>vl-file-directory-p</i>	106
1.11.18. Функция <i>vl-file-rename</i>	107
1.11.19. Функция <i>vl-file-size</i>	107
1.11.20. Функция <i>vl-file-systime</i>	107
1.11.21. Функция <i>vl-filename-base</i>	108
1.11.22. Функция <i>vl-filename-directory</i>	108
1.11.23. Функция <i>vl-filename-extension</i>	108
1.11.24. Функция <i>vl-filename-mktemp</i>	109
1.11.25. Функция <i>vl-mkdir</i>	110
1.11.26. Функция <i>vl-princ-to-string</i>	110
1.11.27. Функция <i>vl-prinl-to-string</i>	111
1.11.28. Пример работы с файлами.....	111
1.12. Функции доступа к примитивам AutoCAD.....	112
1.12.1. Функция <i>entlast</i>	113
1.12.2. Функция <i>entnext</i>	114
1.12.3. Функция <i>entsel</i>	115
1.12.4. Функция <i>entdel</i>	115
1.12.5. Функция <i>entget</i>	116
1.12.6. Функция <i>entmake</i>	117
1.12.7. Функция <i>entmakex</i>	118
1.12.8. Функция <i>entmod</i>	118
1.12.9. Функция <i>entupd</i>	119
1.12.10. Функция <i>handent</i>	119
1.12.11. Функция <i>nentsel</i>	120
1.12.12. Функция <i>nentselp</i>	121
1.12.13. Наборы.....	121
1.12.14. Функция <i>ssget</i>	122
1.12.14.1. Фильтры со сложными условиями.....	124

1.12.15. Функция <i>ssadd</i>	126
1.12.16. Функция <i>ssdel</i>	126
1.12.17. Функция <i>sslength</i>	127
1.12.18. Функция <i>ssmemb</i>	127
1.12.19. Функция <i>ssname</i>	127
1.12.20. Функция <i>ssnamex</i>	128
1.12.21. Функция <i>sssetfirst</i>	130
1.12.22. Функция <i>ssgetfirst</i>	131
1.12.23. Пример работы с примитивами AutoCAD.....	131
1.13. Функции доступа к табличным данным.....	133
1.13.1. Символьные таблицы.....	133
1.13.2. Функция <i>tblsearch</i>	134
1.13.3. Функция <i>tblnext</i>	134
1.13.4. Функция <i>tblobjname</i>	135
1.13.5. Схема работы с таблицами.....	136
1.13.6. Функция <i>setview</i>	136
1.13.7. Функция <i>svinvalid</i>	138
1.14. Функции, работающие с расширенными данными.....	139
1.14.1. Структура расширенных данных.....	139
1.14.2. Функция <i>regapp</i>	141
1.14.3. Присоединение расширенных данных.....	141
1.14.4. Функция <i>xdsize</i>	143
1.14.5. Функция <i>xdroom</i>	143
1.14.5.1. О примитиве VIEWPORT.....	143
1.15. Функции, работающие со словарями и X-записями.....	144
1.15.1. Словари.....	144
1.15.2. Функция <i>namedobjdict</i>	145
1.15.3. Функция <i>dictsearch</i>	146
1.15.4. Функция <i>dictnext</i>	147
1.15.5. Функция <i>dictadd</i>	149
1.15.6. Функция <i>dictremove</i>	150
1.15.7. Функция <i>dictrename</i>	151
1.15.8. Функция <i>layoutlist</i>	151
1.15.9. X-записи и наполнение словарей.....	152
1.16. Разработка функций пользователя.....	154
1.16.1. Функция <i>lambda</i>	154
1.16.2. Функция <i>defun</i>	155
1.16.3. Создание дополнительных команд системы AutoCAD с помощью пользовательских функций.....	158
1.16.4. Функция <i>defun-q</i>	159
1.16.5. Функция <i>defun-q-list-ref</i>	160
1.16.6. Функция <i>defun-q-list-set</i>	161
1.16.7. Функция <i>function</i>	161
1.16.8. Функция <i>vlisp-compile</i>	162
1.16.9. Функция <i>exit</i>	162

1.16.10. Функция <i>quit</i>	163
1.16.11. Функция <i>*error*</i>	163
1.16.12. Функция <i>vl-acad-defun</i>	164
1.16.13. Функция <i>vl-acad-undefun</i>	164
1.16.14. Функция <i>vl-exit-with-error</i>	165
1.16.15. Функция <i>vl-exit-with-value</i>	165
1.16.16. Функция <i>atoms-family</i>	165
1.16.17. Функция <i>trace</i>	166
1.16.18. Функция <i>untrace</i>	167
1.17. Функции работы с памятью, приложениями и реестром Windows	167
1.17.1. Функция <i>ver</i>	167
1.17.2. Функция <i>mem</i>	167
1.17.3. Функция <i>alloc</i>	168
1.17.4. Функция <i>expand</i>	168
1.17.5. Функция <i>gc</i>	169
1.17.6. Функция <i>vl-registry-read</i>	169
1.17.7. Функция <i>vl-registry-write</i>	169
1.17.8. Функция <i>vl-registry-delete</i>	170
1.17.9. Функция <i>vl-registry-descendents</i>	170
1.17.10. Функция <i>arx</i>	171
1.17.11. Функция <i>arxload</i>	171
1.17.12. Функция <i>arxunload</i>	171
1.17.13. Функция <i>autoload</i>	172
1.17.14. Функция <i>autoarxload</i>	172
1.17.15. Функция <i>vl-arx-import</i>	173
1.17.16. Функция <i>vl-doc-export</i>	173
1.17.17. Функция <i>vl-doc-import</i>	174
1.17.18. Функция <i>vl-list-exported-functions</i>	174
1.17.19. Функция <i>vl-list-loaded-vlx</i>	175
1.17.20. Функция <i>vl-vlx-loaded-p</i>	176
1.17.21. Функция <i>vl-unload-vlx</i>	176
1.17.22. Функция <i>vl-get-resource</i>	177
1.17.23. Функция <i>vl-vbaload</i>	177
1.17.24. Функция <i>vl-vbarun</i>	177
1.17.25. Функция <i>vl-bb-set</i>	178
1.17.26. Функция <i>vl-bb-ref</i>	178
1.17.27. Функция <i>vl-doc-set</i>	179
1.17.28. Функция <i>vl-doc-ref</i>	179
1.17.29. Функция <i>vl-load-all</i>	179
1.17.30. Функция <i>vl-propagate</i>	180
1.17.31. Глобальные и локальные переменные	180
1.17.32. Пространства имен.....	183
1.17.33. Проблемы многодокументного режима	184
1.17.34. Функция <i>startapp</i>	185

1.18. Другие функции	185
1.18.1. Функция <i>osnap</i>	185
1.18.2. Функция <i>redraw</i>	186
1.18.3. Функция <i>graphscr</i>	187
1.18.4. Функция <i>textscr</i>	187
1.18.5. Функция <i>textpage</i>	187
1.18.6. Функция <i>grtext</i>	187
1.18.7. Функция <i>grdraw</i>	190
1.18.8. Функция <i>grvecs</i>	190
1.18.9. Функция <i>grclear</i>	191
1.18.10. Функция <i>gread</i>	191
1.18.11. Функция <i>initdia</i>	194
1.18.12. Функция <i>tablet</i>	195
1.18.13. Функция <i>acad-push-dbmod</i>	196
1.18.14. Функция <i>acad-pop-dbmod</i>	196
1.18.15. Функция <i>acditableupdate</i>	196
1.18.16. Функция <i>acet-layerp-mode</i>	197
1.18.17. Функция <i>acet-layerp-mark</i>	197
1.18.18. Функция <i>acet-laytrans</i>	198
1.18.19. Функция <i>acet-ms-to-ps</i>	198
1.18.20. Функция <i>acet-ps-to-ms</i>	199
1.19. Файлы автоматической загрузки функций пользователя	199
1.19.1. Файл <i>acad.lsp</i>	200
1.19.2. Файл <i>acaddoc.lsp</i>	201
1.19.3. Системная переменная <i>ACADLSPASDOC</i>	201
1.19.4. Функция <i>s::startup</i>	201
Глава 2. Среда разработки приложений.....	202
2.1. Обзор меню.....	202
2.2. Панели инструментов Visual LISP	221
2.3. Написание и проверка синтаксиса программ в текстовом редакторе среды Visual LISP.....	222
2.4. Разработка приложений на языке LISP	223
2.4.1. Средства отладки программ	223
2.4.2. Проекты и FAS-приложения	225
2.4.3. VLX-приложения.....	228
Глава 3. Диалоговые окна	234
3.1. Язык DCL.....	235
3.1.1. Директивы	235
3.1.2. Пример диалогового окна.....	237
3.1.3. Директива описания диалога.....	238
3.2. Открытие и закрытие диалоговых окон	242
3.2.1. Функция <i>load_dialog</i>	242
3.2.2. Функция <i>unload_dialog</i>	242

3.2.3. Функция <i>new_dialog</i>	243
3.2.4. Функция <i>start_dialog</i>	243
3.2.5. Функция <i>done_dialog</i>	244
3.2.6. Функция <i>term_dialog</i>	244
3.2.7. Пример вызова диалогового окна	244
3.3. Элементы диалога	245
3.3.1. Типы элементов	245
3.3.2. Элементы с кнопкой выхода	246
3.3.3. Атрибуты элементов	247
3.3.4. Примеры элементов	251
3.3.4.1. Кнопка	251
3.3.4.2. Текстовые поля	252
3.3.4.3. Элементы с изображениями	252
3.3.4.4. Элементы со списками	253
3.3.4.5. Переключатели	254
3.3.4.6. Скользящая шкала	256
3.3.4.7. Флажок	257
3.3.5. Группировка полей	257
3.3.6. Элементы надписей	260
3.3.7. Элементы промежутков	261
3.4. Функции, работающие с элементами диалога	263
3.4.1. Функция <i>action_tile</i>	263
3.4.2. Функция <i>mode_tile</i>	264
3.4.3. Функция <i>set_tile</i>	265
3.4.4. Функция <i>get_tile</i>	265
3.4.5. Функция <i>get_attr</i>	265
3.4.6. Функция <i>client_data_tile</i>	266
3.4.7. Переменные действия	266
3.4.8. Пример	267
3.5. Функции, работающие со списками	272
3.5.1. Функция <i>start_list</i>	272
3.5.2. Функция <i>add_list</i>	273
3.5.3. Функция <i>end_list</i>	273
3.5.4. Пример окна со списками	273
3.5.5. Ограничения списков в DCL-файлах	274
3.5.6. Особенности применения функций <i>set_tile</i> и <i>get_tile</i> для списков	275
3.6. Функции, работающие с графическими элементами	276
3.6.1. Функция <i>start_image</i>	276
3.6.2. Функция <i>end_image</i>	277
3.6.3. Функция <i>fill_image</i>	277
3.6.4. Функция <i>slide_image</i>	278
3.6.5. Функция <i>vector_image</i>	278
3.6.6. Функция <i>dimx_tile</i>	279
3.6.7. Функция <i>dimy_tile</i>	279

3.6.8. Примеры окон с графическими элементами	279
3.6.8.1. Просмотр изображений из библиотеки слайдов	279
3.6.8.2. Изменение цветов графической кнопки.....	283
3.6.8.3. Изменение рисунка, наносимого на кнопку	285
3.7. Дополнительные приемы программирования диалоговых окон	289
3.7.1. Временный выход из диалогового окна.....	289
3.7.2. Вложенные диалоговые окна	291
3.7.3. Применение паролей	293
Глава 4. Меню и адаптации	294
4.1. Коротко о меню и адаптациях.....	294
4.1.1. Группы меню и группы адаптации	295
4.1.2. Разделы меню (адаптации)	295
4.2. Структура заголовков и пунктов меню	296
4.3. Функции LISP, работающие с меню.....	296
4.3.1. Функция <i>menugroup</i>	296
4.3.2. Функция <i>menustd</i>	297
4.3.3. Операции загрузки (выгрузки) групп	298
4.4. Падающие меню.....	298
4.4.1. Добавление в строку меню	299
4.4.2. Удаление из строки меню	299
4.4.3. Вывод содержимого меню на экран.....	300
4.4.4. Отметка пункта меню.....	300
4.4.5. Отключение пункта меню.....	301
4.4.6. Восстановление пункта меню	301
4.4.7. Замена меню.....	301
4.4.8. Справка о состоянии пункта меню	302
4.5. Экранные меню	302
4.6. Графические меню	303
4.7. Панели инструментов	303
4.8. Контекстные меню.....	304
4.8.1. Разделы с контекстными меню	304
4.8.2. Схема работы контекстных меню.....	304
4.9. MNL-файлы	306
4.10. ActiveX	308
Глава 5. Справочные системы	309
5.1. HELP-файлы Windows.....	309
5.1.1. Структура HELP-файлов.....	309
5.1.2. CNT-файлы	311
5.1.3. Функция <i>help</i>	312
5.1.4. Вызов WIN-справки с помощью функции <i>startapp</i>	313
5.2. HTML-справки	313
5.2.1. Вызов HTML-справки с помощью функции <i>help</i>	313
5.2.2. Вызов HTML-справки с помощью функции <i>startapp</i>	313

5.3. Устаревшие справочные файлы AutoCAD.....	314
5.3.1. Функция <i>acad_helpdlg</i>	314
5.4. Контекстно-зависимая справка к командам пользователя	315
5.4.1. Функция <i>setfunhelp</i>	315
5.4.2. Особенности применения функции <i>setfunhelp</i>	315

Глава 6. Технология ActiveX.....316

6.1. Расширение языка AutoLISP	316
6.1.1. Объекты и семейства.....	316
6.1.2. Функция <i>vl-load-com</i>	323
6.1.3. Свойства.....	324
6.1.4. Функция <i>vla-get-acad-object</i>	324
6.1.5. Функция <i>vla-get-ActiveDocument</i>	324
6.1.6. Функция <i>vla-get-ModelSpace</i>	325
6.1.7. Функция <i>vla-get-PaperSpace</i>	325
6.1.8. Подготовка к работе с ActiveX.....	325
6.1.9. Методы.....	326
6.1.10. VLA-объекты.....	326
6.1.11. Функция <i>vla-ename->vla-object</i>	327
6.1.12. Функция <i>vla-vla-object->ename</i>	327
6.1.13. Работа со свойствами	327
6.1.14. Варианты	328
6.1.15. Функция <i>vla-make-variant</i>	328
6.1.16. Функция <i>vla-variant-type</i>	329
6.1.17. Функция <i>vla-variant-value</i>	330
6.1.18. Функция <i>vla-variant-change-type</i>	330
6.1.19. Функция <i>vla-3D-point</i>	330
6.1.20. Безопасные массивы.....	331
6.1.21. Функция <i>vla-make-safearray</i>	331
6.1.22. Функция <i>vla-safearray-put-element</i>	332
6.1.23. Функция <i>vla-safearray-fill</i>	332
6.1.24. Пример создания VLA-объекта	333
6.1.25. Функция <i>vla-dump-object</i>	333
6.1.26. Другие функции ActiveX	335
6.2. Функции команд и методов.....	335
6.2.1. Функция <i>vla-add-cmd</i>	335
6.2.2. Функция <i>vla-remove-cmd</i>	336
6.2.3. Функция <i>vla-invoke-method</i>	337
6.2.4. Функция <i>vla-product-key</i>	337
6.3. Функции операций с объектами	337
6.3.1. Функция <i>vla-erased-p</i>	338
6.3.2. Функция <i>vla-method-applicable-p</i>	338
6.3.3. Функция <i>vla-typeinfo-available-p</i>	338
6.3.4. Функция <i>vla-object-released-p</i>	338

6.3.5. Функция <i>vlox-release-object</i>	339
6.3.6. Функция <i>vlox-read-enabled-p</i>	339
6.3.7. Функция <i>vlox-write-enabled-p</i>	339
6.3.8. Функция <i>vlox-create-object</i>	339
6.3.9. Функция <i>vlox-get-object</i>	340
6.3.10. Функция <i>vlox-get-or-create-object</i>	340
6.3.11. Функция <i>vlox-import-type-library</i>	340
6.3.12. Функция <i>vlox-map-collection</i>	341
6.3.13. Функция <i>vlox-for</i>	341
6.4. Функции операций со словарями	342
6.4.1. Функция <i>vlox-ldata-put</i>	342
6.4.2. Функция <i>vlox-ldata-get</i>	343
6.4.3. Функция <i>vlox-ldata-list</i>	343
6.4.4. Функция <i>vlox-ldata-delete</i>	343
6.4.5. Функция <i>vlox-ldata-test</i>	344
6.4.6. Словарь Extension Dictionary	344
6.4.6.1. Словарь фильтров слоев	344
6.4.6.2. Объект VLO-VL	346
6.5. Функции преобразований	348
6.5.1. Функция <i>vlox-safearray-get-dim</i>	348
6.5.2. Функция <i>vlox-safearray-get-l-bound</i>	348
6.5.3. Функция <i>vlox-safearray-get-u-bound</i>	348
6.5.4. Функция <i>vlox-safearray-get-element</i>	349
6.5.5. Функция <i>vlox-safearray-type</i>	349
6.5.6. Функция <i>vlox-safearray->list</i>	349
6.5.7. Функция <i>vlox-tmatrix</i>	349
6.6. Функции операций со свойствами	350
6.6.1. Функция <i>vlox-get-property</i>	350
6.6.2. Функция <i>vlox-put-property</i>	350
6.6.3. Функция <i>vlox-property-available-p</i>	350
6.7. Функции операций с кривыми	351
6.7.1. Функция <i>vlox-curve-getArea</i>	351
6.7.2. Функция <i>vlox-curve-getStartPoint</i>	351
6.7.3. Функция <i>vlox-curve-getEndPoint</i>	351
6.7.4. Функция <i>vlox-curve-getPointAtDist</i>	351
6.7.5. Функция <i>vlox-curve-getDistAtPoint</i>	352
6.7.6. Функция <i>vlox-curve-getStartParam</i>	352
6.7.7. Функция <i>vlox-curve-getEndParam</i>	352
6.7.8. Функция <i>vlox-curve-getParamAtPoint</i>	352
6.7.9. Функция <i>vlox-curve-getPointAtParam</i>	353
6.7.10. Функция <i>vlox-curve-getParamAtDist</i>	353
6.7.11. Функция <i>vlox-curve-getDistAtParam</i>	353
6.7.12. Функция <i>vlox-curve-isClosed</i>	353
6.7.13. Функция <i>vlox-curve-isPeriodic</i>	354
6.7.14. Функция <i>vlox-curve-isPlanar</i>	354

6.7.15. Функция <i>vla</i> - <i>curve-getClosestPointTo</i>	354
6.7.16. Функция <i>vla</i> - <i>curve-getClosestPointToProjection</i>	354
6.7.17. Функция <i>vla</i> - <i>curve-getFirstDeriv</i>	355
6.7.18. Функция <i>vla</i> - <i>curve-getSecondDeriv</i>	355
6.7.19. Что такое параметр кривой.....	355
6.8. Примеры работы с технологией ActiveX.....	355
6.8.1. Доступ к неактивному документу.....	356
6.8.2. Динамическое изменение контекстного меню.....	356
Глава 7. Методы ActiveX.....	359
7.1. Дополнительные соглашения.....	359
7.2. Функции методов.....	360
7.2.1. Метод <i>Activate</i>	362
7.2.2. Метод <i>Add</i>	362
7.2.3. Метод <i>Add3DFace</i>	365
7.2.4. Метод <i>Add3DMesh</i>	365
7.2.5. Метод <i>Add3DPoly</i>	366
7.2.6. Метод <i>AddArc</i>	366
7.2.7. Метод <i>AddAttribute</i>	367
7.2.8. Метод <i>AddBox</i>	367
7.2.9. Метод <i>AddCircle</i>	368
7.2.10. Метод <i>AddCone</i>	368
7.2.11. Метод <i>AddCustomInfo</i>	368
7.2.12. Метод <i>AddCustomObject</i>	369
7.2.13. Метод <i>AddCylinder</i>	369
7.2.14. Метод <i>AddDim3PointAngular</i>	369
7.2.15. Метод <i>AddDimAligned</i>	370
7.2.16. Метод <i>AddDimAngular</i>	370
7.2.17. Метод <i>AddDimArc</i>	371
7.2.18. Метод <i>AddDimDiametric</i>	371
7.2.19. Метод <i>AddDimOrdinate</i>	371
7.2.20. Метод <i>AddDimRadial</i>	372
7.2.21. Метод <i>AddDimRadialLarge</i>	372
7.2.22. Метод <i>AddDimRotated</i>	373
7.2.23. Метод <i>AddEllipse</i>	373
7.2.24. Метод <i>AddEllipticalCone</i>	374
7.2.25. Метод <i>AddEllipticalCylinder</i>	374
7.2.26. Метод <i>AddExtrudedSolid</i>	374
7.2.27. Метод <i>AddExtrudedSolidAlongPath</i>	375
7.2.28. Метод <i>AddFitPoint</i>	375
7.2.29. Метод <i>AddHatch</i>	376
7.2.30. Метод <i>AddItems</i>	377
7.2.31. Метод <i>AddLeader</i>	377
7.2.32. Метод <i>AddLightWeightPolyline</i>	378
7.2.33. Метод <i>AddLine</i>	378

7.2.34. Метод <i>AddMenuItem</i>	379
7.2.35. Метод <i>AddMInsertBlock</i>	379
7.2.36. Метод <i>AddMLine</i>	380
7.2.37. Метод <i>AddMText</i>	380
7.2.38. Метод <i>AddObject</i>	381
7.2.39. Метод <i>AddPoint</i>	381
7.2.40. Метод <i>AddPolyfaceMesh</i>	382
7.2.41. Метод <i>AddPolyline</i>	382
7.2.42. Метод <i>AddPViewport</i>	383
7.2.43. Метод <i>AddRaster</i>	383
7.2.44. Метод <i>AddRay</i>	384
7.2.45. Метод <i>AddRegion</i>	384
7.2.46. Метод <i>AddRevolvedSolid</i>	385
7.2.47. Метод <i>AddSeparator</i>	385
7.2.48. Метод <i>AddShape</i>	386
7.2.49. Метод <i>AddSolid</i>	386
7.2.50. Метод <i>AddSphere</i>	386
7.2.51. Метод <i>AddSpline</i>	387
7.2.52. Метод <i>AddSubMenu</i>	387
7.2.53. Метод <i>AddTable</i>	388
7.2.54. Метод <i>AddText</i>	388
7.2.55. Метод <i>AddTolerance</i>	389
7.2.56. Метод <i>AddToolBarButton</i>	389
7.2.57. Метод <i>AddTorus</i>	390
7.2.58. Метод <i>AddTrace</i>	390
7.2.59. Метод <i>AddVertex</i>	391
7.2.60. Метод <i>AddWedge</i>	391
7.2.61. Метод <i>AddXline</i>	391
7.2.62. Метод <i>AddXRecord</i>	392
7.2.63. Метод <i>AngleFromXAxis</i>	392
7.2.64. Метод <i>AngleToReal</i>	392
7.2.65. Метод <i>AngleToString</i>	393
7.2.66. Метод <i>AppendInnerLoop</i>	393
7.2.67. Метод <i>AppendItems</i>	394
7.2.68. Метод <i>AppendOuterLoop</i>	394
7.2.69. Метод <i>AppendVertex</i>	395
7.2.70. Метод <i>ArrayPolar</i>	395
7.2.71. Метод <i>ArrayRectangular</i>	396
7.2.72. Метод <i>AttachExternalReference</i>	396
7.2.73. Метод <i>AttachToolBarToFlyout</i>	397
7.2.74. Метод <i>AuditInfo</i>	398
7.2.75. Метод <i>Bind</i>	398
7.2.76. Метод <i>Block</i>	398
7.2.77. Метод <i>Boolean</i>	399
7.2.78. Метод <i>CheckInterference</i>	399

7.2.79. Метод <i>Clear</i>	399
7.2.80. Метод <i>ClearSubSelection</i>	400
7.2.81. Метод <i>ClearTableStyleOverrides</i>	400
7.2.82. Метод <i>ClipBoundary</i>	400
7.2.83. Метод <i>Close</i>	401
7.2.84. Метод <i>Connect</i>	402
7.2.85. Метод <i>ConvertToAnonymousBlock</i>	402
7.2.86. Метод <i>ConvertToStaticBlock</i>	403
7.2.87. Метод <i>Copy</i>	403
7.2.88. Метод <i>CopyFrom</i>	403
7.2.89. Метод <i>CopyObjects</i>	404
7.2.90. Метод <i>CopyProfile</i>	404
7.2.91. Метод <i>CreateEntry</i>	405
7.2.92. Метод <i>CreateLabel</i>	405
7.2.93. Метод <i>CreateLink</i>	406
7.2.94. Метод <i>CreateTypedArray</i>	407
7.2.95. Метод <i>Delete</i>	408
7.2.96. Метод <i>DeleteCellContent</i>	408
7.2.97. Метод <i>DeleteColumns</i>	408
7.2.98. Метод <i>DeleteConfiguration</i>	409
7.2.99. Метод <i>DeleteFitPoint</i>	409
7.2.100. Метод <i>DeleteProfile</i>	409
7.2.101. Метод <i>DeleteRows</i>	410
7.2.102. Метод <i>Detach</i>	410
7.2.103. Метод <i>Disconnect</i>	410
7.2.104. Метод <i>Display</i>	411
7.2.105. Метод <i>DisplayPlotPreview</i>	411
7.2.106. Метод <i>DistanceToReal</i>	411
7.2.107. Метод <i>Dock</i>	412
7.2.108. Метод <i>ElevateOrder</i>	412
7.2.109. Метод <i>EndUndoMark</i>	413
7.2.110. Метод <i>Erase</i>	413
7.2.111. Метод <i>Eval</i>	413
7.2.112. Метод <i>Evaluate</i>	414
7.2.113. Метод <i>Explode</i>	414
7.2.114. Метод <i>Export</i>	414
7.2.115. Метод <i>ExportProfile</i>	415
7.2.116. Метод <i>FieldCode</i>	416
7.2.117. Метод <i>Float</i>	416
7.2.118. Метод <i>GenerateLayout</i>	416
7.2.119. Метод <i>GenerateUsageData</i>	417
7.2.120. Метод <i>GetAcadState</i>	417
7.2.121. Метод <i>GetAlignment</i>	417
7.2.122. Метод <i>GetAllProfileNames</i>	418
7.2.123. Метод <i>GetAngle</i>	418

7.2.124. Метод <i>GetAttachmentPoint</i>	418
7.2.125. Метод <i>GetAttributes</i>	419
7.2.126. Метод <i>GetAutoScale</i>	419
7.2.127. Метод <i>GetBackgroundColor</i>	420
7.2.128. Метод <i>GetBackgroundColorNone</i>	420
7.2.129. Метод <i>GetBitmaps</i>	420
7.2.130. Метод <i>GetBlockAttributeValue</i>	421
7.2.131. Метод <i>GetBlockRotation</i>	421
7.2.132. Метод <i>GetBlockScale</i>	421
7.2.133. Метод <i>GetBlockTableRecordId</i>	422
7.2.134. Метод <i>GetBoundingBox</i>	422
7.2.135. Метод <i>GetBulge</i>	422
7.2.136. Метод <i>GetCanonicalMediaNames</i>	423
7.2.137. Метод <i>GetCellAlignment</i>	423
7.2.138. Метод <i>GetCellBackgroundColor</i>	423
7.2.139. Метод <i>GetCellBackgroundColorNone</i>	424
7.2.140. Метод <i>GetCellContentColor</i>	424
7.2.141. Метод <i>GetCellExtents</i>	424
7.2.142. Метод <i>GetCellGridColor</i>	425
7.2.143. Метод <i>GetCellGridLineWeight</i>	426
7.2.144. Метод <i>GetCellGridVisibility</i>	426
7.2.145. Метод <i>GetCellStyleOverrides</i>	426
7.2.146. Метод <i>GetCellTextHeight</i>	427
7.2.147. Метод <i>GetCellTextStyle</i>	427
7.2.148. Метод <i>GetCellType</i>	427
7.2.149. Метод <i>GetColor</i>	428
7.2.150. Метод <i>GetColumnWidth</i>	428
7.2.151. Метод <i>GetConstantAttributes</i>	428
7.2.152. Метод <i>GetContentColor</i>	429
7.2.153. Метод <i>GetControlPoint</i>	429
7.2.154. Метод <i>GetCorner</i>	430
7.2.155. Метод <i>GetCustomByIndex</i>	430
7.2.156. Метод <i>GetCustomByKey</i>	430
7.2.157. Метод <i>GetCustomScale</i>	431
7.2.158. Метод <i>GetDistance</i>	431
7.2.159. Метод <i>GetDynamicBlockProperties</i>	432
7.2.160. Метод <i>GetEntity</i>	432
7.2.161. Метод <i>GetExtensionDictionary</i>	432
7.2.162. Метод <i>GetFieldId</i>	433
7.2.163. Метод <i>GetFitPoint</i>	433
7.2.164. Метод <i>GetFont</i>	434
7.2.165. Метод <i>GetFullDrawOrder</i>	435
7.2.166. Метод <i>GetGridColor</i>	435
7.2.167. Метод <i>GetGridLineWeight</i>	436
7.2.168. Метод <i>GetGridSpacing</i>	436

7.2.169. Метод <i>GetGridVisibility</i>	437
7.2.170. Метод <i>GetInput</i>	437
7.2.171. Метод <i>GetInteger</i>	437
7.2.172. Метод <i>GetInterfaceObject</i>	438
7.2.173. Метод <i>GetInvisibleEdge</i>	438
7.2.174. Метод <i>GetKeyword</i>	438
7.2.175. Метод <i>GetLabelTemplates</i>	439
7.2.176. Метод <i>GetLinks</i>	439
7.2.177. Метод <i>GetLinksByKey</i>	440
7.2.178. Метод <i>GetLinksByLabelTemplate</i>	441
7.2.179. Метод <i>GetLinksBySQLCond</i>	441
7.2.180. Метод <i>GetLinkTemplates</i>	442
7.2.181. Метод <i>GetLocaleMediaName</i>	442
7.2.182. Метод <i>GetLoopAt</i>	443
7.2.183. Метод <i>GetMinimumColumnWidth</i>	443
7.2.184. Метод <i>GetMinimumRowHeight</i>	443
7.2.185. Метод <i>GetName</i>	444
7.2.186. Метод <i>GetObject</i>	444
7.2.187. Метод <i>GetOrientation</i>	444
7.2.188. Метод <i>GetPaperMargins</i>	445
7.2.189. Метод <i>GetPaperSize</i>	445
7.2.190. Метод <i>GetPlotDeviceNames</i>	446
7.2.191. Метод <i>GetPlotStyleTableNames</i>	446
7.2.192. Метод <i>GetPoint</i>	446
7.2.193. Метод <i>GetProjectFilePath</i>	447
7.2.194. Метод <i>GetQueries</i>	447
7.2.195. Метод <i>GetReal</i>	448
7.2.196. Метод <i>GetRelativeDrawOrder</i>	448
7.2.197. Метод <i>GetRemoteFile</i>	449
7.2.198. Метод <i>GetRowHeight</i>	449
7.2.199. Метод <i>GetRowType</i>	449
7.2.200. Метод <i>GetSnapSpacing</i>	450
7.2.201. Метод <i>GetString</i>	450
7.2.202. Метод <i>GetSubEntity</i>	451
7.2.203. Метод <i>GetSubSelection</i>	451
7.2.204. Метод <i>GetText</i>	452
7.2.205. Метод <i>GetTextHeight</i>	452
7.2.206. Метод <i>GetTextRotation</i>	453
7.2.207. Метод <i>GetTextStyle</i>	453
7.2.208. Метод <i>GetUCSMatrix</i>	453
7.2.209. Метод <i>GetVariable</i>	454
7.2.210. Метод <i>GetWeight</i>	454
7.2.211. Метод <i>GetWidth</i>	455
7.2.212. Метод <i>GetWindowToPlot</i>	455
7.2.213. Метод <i>GetXData</i>	455

7.2.214. Метод <i>GetXRecordData</i>	456
7.2.215. Метод <i>HandleToObject</i>	457
7.2.216. Метод <i>Hide</i>	457
7.2.217. Метод <i>Highlight</i>	457
7.2.218. Метод <i>HitTest</i>	458
7.2.219. Метод <i>Import</i>	458
7.2.220. Метод <i>ImportProfile</i>	459
7.2.221. Метод <i>IndexOf</i>	460
7.2.222. Метод <i>InitializeUserInput</i>	460
7.2.223. Метод <i>InsertBlock</i>	461
7.2.224. Метод <i>InsertColumns</i>	461
7.2.225. Метод <i>InsertInMenuBar</i>	462
7.2.226. Метод <i>InsertLoopAt</i>	462
7.2.227. Метод <i>InsertMenuInMenuBar</i>	463
7.2.228. Метод <i>InsertRows</i>	463
7.2.229. Метод <i>Intersect</i>	464
7.2.230. Метод <i>IntersectWith</i>	464
7.2.231. Метод <i>IsConnected</i>	465
7.2.232. Метод <i>IsMergedCell</i>	465
7.2.233. Метод <i>IsRemoteFile</i>	466
7.2.234. Метод <i>IsURL</i>	466
7.2.235. Метод <i>Item</i>	467
7.2.236. Метод <i>LaunchBrowserDialog</i>	467
7.2.237. Метод <i>ListArx</i>	468
7.2.238. Метод <i>Load</i>	468
7.2.239. Метод <i>LoadArx</i>	469
7.2.240. Метод <i>LoadDVB</i>	469
7.2.241. Метод <i>LoadShapeFile</i>	469
7.2.242. Метод <i>MergeCells</i>	470
7.2.243. Метод <i>Mirror</i>	470
7.2.244. Метод <i>Mirror3D</i>	470
7.2.245. Метод <i>Move</i>	471
7.2.246. Метод <i>MoveAbove</i>	471
7.2.247. Метод <i>MoveBelow</i>	472
7.2.248. Метод <i>MoveToBottom</i>	472
7.2.249. Метод <i>MoveToTop</i>	473
7.2.250. Метод <i>New</i>	473
7.2.251. Метод <i>NumCustomInfo</i>	473
7.2.252. Метод <i>ObjectIDToObject</i>	474
7.2.253. Метод <i>Offset</i>	474
7.2.254. Метод <i>Open</i>	475
7.2.255. Метод <i>PlotToDevice</i>	475
7.2.256. Метод <i>PlotToFile</i>	475
7.2.257. Метод <i>PolarPoint</i>	476
7.2.258. Метод <i>Prompt</i>	476

7.2.259. Метод <i>PurgeAll</i>	477
7.2.260. Метод <i>PurgeFitData</i>	477
7.2.261. Метод <i>PutRemoteFile</i>	477
7.2.262. Метод <i>Quit</i>	477
7.2.263. Метод <i>RealToString</i>	478
7.2.264. Метод <i>RecomputeTableBlock</i>	478
7.2.265. Метод <i>Refresh</i>	478
7.2.266. Метод <i>RefreshPlotDeviceInfo</i>	479
7.2.267. Метод <i>Regen</i>	479
7.2.268. Метод <i>Reload</i>	480
7.2.269. Метод <i>ReloadLabels</i>	480
7.2.270. Метод <i>Remove</i>	481
7.2.271. Метод <i>RemoveCustomByIndex</i>	481
7.2.272. Метод <i>RemoveCustomByKey</i>	481
7.2.273. Метод <i>RemoveEntry</i>	481
7.2.274. Метод <i>RemoveFromMenuBar</i>	482
7.2.275. Метод <i>RemoveItems</i>	482
7.2.276. Метод <i>RemoveMenuFromMenuBar</i>	483
7.2.277. Метод <i>Rename</i>	483
7.2.278. Метод <i>RenameProfile</i>	483
7.2.279. Метод <i>Replace</i>	483
7.2.280. Метод <i>ReselectSubRegion</i>	484
7.2.281. Метод <i>ResetBlock</i>	484
7.2.282. Метод <i>ResetProfile</i>	484
7.2.283. Метод <i>Restore</i>	485
7.2.284. Метод <i>Reverse</i>	485
7.2.285. Метод <i>Rotate</i>	485
7.2.286. Метод <i>Rotate3D</i>	486
7.2.287. Метод <i>RunMacro</i>	486
7.2.288. Метод <i>Save</i>	487
7.2.289. Метод <i>SaveAs</i>	487
7.2.290. Метод <i>ScaleEntity</i>	488
7.2.291. Метод <i>SectionSolid</i>	488
7.2.292. Метод <i>Select</i>	489
7.2.293. Метод <i>SelectAtPoint</i>	490
7.2.294. Метод <i>SelectByPolygon</i>	490
7.2.295. Метод <i>SelectOnScreen</i>	491
7.2.296. Метод <i>SelectSubRegion</i>	492
7.2.297. Метод <i>SendCommand</i>	492
7.2.298. Метод <i>SendModelessOperationEnded</i>	493
7.2.299. Метод <i>SendModelessOperationStart</i>	493
7.2.300. Метод <i>SetAlignment</i>	493
7.2.301. Метод <i>SetAutoScale</i>	494
7.2.302. Метод <i>SetBackgroundColor</i>	494
7.2.303. Метод <i>SetBackgroundColorNone</i>	495

7.2.304. Метод <i>SetBitmaps</i>	495
7.2.305. Метод <i>SetBlockAttributeValue</i>	495
7.2.306. Метод <i>SetBlockRotation</i>	496
7.2.307. Метод <i>SetBlockScale</i>	496
7.2.308. Метод <i>SetBlockTableRecordId</i>	496
7.2.309. Метод <i>SetBulge</i>	497
7.2.310. Метод <i>SetCellAlignment</i>	497
7.2.311. Метод <i>SetCellBackgroundColor</i>	498
7.2.312. Метод <i>SetCellBackgroundColorNone</i>	498
7.2.313. Метод <i>SetCellContentColor</i>	498
7.2.314. Метод <i>SetCellGridColor</i>	499
7.2.315. Метод <i>SetCellGridLineWeight</i>	499
7.2.316. Метод <i>SetCellGridVisibility</i>	500
7.2.317. Метод <i>SetCellTextHeight</i>	500
7.2.318. Метод <i>SetCellTextStyle</i>	500
7.2.319. Метод <i>SetCellType</i>	501
7.2.320. Метод <i>SetColor</i>	501
7.2.321. Метод <i>SetColorBookColor</i>	501
7.2.322. Метод <i>SetColumnWidth</i>	502
7.2.323. Метод <i>SetContentColor</i>	502
7.2.324. Метод <i>SetControlPoint</i>	503
7.2.325. Метод <i>SetCustomByIndex</i>	503
7.2.326. Метод <i>SetCustomByKey</i>	503
7.2.327. Метод <i>SetCustomScale</i>	504
7.2.328. Метод <i>SetDatabase</i>	504
7.2.329. Метод <i>SetFieldId</i>	504
7.2.330. Метод <i>SetFitPoint</i>	505
7.2.331. Метод <i>SetFont</i>	505
7.2.332. Метод <i>SetGridColor</i>	505
7.2.333. Метод <i>SetGridLineWeight</i>	506
7.2.334. Метод <i>SetGridSpacing</i>	506
7.2.335. Метод <i>SetGridVisibility</i>	506
7.2.336. Метод <i>SetInvisibleEdge</i>	507
7.2.337. Метод <i>SetLayoutsToPlot</i>	507
7.2.338. Метод <i>SetNames</i>	508
7.2.339. Метод <i>SetPattern</i>	508
7.2.340. Метод <i>SetProjectFilePath</i>	508
7.2.341. Метод <i>SetRelativeDrawOrder</i>	509
7.2.342. Метод <i>SetRGB</i>	509
7.2.343. Метод <i>SetRowHeight</i>	509
7.2.344. Метод <i>SetSnapSpacing</i>	510
7.2.345. Метод <i>SetSubSelection</i>	510
7.2.346. Метод <i>SetText</i>	510
7.2.347. Метод <i>SetTextHeight</i>	511
7.2.348. Метод <i>SetTextRotation</i>	511

7.2.349. Метод <i>SetTextStyle</i>	511
7.2.350. Метод <i>SetVariable</i>	512
7.2.351. Метод <i>SetView</i>	512
7.2.352. Метод <i>SetWeight</i>	512
7.2.353. Метод <i>SetWidth</i>	513
7.2.354. Метод <i>SetWindowToPlot</i>	513
7.2.355. Метод <i>SetXData</i>	513
7.2.356. Метод <i>SetXRecordData</i>	514
7.2.357. Метод <i>Show</i>	515
7.2.358. Метод <i>SliceSolid</i>	515
7.2.359. Метод <i>Split</i>	516
7.2.360. Метод <i>StartBatchMode</i>	516
7.2.361. Метод <i>StartUndoMark</i>	517
7.2.362. Метод <i>Subtract</i>	517
7.2.363. Метод <i>SwapOrder</i>	517
7.2.364. Метод <i>SyncModelView</i>	518
7.2.365. Метод <i>TransformBy</i>	518
7.2.366. Метод <i>TranslateCoordinates</i>	518
7.2.367. Метод <i>Union</i>	519
7.2.368. Метод <i>Unload</i>	520
7.2.369. Метод <i>UnloadArx</i>	520
7.2.370. Метод <i>UnloadDVB</i>	521
7.2.371. Метод <i>UnmergeCells</i>	521
7.2.372. Метод <i>Update</i>	521
7.2.373. Метод <i>UpdateEntry</i>	521
7.2.374. Метод <i>Wblock</i>	522
7.2.375. Метод <i>ZoomAll</i>	522
7.2.376. Метод <i>ZoomCenter</i>	522
7.2.377. Метод <i>ZoomExtents</i>	523
7.2.378. Метод <i>ZoomPickWindow</i>	523
7.2.379. Метод <i>ZoomPrevious</i>	523
7.2.380. Метод <i>ZoomScaled</i>	523
7.2.381. Метод <i>ZoomWindow</i>	524
Глава 8. Свойства ActiveX	525
8.1. Дополнительные соглашения	525
8.2. Функции свойств	525
8.2.1. Свойство <i>Action</i>	527
8.2.2. Свойство <i>Active</i>	528
8.2.3. Свойство <i>ActiveDimStyle</i>	528
8.2.4. Свойство <i>ActiveDocument</i>	528
8.2.5. Свойство <i>ActiveLayer</i>	529
8.2.6. Свойство <i>ActiveLayout</i>	529
8.2.7. Свойство <i>ActiveLinetype</i>	529
8.2.8. Свойство <i>ActiveProfile</i>	529
8.2.9. Свойство <i>ActivePViewport</i>	530

8.2.10. Свойство <i>ActiveSelectionSet</i>	530
8.2.11. Свойство <i>ActiveSpace</i>	530
8.2.12. Свойство <i>ActiveTextStyle</i>	530
8.2.13. Свойство <i>ActiveUCS</i>	531
8.2.14. Свойство <i>ActiveViewport</i>	531
8.2.15. Свойство <i>ADCInsertUnitsDefaultSource</i>	531
8.2.16. Свойство <i>ADCInsertUnitsDefaultTarget</i>	532
8.2.17. Свойство <i>AffectsGraphics</i>	532
8.2.18. Свойство <i>Algorithm</i>	533
8.2.19. Свойство <i>Alignment</i>	533
8.2.20. Свойство <i>AlignmentPointAcquisition</i>	534
8.2.21. Свойство <i>AllowedValues</i>	534
8.2.22. Свойство <i>AllowLongSymbolNames</i>	535
8.2.23. Свойство <i>AltFontFile</i>	535
8.2.24. Свойство <i>AltRoundDistance</i>	535
8.2.25. Свойство <i>AltSuppressLeadingZeros</i>	536
8.2.26. Свойство <i>AltSuppressTrailingZeros</i>	536
8.2.27. Свойство <i>AltSuppressZeroFeet</i>	536
8.2.28. Свойство <i>AltSuppressZeroInches</i>	537
8.2.29. Свойство <i>AltTabletMenuFile</i>	537
8.2.30. Свойство <i>AltTextPrefix</i>	537
8.2.31. Свойство <i>AltTextSuffix</i>	538
8.2.32. Свойство <i>AltTolerancePrecision</i>	538
8.2.33. Свойство <i>AltToleranceSuppressLeadingZeros</i>	538
8.2.34. Свойство <i>AltToleranceSuppressTrailingZeros</i>	539
8.2.35. Свойство <i>AltToleranceSuppressZeroFeet</i>	539
8.2.36. Свойство <i>AltToleranceSuppressZeroInches</i>	540
8.2.37. Свойство <i>AltUnits</i>	540
8.2.38. Свойство <i>AltUnitsFormat</i>	540
8.2.39. Свойство <i>AltUnitsPrecision</i>	541
8.2.40. Свойство <i>AltUnitsScale</i>	541
8.2.41. Свойство <i>Angle</i>	541
8.2.42. Свойство <i>AngleFormat</i>	542
8.2.43. Свойство <i>AngleVertex</i>	542
8.2.44. Свойство <i>Annotation</i>	542
8.2.45. Свойство <i>Application</i>	543
8.2.46. Свойство <i>ArcEndParam</i>	543
8.2.47. Свойство <i>ArcLength</i>	543
8.2.48. Свойство <i>ArcPoint</i>	544
8.2.49. Свойство <i>ArcSmoothness</i>	544
8.2.50. Свойство <i>ArcStartParam</i>	544
8.2.51. Свойство <i>Area</i>	545
8.2.52. Свойство <i>Arrowhead1Block</i>	545
8.2.53. Свойство <i>Arrowhead1Type</i>	545
8.2.54. Свойство <i>Arrowhead2Block</i>	546

8.2.55. Свойство <i>Arrowhead2Type</i>	547
8.2.56. Свойство <i>ArrowheadBlock</i>	547
8.2.57. Свойство <i>ArrowheadSize</i>	547
8.2.58. Свойство <i>ArrowheadType</i>	548
8.2.59. Свойство <i>AssociativeHatch</i>	548
8.2.60. Свойство <i>AttachmentPoint</i>	548
8.2.61. Свойство <i>Author</i>	549
8.2.62. Свойство <i>AutoAudit</i>	549
8.2.63. Свойство <i>AutomaticPlotLog</i>	549
8.2.64. Свойство <i>AutoSaveInterval</i>	550
8.2.65. Свойство <i>AutoSavePath</i>	550
8.2.66. Свойство <i>AutoSnapAperture</i>	550
8.2.67. Свойство <i>AutoSnapApertureSize</i>	551
8.2.68. Свойство <i>AutoSnapMagnet</i>	551
8.2.69. Свойство <i>AutoSnapMarker</i>	551
8.2.70. Свойство <i>AutoSnapMarkerColor</i>	552
8.2.71. Свойство <i>AutoSnapMarkerSize</i>	552
8.2.72. Свойство <i>AutoSnapToolTip</i>	553
8.2.73. Свойство <i>AutoTrackingVecColor</i>	553
8.2.74. Свойство <i>AutoTrackTooltip</i>	553
8.2.75. Свойство <i>BackgroundFill</i>	554
8.2.76. Свойство <i>Backward</i>	554
8.2.77. Свойство <i>BasePoint</i>	554
8.2.78. Свойство <i>BatchPlotProgress</i>	555
8.2.79. Свойство <i>BeepOnError</i>	555
8.2.80. Свойство <i>BigFontFile</i>	555
8.2.81. Свойство <i>BitFlags</i>	556
8.2.82. Свойство <i>Block</i>	556
8.2.83. Свойство <i>Blocks</i>	556
8.2.84. Свойство <i>BlockScaling</i>	557
8.2.85. Свойство <i>Blue</i>	557
8.2.86. Свойство <i>BookName</i>	557
8.2.87. Свойство <i>Brightness</i>	557
8.2.88. Свойство <i>CanonicalMediaName</i>	558
8.2.89. Свойство <i>Caption</i>	558
8.2.90. Свойство <i>Catalog</i>	558
8.2.91. Свойство <i>CategoryName</i>	559
8.2.92. Свойство <i>Center</i>	559
8.2.93. Свойство <i>CenterMarkSize</i>	559
8.2.94. Свойство <i>CenterPlot</i>	560
8.2.95. Свойство <i>CenterPoint</i>	560
8.2.96. Свойство <i>CenterType</i>	560
8.2.97. Свойство <i>Centroid</i>	561
8.2.98. Свойство <i>Check</i>	561
8.2.99. Свойство <i>ChordPoint</i>	562

8.2.100. Свойство <i>Circumference</i>	562
8.2.101. Свойство <i>Clipped</i>	562
8.2.102. Свойство <i>ClippingEnabled</i>	563
8.2.103. Свойство <i>Closed</i>	563
8.2.104. Свойство <i>Color</i>	563
8.2.105. Свойство <i>ColorBookPath</i>	564
8.2.106. Свойство <i>ColorIndex</i>	564
8.2.107. Свойство <i>ColorMethod</i>	564
8.2.108. Свойство <i>ColorName</i>	565
8.2.109. Свойство <i>Columns</i>	565
8.2.110. Свойство <i>ColumnSpacing</i>	565
8.2.111. Свойство <i>ColumnWidth</i>	566
8.2.112. Свойство <i>Comment</i>	566
8.2.113. Свойство <i>Comments</i>	566
8.2.114. Свойство <i>ConfigFile</i>	566
8.2.115. Свойство <i>ConfigName</i>	567
8.2.116. Свойство <i>Constant</i>	567
8.2.117. Свойство <i>ConstantWidth</i>	567
8.2.118. Свойство <i>ContinuousPlotLog</i>	567
8.2.119. Свойство <i>ContourLinesPerSurface</i>	568
8.2.120. Свойство <i>Contrast</i>	568
8.2.121. Свойство <i>ControlPoints</i>	568
8.2.122. Свойство <i>Coordinate</i>	569
8.2.123. Свойство <i>Coordinates</i>	569
8.2.124. Свойство <i>Count</i>	570
8.2.125. Свойство <i>CreateBackup</i>	570
8.2.126. Свойство <i>CursorSize</i>	570
8.2.127. Свойство <i>CustomDictionary</i>	571
8.2.128. Свойство <i>CustomScale</i>	571
8.2.129. Свойство <i>Database</i>	571
8.2.130. Свойство <i>DataSource</i>	571
8.2.131. Свойство <i>DataSourceLocation</i>	572
8.2.132. Свойство <i>DbConnect</i>	572
8.2.133. Свойство <i>DecimalSeparator</i>	572
8.2.134. Свойство <i>DefaultInternetURL</i>	573
8.2.135. Свойство <i>DefaultOutputDevice</i>	573
8.2.136. Свойство <i>DefaultPlotStyleForLayer</i>	573
8.2.137. Свойство <i>DefaultPlotStyleForObjects</i>	574
8.2.138. Свойство <i>DefaultPlotStyleTable</i>	574
8.2.139. Свойство <i>DefaultPlotToFilePath</i>	574
8.2.140. Свойство <i>DefinedSize</i>	575
8.2.141. Свойство <i>Degree</i>	575
8.2.142. Свойство <i>Delta</i>	575
8.2.143. Свойство <i>DemandLoadARXApp</i>	576
8.2.144. Свойство <i>Description</i>	576

8.2.145. Свойство <i>Diameter</i>	576
8.2.146. Свойство <i>Dictionaries</i>	577
8.2.147. Свойство <i>DimensionLineColor</i>	577
8.2.148. Свойство <i>DimensionLineExtend</i>	577
8.2.149. Свойство <i>DimensionLineWeight</i>	578
8.2.150. Свойство <i>DimLine1Suppress</i>	579
8.2.151. Свойство <i>DimLine2Suppress</i>	579
8.2.152. Свойство <i>DimLineInside</i>	579
8.2.153. Свойство <i>DimLineSuppress</i>	580
8.2.154. Свойство <i>DimStyles</i>	580
8.2.155. Свойство <i>Direction</i>	580
8.2.156. Свойство <i>DirectionVector</i>	581
8.2.157. Свойство <i>Display</i>	581
8.2.158. Свойство <i>DisplayableAttributes</i>	582
8.2.159. Свойство <i>DisplayGrips</i>	582
8.2.160. Свойство <i>DisplayGripsWithinBlocks</i>	582
8.2.161. Свойство <i>DisplayLayoutTabs</i>	583
8.2.162. Свойство <i>DisplayLocked</i>	583
8.2.163. Свойство <i>DisplayOLEScale</i>	583
8.2.164. Свойство <i>DisplayScreenMenu</i>	584
8.2.165. Свойство <i>DisplayScrollBars</i>	584
8.2.166. Свойство <i>DisplaySilhouette</i>	584
8.2.167. Свойство <i>DisplayValues</i>	585
8.2.168. Свойство <i>DockedVisibleLines</i>	585
8.2.169. Свойство <i>DockStatus</i>	586
8.2.170. Свойство <i>Document</i>	586
8.2.171. Свойство <i>Documents</i>	586
8.2.172. Свойство <i>Drafting</i>	587
8.2.173. Свойство <i>DrawingDirection</i>	587
8.2.174. Свойство <i>DriversPath</i>	587
8.2.175. Свойство <i>EffectiveName</i>	588
8.2.176. Свойство <i>Elevation</i>	588
8.2.177. Свойство <i>ElevationModelSpace</i>	588
8.2.178. Свойство <i>ElevationPaperSpace</i>	589
8.2.179. Свойство <i>Enable</i>	589
8.2.180. Свойство <i>EnableStartupDialog</i>	589
8.2.181. Свойство <i>EndAngle</i>	590
8.2.182. Свойство <i>EndParameter</i>	590
8.2.183. Свойство <i>EndPoint</i>	590
8.2.184. Свойство <i>EndSubMenuLevel</i>	591
8.2.185. Свойство <i>EndTangent</i>	591
8.2.186. Свойство <i>EntityColor</i>	591
8.2.187. Свойство <i>Explodable</i>	592
8.2.188. Свойство <i>ExtensionLineColor</i>	592
8.2.189. Свойство <i>ExtensionLineExtend</i>	592

8.2.190. Свойство <i>ExtensionLineOffset</i>	593
8.2.191. Свойство <i>ExtensionLineWeight</i>	593
8.2.192. Свойство <i>ExtLine1EndPoint</i>	593
8.2.193. Свойство <i>ExtLine1Point</i>	594
8.2.194. Свойство <i>ExtLine1StartPoint</i>	594
8.2.195. Свойство <i>ExtLine1Suppress</i>	594
8.2.196. Свойство <i>ExtLine2EndPoint</i>	595
8.2.197. Свойство <i>ExtLine2Point</i>	595
8.2.198. Свойство <i>ExtLine2StartPoint</i>	595
8.2.199. Свойство <i>ExtLine2Suppress</i>	596
8.2.200. Свойство <i>Fade</i>	596
8.2.201. Свойство <i>Feature</i>	596
8.2.202. Свойство <i>FieldLength</i>	596
8.2.203. Свойство <i>FieldName</i>	597
8.2.204. Свойство <i>FileDependencies</i>	597
8.2.205. Свойство <i>FileName</i>	597
8.2.206. Свойство <i>Files</i>	598
8.2.207. Свойство <i>FileSize</i>	598
8.2.208. Свойство <i>FingerprintGuid</i>	598
8.2.209. Свойство <i>Fit</i>	598
8.2.210. Свойство <i>FitPoints</i>	599
8.2.211. Свойство <i>FitTolerance</i>	599
8.2.212. Свойство <i>FloatingRows</i>	600
8.2.213. Свойство <i>FlowDirection</i>	600
8.2.214. Свойство <i>Flyout</i>	600
8.2.215. Свойство <i>FontFile</i>	601
8.2.216. Свойство <i>FontFileMap</i>	601
8.2.217. Свойство <i>ForceLineInside</i>	601
8.2.218. Свойство <i>Format</i>	602
8.2.219. Свойство <i>FoundPath</i>	602
8.2.220. Свойство <i>FractionFormat</i>	602
8.2.221. Свойство <i>Freeze</i>	603
8.2.222. Свойство <i>FullCRCValidation</i>	603
8.2.223. Свойство <i>FullFileName</i>	604
8.2.224. Свойство <i>FullName</i>	604
8.2.225. Свойство <i>FullScreenTrackingVector</i>	605
8.2.226. Свойство <i>GradientAngle</i>	605
8.2.227. Свойство <i>GradientCentered</i>	605
8.2.228. Свойство <i>GradientColor1</i>	606
8.2.229. Свойство <i>GradientColor2</i>	606
8.2.230. Свойство <i>GradientName</i>	607
8.2.231. Свойство <i>GraphicsWinLayoutBackgrndColor</i>	607
8.2.232. Свойство <i>GraphicsWinModelBackgrndColor</i>	608
8.2.233. Свойство <i>Green</i>	608
8.2.234. Свойство <i>GridOn</i>	608

8.2.235. Свойство <i>GripColorSelected</i>	609
8.2.236. Свойство <i>GripColorUnselected</i>	609
8.2.237. Свойство <i>GripSize</i>	610
8.2.238. Свойство <i>Groups</i>	610
8.2.239. Свойство <i>Handle</i>	610
8.2.240. Свойство <i>HasAttributes</i>	610
8.2.241. Свойство <i>HasExtensionDictionary</i>	611
8.2.242. Свойство <i>HasLeader</i>	611
8.2.243. Свойство <i>HasSheetView</i>	611
8.2.244. Свойство <i>HasSubSelection</i>	612
8.2.245. Свойство <i>HasVpAssociation</i>	612
8.2.246. Свойство <i>HatchObjectType</i>	612
8.2.247. Свойство <i>HatchStyle</i>	613
8.2.248. Свойство <i>HeaderSuppressed</i>	613
8.2.249. Свойство <i>Height</i>	613
8.2.250. Свойство <i>HelpFilePath</i>	614
8.2.251. Свойство <i>HelpString</i>	614
8.2.252. Свойство <i>HistoryLines</i>	614
8.2.253. Свойство <i>HorizontalTextPosition</i>	615
8.2.254. Свойство <i>HorzCellMargin</i>	615
8.2.255. Свойство <i>HWND</i>	615
8.2.256. Свойство <i>HyperlinkBase</i>	616
8.2.257. Свойство <i>HyperlinkDisplayCursor</i>	616
8.2.258. Свойство <i>HyperlinkDisplayTooltip</i>	616
8.2.259. Свойство <i>Hyperlinks</i>	617
8.2.260. Свойство <i>ImageFile</i>	617
8.2.261. Свойство <i>ImageFrameHighlight</i>	617
8.2.262. Свойство <i>ImageHeight</i>	617
8.2.263. Свойство <i>ImageVisibility</i>	618
8.2.264. Свойство <i>ImageWidth</i>	618
8.2.265. Свойство <i>IncrementalSavePercent</i>	619
8.2.266. Свойство <i>Index</i>	619
8.2.267. Свойство <i>InsertionPoint</i>	619
8.2.268. Свойство <i>InsUnits</i>	620
8.2.269. Свойство <i>InsUnitsFactor</i>	620
8.2.270. Свойство <i>Invisible</i>	620
8.2.271. Свойство <i>IsCloned</i>	620
8.2.272. Свойство <i>IsDynamicBlock</i>	621
8.2.273. Свойство <i>IsLayout</i>	621
8.2.274. Свойство <i>IsModified</i>	622
8.2.275. Свойство <i>ISOPenWidth</i>	622
8.2.276. Свойство <i>IsOwnerXlated</i>	623
8.2.277. Свойство <i>IsPartial</i>	623
8.2.278. Свойство <i>IsPeriodic</i>	624
8.2.279. Свойство <i>IsPlanar</i>	624

8.2.280. Свойство <i>IsPrimary</i>	624
8.2.281. Свойство <i>IsQuiescent</i>	625
8.2.282. Свойство <i>IsRational</i>	625
8.2.283. Свойство <i>Issuer</i>	625
8.2.284. Свойство <i>IsXRef</i>	626
8.2.285. Свойство <i>JogAngle</i>	626
8.2.286. Свойство <i>JogLocation</i>	626
8.2.287. Свойство <i>Justification</i>	627
8.2.288. Свойство <i>Key</i>	627
8.2.289. Свойство <i>KeyboardAccelerator</i>	628
8.2.290. Свойство <i>KeyboardPriority</i>	628
8.2.291. Свойство <i>KeyDescriptions</i>	628
8.2.292. Свойство <i>KeyLength</i>	629
8.2.293. Свойство <i>KeyValues</i>	629
8.2.294. Свойство <i>Keywords</i>	629
8.2.295. Свойство <i>Knots</i>	630
8.2.296. Свойство <i>Label</i>	630
8.2.297. Свойство <i>LabelBlockId</i>	631
8.2.298. Свойство <i>LabelTemplate</i>	631
8.2.299. Свойство <i>LargeButtons</i>	632
8.2.300. Свойство <i>LastHeight</i>	632
8.2.301. Свойство <i>LastSavedBy</i>	632
8.2.302. Свойство <i>Layer</i>	633
8.2.303. Свойство <i>LayerOn</i>	633
8.2.304. Свойство <i>Layers</i>	633
8.2.305. Свойство <i>LayerState</i>	633
8.2.306. Свойство <i>Layout</i>	634
8.2.307. Свойство <i>LayoutCreateViewport</i>	634
8.2.308. Свойство <i>LayoutCrosshairColor</i>	634
8.2.309. Свойство <i>LayoutDisplayMargins</i>	635
8.2.310. Свойство <i>LayoutDisplayPaper</i>	635
8.2.311. Свойство <i>LayoutDisplayPaperShadow</i>	635
8.2.312. Свойство <i>LayoutId</i>	636
8.2.313. Свойство <i>Layouts</i>	636
8.2.314. Свойство <i>LayoutShowPlotSetup</i>	636
8.2.315. Свойство <i>Leader1Point</i>	637
8.2.316. Свойство <i>Leader2Point</i>	637
8.2.317. Свойство <i>LeaderAttachment</i>	637
8.2.318. Свойство <i>LeaderLength</i>	638
8.2.319. Свойство <i>LeaderObjectId</i>	638
8.2.320. Свойство <i>LeaderOffset</i>	638
8.2.321. Свойство <i>Left</i>	639
8.2.322. Свойство <i>Length</i>	639
8.2.323. Свойство <i>LensLength</i>	639
8.2.324. Свойство <i>Limits</i>	640

8.2.325. Свойство <i>LinearScaleFactor</i>	640
8.2.326. Свойство <i>LineSpacingDistance</i>	641
8.2.327. Свойство <i>LineSpacingFactor</i>	641
8.2.328. Свойство <i>LineSpacingStyle</i>	641
8.2.329. Свойство <i>Linetype</i>	642
8.2.330. Свойство <i>LinetypeGeneration</i>	642
8.2.331. Свойство <i>Linetypes</i>	642
8.2.332. Свойство <i>LinetypeScale</i>	642
8.2.333. Свойство <i>LineWeight</i>	643
8.2.334. Свойство <i>LineWeightDisplay</i>	643
8.2.335. Свойство <i>Link</i>	643
8.2.336. Свойство <i>LinkTemplate</i>	644
8.2.337. Свойство <i>LinkType</i>	644
8.2.338. Свойство <i>LoadAcadLspInAllDocuments</i>	645
8.2.339. Свойство <i>LocaleId</i>	645
8.2.340. Свойство <i>Lock</i>	645
8.2.341. Свойство <i>LockAspectRatio</i>	646
8.2.342. Свойство <i>LockPosition</i>	646
8.2.343. Свойство <i>LogFileOn</i>	646
8.2.344. Свойство <i>LogFilePath</i>	647
8.2.345. Свойство <i>LowerLeftCorner</i>	647
8.2.346. Свойство <i>Macro</i>	647
8.2.347. Свойство <i>MainDictionary</i>	648
8.2.348. Свойство <i>MajorAxis</i>	648
8.2.349. Свойство <i>MajorRadius</i>	648
8.2.350. Свойство <i>Mask</i>	649
8.2.351. Свойство <i>MaxActiveViewports</i>	650
8.2.352. Свойство <i>MaxAutoCADWindow</i>	650
8.2.353. Свойство <i>MClose</i>	650
8.2.354. Свойство <i>MDensity</i>	651
8.2.355. Свойство <i>Measurement</i>	651
8.2.356. Свойство <i>MenuBar</i>	651
8.2.357. Свойство <i>MenuFile</i>	651
8.2.358. Свойство <i>MenuFileName</i>	652
8.2.359. Свойство <i>MenuGroups</i>	652
8.2.360. Свойство <i>Menus</i>	652
8.2.361. Свойство <i>MinimumTableHeight</i>	653
8.2.362. Свойство <i>MinimumTableWidth</i>	653
8.2.363. Свойство <i>MinorAxis</i>	653
8.2.364. Свойство <i>MinorRadius</i>	653
8.2.365. Свойство <i>MLineStyle</i>	654
8.2.366. Свойство <i>Mode</i>	654
8.2.367. Свойство <i>ModelCrosshairColor</i>	655
8.2.368. Свойство <i>ModelSpace</i>	655
8.2.369. Свойство <i>ModelType</i>	655

8.2.370. Свойство <i>ModelView</i>	656
8.2.371. Свойство <i>MomentOfInertia</i>	656
8.2.372. Свойство <i>MRUNumber</i>	656
8.2.373. Свойство <i>MSpace</i>	657
8.2.374. Свойство <i>MTextObjectId</i>	657
8.2.375. Свойство <i>MVertexCount</i>	657
8.2.376. Свойство <i>Name</i>	658
8.2.377. Свойство <i>NameNoMnemonic</i>	658
8.2.378. Свойство <i>NClose</i>	658
8.2.379. Свойство <i>NDensity</i>	659
8.2.380. Свойство <i>Normal</i>	659
8.2.381. Свойство <i>NumberOfControlPoints</i>	660
8.2.382. Свойство <i>NumberOfCopies</i>	660
8.2.383. Свойство <i>NumberOfFaces</i>	660
8.2.384. Свойство <i>NumberOfFitPoints</i>	660
8.2.385. Свойство <i>NumberOfLoops</i>	661
8.2.386. Свойство <i>NumberOfVertices</i>	661
8.2.387. Свойство <i>NumericScale</i>	661
8.2.388. Свойство <i>NVertexCount</i>	662
8.2.389. Свойство <i>ObjectID</i>	662
8.2.390. Свойство <i>ObjectName</i>	662
8.2.391. Свойство <i>ObjectSnapMode</i>	662
8.2.392. Свойство <i>ObjectSortByPlotting</i>	663
8.2.393. Свойство <i>ObjectSortByPSOutput</i>	663
8.2.394. Свойство <i>ObjectSortByRedraws</i>	663
8.2.395. Свойство <i>ObjectSortByRegens</i>	664
8.2.396. Свойство <i>ObjectSortBySelection</i>	664
8.2.397. Свойство <i>ObjectSortBySnap</i>	664
8.2.398. Свойство <i>ObliqueAngle</i>	665
8.2.399. Свойство <i>OleItemType</i>	665
8.2.400. Свойство <i>OleLaunch</i>	666
8.2.401. Свойство <i>OlePlotQuality</i>	666
8.2.402. Свойство <i>OleQuality</i>	666
8.2.403. Свойство <i>OleSourceApp</i>	667
8.2.404. Свойство <i>OnMenuBar</i>	667
8.2.405. Свойство <i>OpenSave</i>	668
8.2.406. Свойство <i>Origin</i>	668
8.2.407. Свойство <i>OrthoOn</i>	668
8.2.408. Свойство <i>Output</i>	668
8.2.409. Свойство <i>OverrideCenter</i>	669
8.2.410. Свойство <i>OwnerID</i>	669
8.2.411. Свойство <i>PageSetupOverridesTemplateFile</i>	669
8.2.412. Свойство <i>PaperSpace</i>	670
8.2.413. Свойство <i>PaperUnits</i>	670
8.2.414. Свойство <i>Parent</i>	670

8.2.415. Свойство <i>Password</i>	671
8.2.416. Свойство <i>Path</i>	671
8.2.417. Свойство <i>PatternAngle</i>	671
8.2.418. Свойство <i>PatternDouble</i>	671
8.2.419. Свойство <i>PatternName</i>	672
8.2.420. Свойство <i>PatternScale</i>	672
8.2.421. Свойство <i>PatternSpace</i>	673
8.2.422. Свойство <i>PatternType</i>	673
8.2.423. Свойство <i>Perimeter</i>	673
8.2.424. Свойство <i>PickAdd</i>	674
8.2.425. Свойство <i>PickAuto</i>	674
8.2.426. Свойство <i>PickBoxSize</i>	674
8.2.427. Свойство <i>PickDrag</i>	675
8.2.428. Свойство <i>PickFirst</i>	675
8.2.429. Свойство <i>PickfirstSelectionSet</i>	675
8.2.430. Свойство <i>PickGroup</i>	675
8.2.431. Свойство <i>Plot</i>	676
8.2.432. Свойство <i>PlotConfigurations</i>	676
8.2.433. Свойство <i>PlotHidden</i>	676
8.2.434. Свойство <i>PlotLegacy</i>	677
8.2.435. Свойство <i>PlotLogFilePath</i>	677
8.2.436. Свойство <i>PlotOrigin</i>	677
8.2.437. Свойство <i>PlotPolicy</i>	678
8.2.438. Свойство <i>PlotRotation</i>	678
8.2.439. Свойство <i>PlotStyleName</i>	678
8.2.440. Свойство <i>Plottable</i>	679
8.2.441. Свойство <i>PlotType</i>	679
8.2.442. Свойство <i>PlotViewportBorders</i>	679
8.2.443. Свойство <i>PlotViewportsFirst</i>	680
8.2.444. Свойство <i>PlotWithLineweights</i>	680
8.2.445. Свойство <i>PlotWithPlotStyles</i>	680
8.2.446. Свойство <i>PolarTrackingVector</i>	681
8.2.447. Свойство <i>PostScriptPrologFile</i>	681
8.2.448. Свойство <i>Precision</i>	682
8.2.449. Свойство <i>Preferences</i>	682
8.2.450. Свойство <i>Preset</i>	682
8.2.451. Свойство <i>PrimaryUnitsPrecision</i>	683
8.2.452. Свойство <i>PrincipalDirections</i>	683
8.2.453. Свойство <i>PrincipalMoments</i>	684
8.2.454. Свойство <i>PrinterConfigPath</i>	684
8.2.455. Свойство <i>PrinterDescPath</i>	684
8.2.456. Свойство <i>PrinterPaperSizeAlert</i>	684
8.2.457. Свойство <i>PrinterSpoolAlert</i>	685
8.2.458. Свойство <i>PrinterStyleSheetPath</i>	685
8.2.459. Свойство <i>PrintFile</i>	686

8.2.460. Свойство <i>PrintSpoolerPath</i>	686
8.2.461. Свойство <i>PrintSpoolExecutable</i>	686
8.2.462. Свойство <i>ProductOfInertia</i>	686
8.2.463. Свойство <i>Profiles</i>	687
8.2.464. Свойство <i>PromptString</i>	687
8.2.465. Свойство <i>PropertyName</i>	687
8.2.466. Свойство <i>ProviderName</i>	688
8.2.467. Свойство <i>ProviderType</i>	688
8.2.468. Свойство <i>ProxyImage</i>	688
8.2.469. Свойство <i>QNewTemplateFile</i>	689
8.2.470. Свойство <i>QuietErrorMode</i>	689
8.2.471. Свойство <i>RadiiOfGyration</i>	689
8.2.472. Свойство <i>Radius</i>	690
8.2.473. Свойство <i>RadiusRatio</i>	690
8.2.474. Свойство <i>ReadOnly</i>	690
8.2.475. Свойство <i>Red</i>	690
8.2.476. Свойство <i>ReferenceCount</i>	691
8.2.477. Свойство <i>RegenerateTableSuppressed</i>	691
8.2.478. Свойство <i>RegisteredApplications</i>	691
8.2.479. Свойство <i>RenderSmoothness</i>	691
8.2.480. Свойство <i>RevisionNumber</i>	692
8.2.481. Свойство <i>Rotation</i>	692
8.2.482. Свойство <i>RoundDistance</i>	692
8.2.483. Свойство <i>RowHeight</i>	693
8.2.484. Свойство <i>Rows</i>	693
8.2.485. Свойство <i>RowSpacing</i>	693
8.2.486. Свойство <i>SaveAsType</i>	693
8.2.487. Свойство <i>Saved</i>	694
8.2.488. Свойство <i>SavePreviewThumbnail</i>	695
8.2.489. Свойство <i>ScaleFactor</i>	695
8.2.490. Свойство <i>ScaleHeight</i>	695
8.2.491. Свойство <i>ScaleLineweights</i>	696
8.2.492. Свойство <i>ScaleWidth</i>	696
8.2.493. Свойство <i>Schema</i>	696
8.2.494. Свойство <i>SCMCommandMode</i>	697
8.2.495. Свойство <i>SCMDefaultMode</i>	697
8.2.496. Свойство <i>SCMEditMode</i>	697
8.2.497. Свойство <i>SCMTimeMode</i>	698
8.2.498. Свойство <i>SCMTimeValue</i>	698
8.2.499. Свойство <i>SecondPoint</i>	698
8.2.500. Свойство <i>SegmentPerPolyline</i>	699
8.2.501. Свойство <i>Selection</i>	699
8.2.502. Свойство <i>SelectionSets</i>	699
8.2.503. Свойство <i>SerialNumber</i>	699
8.2.504. Свойство <i>ShadePlot</i>	700

8.2.505. Свойство <i>SheetView</i>	700
8.2.506. Свойство <i>ShortcutMenu</i>	700
8.2.507. Свойство <i>ShortCutMenuDisplay</i>	701
8.2.508. Свойство <i>Show</i>	701
8.2.509. Свойство <i>ShowPlotStyles</i>	701
8.2.510. Свойство <i>ShowProxyDialogBox</i>	702
8.2.511. Свойство <i>ShowRasterImage</i>	702
8.2.512. Свойство <i>ShowRotation</i>	702
8.2.513. Свойство <i>ShowWarningMessages</i>	703
8.2.514. Свойство <i>SingleDocumentMode</i>	703
8.2.515. Свойство <i>SnapBasePoint</i>	704
8.2.516. Свойство <i>SnapOn</i>	704
8.2.517. Свойство <i>SnapRotationAngle</i>	704
8.2.518. Свойство <i>SolidFill</i>	705
8.2.519. Свойство <i>StandardScale</i>	705
8.2.520. Свойство <i>StartAngle</i>	706
8.2.521. Свойство <i>StartParameter</i>	707
8.2.522. Свойство <i>StartPoint</i>	707
8.2.523. Свойство <i>StartTangent</i>	707
8.2.524. Свойство <i>StatusId</i>	707
8.2.525. Свойство <i>StoreSQLIndex</i>	708
8.2.526. Свойство <i>StyleName</i>	708
8.2.527. Свойство <i>StyleSheet</i>	709
8.2.528. Свойство <i>Subject</i>	709
8.2.529. Свойство <i>SubMenu</i>	709
8.2.530. Свойство <i>SummaryInfo</i>	709
8.2.531. Свойство <i>SupportPath</i>	710
8.2.532. Свойство <i>SuppressLeadingZeros</i>	710
8.2.533. Свойство <i>SuppressTrailingZeros</i>	710
8.2.534. Свойство <i>SuppressZeroFeet</i>	711
8.2.535. Свойство <i>SuppressZeroInches</i>	711
8.2.536. Свойство <i>SymbolPosition</i>	711
8.2.537. Свойство <i>System</i>	712
8.2.538. Свойство <i>Table</i>	712
8.2.539. Свойство <i>TablesReadOnly</i>	712
8.2.540. Свойство <i>TableStyleOverrides</i>	713
8.2.541. Свойство <i>TabOrder</i>	713
8.2.542. Свойство <i>TagString</i>	713
8.2.543. Свойство <i>Target</i>	714
8.2.544. Свойство <i>TempFileExtension</i>	714
8.2.545. Свойство <i>TempFilePath</i>	714
8.2.546. Свойство <i>TemplateDwgPath</i>	715
8.2.547. Свойство <i>TempXrefPath</i>	715
8.2.548. Свойство <i>Text</i>	715
8.2.549. Свойство <i>TextAlignmentPoint</i>	716

8.2.550. Свойство <i>TextAttachment</i>	716
8.2.551. Свойство <i>TextColor</i>	716
8.2.552. Свойство <i>TextEditor</i>	717
8.2.553. Свойство <i>TextFont</i>	717
8.2.554. Свойство <i>TextFontSize</i>	717
8.2.555. Свойство <i>TextFontStyle</i>	717
8.2.556. Свойство <i>TextFrameDisplay</i>	718
8.2.557. Свойство <i>TextGap</i>	718
8.2.558. Свойство <i>TextGenerationFlag</i>	718
8.2.559. Свойство <i>TextHeight</i>	719
8.2.560. Свойство <i>TextInside</i>	719
8.2.561. Свойство <i>TextInsideAlign</i>	720
8.2.562. Свойство <i>TextMovement</i>	720
8.2.563. Свойство <i>TextOutsideAlign</i>	721
8.2.564. Свойство <i>TextOverride</i>	721
8.2.565. Свойство <i>TextPosition</i>	721
8.2.566. Свойство <i>TextPrecision</i>	722
8.2.567. Свойство <i>TextPrefix</i>	722
8.2.568. Свойство <i>TextRotation</i>	722
8.2.569. Свойство <i>TextString</i>	723
8.2.570. Свойство <i>TextStyle</i>	723
8.2.571. Свойство <i>TextStyles</i>	723
8.2.572. Свойство <i>TextSuffix</i>	723
8.2.573. Свойство <i>TextureMapPath</i>	724
8.2.574. Свойство <i>TextWinBackgrndColor</i>	724
8.2.575. Свойство <i>TextWinTextColor</i>	724
8.2.576. Свойство <i>Thickness</i>	725
8.2.577. Свойство <i>TimeServer</i>	725
8.2.578. Свойство <i>TimeStamp</i>	725
8.2.579. Свойство <i>TipOffset</i>	726
8.2.580. Свойство <i>Title</i>	726
8.2.581. Свойство <i>TitleSuppressed</i>	726
8.2.582. Свойство <i>ToleranceDisplay</i>	727
8.2.583. Свойство <i>ToleranceHeightScale</i>	727
8.2.584. Свойство <i>ToleranceJustification</i>	728
8.2.585. Свойство <i>ToleranceLowerLimit</i>	728
8.2.586. Свойство <i>TolerancePrecision</i>	728
8.2.587. Свойство <i>ToleranceSuppressLeadingZeros</i>	729
8.2.588. Свойство <i>ToleranceSuppressTrailingZeros</i>	729
8.2.589. Свойство <i>ToleranceSuppressZeroFeet</i>	729
8.2.590. Свойство <i>ToleranceSuppressZeroInches</i>	730
8.2.591. Свойство <i>ToleranceUpperLimit</i>	730
8.2.592. Свойство <i>Toolbars</i>	730
8.2.593. Свойство <i>ToolPalettePath</i>	731
8.2.594. Свойство <i>Top</i>	731

8.2.595. Свойство <i>TotalAngle</i>	731
8.2.596. Свойство <i>TranslateIDs</i>	732
8.2.597. Свойство <i>Transparency</i>	732
8.2.598. Свойство <i>TrueColor</i>	732
8.2.599. Свойство <i>TrueColorImages</i>	733
8.2.600. Свойство <i>TwistAngle</i>	733
8.2.601. Свойство <i>Type</i>	733
8.2.602. Свойство <i>UCSIconAtOrigin</i>	735
8.2.603. Свойство <i>UCSIconOn</i>	736
8.2.604. Свойство <i>UCSPerViewport</i>	736
8.2.605. Свойство <i>Units</i>	736
8.2.606. Свойство <i>UnitsFormat</i>	736
8.2.607. Свойство <i>UnitsType</i>	737
8.2.608. Свойство <i>Updatable</i>	737
8.2.609. Свойство <i>UpperRightCorner</i>	738
8.2.610. Свойство <i>UpsideDown</i>	738
8.2.611. Свойство <i>URL</i>	738
8.2.612. Свойство <i>URLDescription</i>	739
8.2.613. Свойство <i>URLNamedLocation</i>	739
8.2.614. Свойство <i>Used</i>	739
8.2.615. Свойство <i>UseLastPlotSettings</i>	739
8.2.616. Свойство <i>User</i>	740
8.2.617. Свойство <i>UserCoordinateSystems</i>	740
8.2.618. Свойство <i>UseStandardScale</i>	740
8.2.619. Свойство <i>Utility</i>	741
8.2.620. Свойство <i>Value</i>	741
8.2.621. Свойство <i>VBE</i>	741
8.2.622. Свойство <i>Verify</i>	742
8.2.623. Свойство <i>Version</i>	742
8.2.624. Свойство <i>VersionGuid</i>	742
8.2.625. Свойство <i>VertCellMargin</i>	742
8.2.626. Свойство <i>VerticalTextPosition</i>	743
8.2.627. Свойство <i>ViewportDefault</i>	743
8.2.628. Свойство <i>ViewportOn</i>	744
8.2.629. Свойство <i>Viewports</i>	744
8.2.630. Свойство <i>Views</i>	744
8.2.631. Свойство <i>ViewToPlot</i>	744
8.2.632. Свойство <i>VisibilityEdge1</i>	745
8.2.633. Свойство <i>VisibilityEdge2</i>	745
8.2.634. Свойство <i>VisibilityEdge3</i>	745
8.2.635. Свойство <i>VisibilityEdge4</i>	745
8.2.636. Свойство <i>Visible</i>	746
8.2.637. Свойство <i>Volume</i>	746
8.2.638. Свойство <i>Weights</i>	746
8.2.639. Свойство <i>Width</i>	747

8.2.640. Свойство <i>WindowLeft</i>	747
8.2.641. Свойство <i>WindowState</i>	747
8.2.642. Свойство <i>WindowTitle</i>	748
8.2.643. Свойство <i>WindowTop</i>	748
8.2.644. Свойство <i>WorkspacePath</i>	748
8.2.645. Свойство <i>XEffectiveScaleFactor</i>	748
8.2.646. Свойство <i>XRefDatabase</i>	749
8.2.647. Свойство <i>XrefDemandLoad</i>	749
8.2.648. Свойство <i>XRefEdit</i>	749
8.2.649. Свойство <i>XRefFadeIntensity</i>	750
8.2.650. Свойство <i>XRefLayerVisibility</i>	750
8.2.651. Свойство <i>XScaleFactor</i>	750
8.2.652. Свойство <i>XVector</i>	750
8.2.653. Свойство <i>YEffectiveScaleFactor</i>	751
8.2.654. Свойство <i>YScaleFactor</i>	751
8.2.655. Свойство <i>YVector</i>	751
8.2.656. Свойство <i>ZEffectiveScaleFactor</i>	752
8.2.657. Свойство <i>ZScaleFactor</i>	752
Глава 9. События и реакторы	753
9.1. Обрабатываемые события	753
9.2. Классификация реакторов	756
9.3. Справочные функции реакторов.....	758
9.3.1. Функция <i>vlr-types</i>	758
9.3.2. Функция <i>vlr-reactors</i>	758
9.3.3. Функция <i>vlr-type</i>	759
9.4. Действия реакторов	759
9.4.1. Функция <i>vlr-reaction-names</i>	759
9.4.2. Функция <i>vlr-beep-reaction</i>	760
9.4.3. Функция <i>vlr-trace-reaction</i>	760
9.5. Создание реакторов	760
9.5.1. Функция <i>vlr-acdb-reactor</i>	761
9.5.2. Функция <i>vlr-docmanager-reactor</i>	762
9.5.3. Функция <i>vlr-linker-reactor</i>	763
9.5.4. Функция <i>vlr-object-reactor</i>	764
9.5.5. Функция <i>vlr-editor-reactor</i>	765
9.5.6. Функция <i>vlr-command-reactor</i>	768
9.5.7. Функция <i>vlr-deepclone-reactor</i>	768
9.5.8. Функция <i>vlr-dwg-reactor</i>	769
9.5.9. Функция <i>vlr-dxf-reactor</i>	770
9.5.10. Функция <i>vlr-insert-reactor</i>	771
9.5.11. Функция <i>vlr-lisp-reactor</i>	772
9.5.12. Функция <i>vlr-miscellaneous-reactor</i>	773
9.5.13. Функция <i>vlr-mouse-reactor</i>	774
9.5.14. Функция <i>vlr-sysvar-reactor</i>	774

9.5.15. Функция <i>vlr-toolbar-reactor</i>	775
9.5.16. Функция <i>vlr-undo-reactor</i>	776
9.5.17. Функция <i>vlr-wblock-reactor</i>	777
9.5.18. Функция <i>vlr-window-reactor</i>	779
9.5.19. Функция <i>vlr-xref-reactor</i>	780
9.5.20. Функция <i>vlr-reaction-set</i>	783
9.6. Операции с реакторами	784
9.6.1. Функция <i>vlr-reactions</i>	784
9.6.2. Функция <i>vlr-remove</i>	784
9.6.3. Функция <i>vlr-remove-all</i>	784
9.6.4. Функция <i>vlr-add</i>	785
9.6.5. Функция <i>vlr-added-p</i>	785
9.6.6. Функция <i>vlr-current-reaction-name</i>	785
9.6.7. Функция <i>vlr-data</i>	785
9.6.8. Функция <i>vlr-data-set</i>	785
9.6.9. Функция <i>vlr-owner-add</i>	786
9.6.10. Функция <i>vlr-owner-remove</i>	786
9.6.11. Функция <i>vlr-owners</i>	786
9.6.12. Функция <i>vlr-notification</i>	786
9.6.13. Функция <i>vlr-set-notification</i>	786
9.7. Временные и постоянные реакторы	787
9.7.1. Функция <i>vlr-pers</i>	787
9.7.2. Функция <i>vlr-pers-release</i>	787
9.7.3. Функция <i>vlr-pers-p</i>	788
9.7.4. Функция <i>vlr-pers-list</i>	788
9.8. Правила работы с реакторами	788
9.9. Примеры реакторов	789
9.9.1. Схема создания реакторов	789
9.9.2. Реактор изменений базы рисунка	790
9.9.3. Реактор DXF-операций	791
9.9.4. Реактор команды ERASE	792
9.9.5. Отслеживание двойного щелчка (реактор мыши)	793
9.9.6. Реактор отрезков	794
9.9.7. Связанные окружности (реактор объектов)	797

Глава 10. Приемы программирования **800**

10.1. Рекурсия	800
10.1.1. Функция <i>PL:String-Rep</i>	801
10.1.2. Функция <i>PL:String->List</i>	802
10.1.3. Функция <i>PL:ReadFile</i>	803
10.1.4. Набор функций для работы с регулярными списками	804
10.2. Выбор примитивов методами ActiveX	810
10.2.1. Интерактивный выбор	812
10.2.2. Программный выбор	812

10.3. Особенности обработки семейств объектов	813
10.4. Пример работы с внешним приложением (чтение файла Excel 2003)	815
10.4.1. Анализ методов и свойств приложения Excel	816
10.4.2. Тестовый файл книги Excel	816
10.4.3. Тексты программ	817
10.4.4. Анализ программ	823
Послесловие.....	825
Приложение 1. Алфавитный перечень функций	826
Приложение 2. DXF-коды объектов AutoCAD	844
Общие DXF-коды	844
Символьные таблицы.....	852
Некоторые примитивы	854
Приложение 3. Индикатор.....	861
Приложение 4. Светофор	864
Приложение 5. Танцы панелей.....	870
Приложение 6. Пересечение объектов	878
Приложение 7. Очистка расширенных данных	881
Приложение 8. Метод <i>CopyObjects</i>.....	884
Приложение 9. Описание компакт-диска.....	890
Перечень материалов.....	890
Использование материалов	892
Литература	893
Предметный указатель	894

ГЛАВА 1



Языки AutoLISP и Visual LISP

Язык программирования AutoLISP — это вариант языка LISP, в который добавлены функции доступа к объектам, таблицам и словарям системы AutoCAD. Основой языка LISP является работа со списками, которые могут иметь произвольную длину и включать элементы разной природы (числа, текстовые строки, указатели файлов и т. д.). В то же время в языке доступны обычные арифметические вычисления, логические операции, работа с файлами и т. д. С помощью языка AutoLISP можно писать программы или вводить в командной строке выражения, которые затем вычисляет система AutoCAD. Для чтения и разбора выражений в систему AutoCAD встроен интерпретатор AutoLISP.

За последние годы язык развился за счет новых функций с префиксом `v1-`, а также функций, предоставляющих доступ к объектной модели AutoCAD и реализующих элементы технологии ActiveX. Такие функции (довольно условно) отнесены к языку Visual LISP. Для простоты объединение языков AutoLISP и Visual LISP будем называть просто LISP.

В данной главе изучаются основные функции языка LISP и приводятся практические примеры. В *главах 3 и 4* будет рассмотрено применение LISP в файлах меню и программирование диалоговых окон с помощью LISP и языка DCL.

Для разработки и отладки программ на LISP удобно пользоваться входящей в систему AutoCAD интегрированной средой разработки приложений, изучаемой в *главе 2*.

Материал излагается по состоянию версии AutoCAD 2006. Изменения, связанные с версиями 2005 и 2006, как правило, в тексте подчеркнуты особо.

1.1. Интерпретатор LISP

В AutoCAD входит интерпретатор LISP, начинающий работать, если вы в командной строке в ответ на стандартную подсказку **Command:** (*Команда:*) введе-

те открывающую скобку — "("). В этот момент AutoCAD настраивается на *ввод и вычисление LISP-выражения*, которое должно иметь форму списка, т. е. начинаться открывающей скобкой и заканчиваться закрывающей, а элементы списка должны отделяться друг от друга пробелами. Например, для получения суммы трех чисел вы можете ввести:

```
(+ 62.8 37.2 1923.5)
```

Как только вы завершите ввод своего выражения нажатием клавиши <Enter> или пробелом, AutoCAD вычислит указанное выражение и выдаст результат (рис. 1.1): 2023.5 (напомним, что десятичная точка в вещественных числах играет роль разделителя между целой и дробной частями).

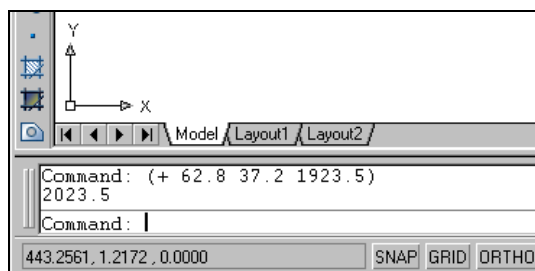


Рис. 1.1. Использование вычислителя выражений интерпретатора LISP

Замечания

- Клавиатурный ввод выражений LISP встречается сравнительно редко. Обычно выражения LISP записываются в текстовый файл с расширением *lsp*, который затем загружается в AutoCAD с помощью функции *load* (см. разд. 1.2.4).
- При работе с длинными числами система AutoCAD отображает в командной строке усеченный результат (обычно не более шести значащих цифр). При этом большие числа показываются в научной форме, например: $7.7791e+007$. Однако если результат вычисления выражения сохраняется в переменной (см. далее), то он имеет максимально возможную точность.

1.1.1. Вычисляемые выражения

В приведенном примере для вычисления было предложено выражение, открывающееся и закрывающееся круглыми скобками и состоящее из четырех элементов: первый — имя функции сложения (+), остальные — аргументы функции, т. е. слагаемые (62.8, 37.2 и 1923.5). Это иллюстрирует форму вычисляемых выражений языка AutoLISP:

```
(<функция> [<аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Здесь <функция> — это имя функции, <аргумент1>, <аргумент2>, ..., <аргументN> — аргументы функции, разделяемые, по крайней мере, одним про-

белом. Квадратные скобки в приведенной форме указывают на возможное отсутствие находящихся между ними аргументов. Количество аргументов зависит от синтаксиса функции. Существуют функции, которые вызываются без аргументов. Алфавитный список базовых функций LISP (без функций методов и свойств, рассматриваемых в главах 7 и 8) приведен в *приложении 1*.

С точки зрения LISP все, что заключено в круглых скобках, является списком, который следует обработать как вычисляемое выражение.

Основное правило AutoLISP — баланс скобок, т. е. количество открывающих скобок должно быть равно количеству закрывающих. Иначе AutoCAD, обнаружив дисбаланс (например, если пользователь забыл ввести конечную скобку), выдаст об этом сообщение (рис. 1.2).

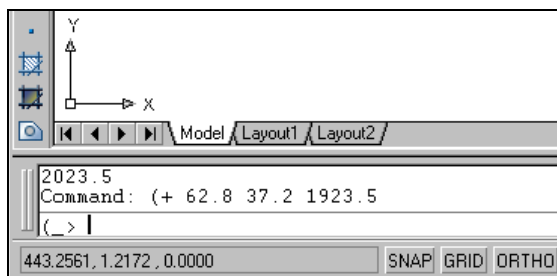


Рис. 1.2. Сообщение о дисбалансе скобок

Для разрешения ситуации, изображенной на рис. 1.2, необходимо ввести закрывающую скобку и нажать клавишу <Enter> или пробел. После этого вы получите результат вычисления. В интегрированной среде разработки Visual LISP (*см. главу 2*) предусмотрены механизмы проверки скобок в файлах с текстами программ.

В качестве любого аргумента функции может быть введено некоторое вычисляемое выражение, ограниченное скобками, или символ LISP (о символах *см. разд. 1.2*).

Выражения LISP могут быть как простыми, так и составными, т. е. в них могут в качестве элементов использоваться другие выражения, причем разделяющими знаками в выражениях являются круглые скобки, пробелы и двойные кавычки (двойные кавычки должны ограничивать с двух сторон текстовые константы — о текстовых или строковых константах *см. разд. 1.2*). Например:

```
(- (+ 14.2 3.09) (* 12.5 0.93))
```

Здесь "-" — имя функции (вычитание), имеющей два аргумента в виде выражений, первое из которых является выражением с операцией сложения (в нем использована функция сложения "+"), а второе — выражением с операцией умножения (в нем использована функция умножения "*"). Сначала интерпретатор языка LISP вычислит выражения во внутренних скобках (в нашем примере полу-

чатся значения 17.29 и 11.625), а затем подставит эти значения и вычислит выражение $(- 17.29 11.625)$, что должно дать результат 5.665 (вещественное число).

1.1.2. Функция *setq* и точность отображения

Интерпретатор LISP вычисляет значение каждого введенного выражения и возвращает его в среду AutoCAD. *Возвращаемое значение*, в свою очередь, либо может быть передано для дальнейших вычислений в выражение более высокого уровня, как в последнем примере *разд. 1.1.1* (аргументы функции верхнего уровня были вычисляемыми выражениями), либо может быть сохранено в *переменной* (человеку, знакомому с программированием, ясно, что это — именованные участки памяти для хранения данных) с помощью функции присвоения *setq*, имеющей следующий синтаксис:

```
(setq <переменная1> <выражение1> [<переменная2> <выражение2> ...  
[<переменнаяN> <выражениеN>] ... ])
```

Функция *setq* может использоваться с любым количеством аргументов, которое должно быть обязательно четным и не менее двух. В качестве аргументов *<переменная1>*, *<переменная2>* и т. д. могут использоваться любые символы (имена) LISP, не занятые в наименованиях функций. О символах речь пойдет в *разд. 1.2*. В качестве аргументов *<выражение1>*, *<выражение2>* и т. д. могут использоваться выражения LISP, имена переменных или константы (например, числа).

Функция *setq* — основное средство для сохранения значений, возвращаемых другими выражениями. Пример:

```
(setq myd (- (+ 6.832 200.1) (+ 423.1 33.22)))
```

Значение выражения $(- (+ 6.832 200.1) (+ 423.1 33.22))$ сохранено в переменной *myd*.

Внимание!

Следует иметь в виду, что значение, отображенное в командной строке, может не совпасть с реальным значением выражения. Вызвано это тем, что по умолчанию в вещественных числах система AutoCAD **показывает в командной строке не более шести значащих цифр**. Например, если бы в итоге вычисления должен был бы получиться результат 329.65142, то все равно вы бы увидели в командной строке только 329.651. Однако внутри программы AutoCAD оперирует числами с **максимальной** точностью (15 знаков после десятичной точки).

Иллюстрация к сделанному замечанию. Попробуйте ввести в командной строке выражение

```
(setq w 329.65142)
```

Вы увидите, что AutoCAD в качестве результата покажет вам 329.651. Но это совсем не означает, что система испортила значение переменной *w*.

Для проверки введите еще одно выражение:

```
(- w 329.0)
```

И получите результат с "потерянными" значащими цифрами: 0.65142.

В дальнейшем можно прочесть значение вычисленной ранее переменной *w* (не забывая о точности отображения!) с помощью операции `!` (восклицательный знак) и имени переменной. Если в командной строке AutoCAD в ответ на стандартную подсказку **Command:** (*Команда:*) ввести `!w`, то AutoCAD выдаст текущее значение этой переменной (в нашем примере: 329.651).

1.1.3. Комментарии

В текст программы полезно вставлять *комментарии*, поясняющие, например, особенности алгоритма. Комментарием считается часть строки, начиная с символа ";" (точка с запятой), после которого не следует сразу же символ "|" (вертикальная черта), и кончая последним символом строки. Закомментированной может быть вся строка программы.

Если комментарий нужно вставить в середину строки, тогда началом комментария должны быть символы "; |", а концом комментария — символы "|;" (причем конец комментария может располагаться на другой строке). Примеры (комментарии выделены полужирным шрифтом):

```
; Закомментирована вся строка  
(setq two_pi (+ pi pi)) ; Закомментирован конец строки  
(setq uv0 (+ 44.73 | внутренний  
 комментарий |;206.7))  
(command "_ .CIRCLE" '(14.0 6.6) 150.55) ; рисование окружности
```

1.2. Символы

Символы LISP — это слова, состоящие из группы буквенно-цифровых знаков и являющиеся именами функций или переменных, используемых для хранения данных пользователя. В символах LISP не должны использоваться разделители выражений, к которым относятся круглые скобки, пробелы и двойные кавычки, и служебные знаки — "." (точка), "," (запятая), ";" (точка с запятой), "'" (апостроф), "/" (косая черта), "\" (обратная косая черта). Написание символов не должно соответствовать написанию чисел (например, символ не должен состоять из одних цифр).

Зарезервированными являются символы, используемые как имена функций, а также некоторые обозначения констант (*см. разд. 1.2.2*). Алфавитный перечень базовых функций языка AutoLISP приведен в *приложении 1*.

Замечание

В именах символов LISP регистр (верхний или нижний) не имеет значения. Поэтому `setq`, `SETQ`, `SETq`, `Setq`, `sEQ` обозначают одну и ту же функцию `setq`.

1.2.1. Типы данных

LISP работает с данными следующих типов:

- ◆ целое число;
- ◆ вещественное число;
- ◆ строка;
- ◆ список, точечная пара, точечный список;
- ◆ дескриптор файла;
- ◆ указатель функции, приложения;
- ◆ примитив AutoCAD;
- ◆ набор.

Кроме того, в главах, посвященных технологии ActiveX, описываются еще некоторые типы данных (варианты, безопасные массивы, VLA-объекты, VLR-объекты).

Замечание

Тип символа может быть получен с помощью функции `type` (см. разд. 1.7.1).

Символам LISP могут быть присвоены значения любых типов (этим язык LISP выгодно отличается от языков высокого уровня типа Fortran). Символ языка LISP может в ходе программы менять тип своего значения. Пользователь в явном виде может задавать константы следующих типов: целые, вещественные, строковые и списковые.

Целые числа могут быть как положительными, так и отрицательными или нулями. Диапазон допустимых значений — от $-2\,147\,483\,648$ до $+2\,147\,483\,647$.

Замечание

Функция `getint`, запрашивающая ввод целого числа, позволяет использовать значения только в диапазоне между $-32\,768$ и $+32\,767$. Аналогичный диапазон значений целых чисел допускался в языке LISP ранних версий.

К целым числам относятся слова, состоящие лишь из цифр. Кроме того, целые числа могут иметь знак (+ или -). Если пользователь пытается задать целое число, не лежащее между описанными выше предельными значениями, то вводимое значение преобразуется в вещественное.

Вещественные числа — это слова, которые состоят из цифр и содержат внутри только одну десятичную точку. Могут иметь знак (+ или -). В состав веществен-

ного числа может входить латинский символ *e* или *E* с показателем степени числа 10 (научная форма представления вещественных чисел).

Вещественные числа не должны начинаться с десятичной точки (например, вместо .5 следует писать 0.5). Допустимые значения вещественных чисел находятся в диапазоне от $-1.7976931348623158e+308$ до $+1.7976931348623158e+308$.

Строковые константы начинаются и заканчиваются кавычками (сами кавычки в длину константы не входят) и содержат внутри себя текст, который может состоять из любых знаков, в том числе пробелов, скобок, апострофов. Особо рассматривается знак "\", который является служебным и предназначен для ввода в текстовую строку некоторых специальных знаков (например, кавычек и самого знака "\"). Константа "" считается строкой нулевой длины. В строковые константы могут входить управляющие знаки печати, например: "\n" — переход на новую строку, "\t" — символ табуляции.

Списковые константы — это списки, состоящие из постоянных элементов любого типа (списки, как и функции, открываются и закрываются круглыми скобками). Для того чтобы AutoCAD не вычислял выражение как функцию, постоянные списки предваряются апострофом перед открывающей скобкой.

Примеры констант:

- ◆ 91
- ◆ -25408
- ◆ 539.3582
- ◆ 777.5e+3
- ◆ "Версия для печати"
- ◆ "336_5" (первая косая черта является служебной, вторая — значащим символом; если распечатать такую строку, то получим 336_5)
- ◆ '(0.0 0.0 1.0)
- ◆ '("Aa" "Bb" "Cc" "Dd" "Ee")
- ◆ '(0 75 "chair" 25 "window" 3 "" "&")

Особые списки, в которых только два элемента и разделителем между ними является точка, называются *точечными парами*. Они создаются с помощью функции cons. Примеры точечных пар:

- ◆ (0 . "POLYLINE")
- ◆ (62 . 29)

Списки, которые состоят из трех или более элементов и у которых последний элемент отделяется точкой, называются *точечными списками*. Они могут создаваться с помощью функции vl-list*. Пример:

- ◆ (120 14 . 67)

Функция `setq` возвращает значение, совпадающее со значением выражения, присваиваемого последней переменной (`<переменнаяN>`), поэтому тип результата совпадает с типом этого выражения.

Есть функции, которые преобразуют значение из одного типа в другой (например, функции `atoi`, `fix`, `float`, `rtos` и др.). Эти функции рассматриваются в *разд. 1.7*.

1.2.2. Предопределенные символы

К основным зарезервированным символам, значения которых пользователям не рекомендуется изменять, помимо имен функций, относятся следующие:

- ◆ `t` — единственная логическая константа, имеющая значение "истина";
- ◆ `nil` — пустое значение (оно же в логических операциях играет роль и значения "ложь"); это значение не эквивалентно числу 0 или пустой строке ("");
- ◆ `pi` — постоянная, равная отношению длины окружности к диаметру (3.14159...);
- ◆ `pause` — постоянная, позволяющая ввести паузу при работе команды AutoCAD (используется в функции `command`).

Представленные четыре символа называются еще *предопределенными символами*.

Замечание

Функциям, требующим в качестве аргумента логическое значение "истина", вместо `t` можно передавать любое значение, отличное от `nil` (например, число, строку и т. д.).

1.2.3. Переменные

Для хранения данных пользователь может вводить свои символы (переменные), не совпадающие по написанию с зарезервированными или ранее занятыми. Создание новых переменных осуществляется с помощью функции `setq`, например:

```
(setq r1 15.33 s24 9)
```

Здесь вводятся переменные `r1` и `s24`, получающие, соответственно, значения 15.33 (вещественное) и 9 (целое). При этом к той части оперативной памяти, которая отведена для текущего рисунка, добавляются участки, занимаемые переменными. Если переменной присвоить значение `nil`, то такая переменная из памяти удаляется, и ее место освобождается для других операций LISP.

Замечание

В многодокументном режиме (когда в одном сеансе AutoCAD открыто несколько рисунков) переменные одного рисунка не доступны в другом рисунке. Для передачи

требуется экспортировать переменные во внедокументное пространство имен (см. разд. 1.17.33).

Переменные могут использоваться в любых выражениях, например:

```
(setq r61 (+ j 81.4 ysa))
```

1.2.4. Функция *load*

Ввод с клавиатуры выражений LISP не всегда удобен, поэтому можно записать предварительно свои строки на языке LISP в файл с расширением *lsp* (текст располагается в файле произвольным образом построчно, не обязательно каждое новое выражение начинать с новой строки; желательно заканчивать строку на разделителе — пробеле или скобке), а затем загрузить этот файл с помощью функции *load*. Синтаксис функции следующий:

```
(load <файл> [<ошибка>])
```

Здесь <файл> — это строка с именем загружаемого файла, а <ошибка> — это выражение, которое нужно выполнить, или значение (например, строка с текстом сообщения), которое нужно вывести, если загружаемый файл не будет обнаружен. Возвращаемое значение в случае удачной загрузки — результат последнего выражения или имя последней функции, если в файле находятся определения функций (см. функцию *defun*). В случае если загружаемый файл не найден, то выполняется <ошибка> (оно может быть текстом сообщения об ошибке).

Например:

```
(load "e25run.lsp" "Файл не найден")
```

Аргумент <ошибка> может быть опущен, в этом случае при ошибке загрузки выдается стандартное сообщение AutoCAD. Аргумент <файл> обычно содержит имя файла, который ищется в соответствии с настройкой рабочего каталога и путей поиска файлов поддержки. Однако можно указывать и полное имя с путем — например, "c:/D_Progs/e25run.lsp" или "c:\\D_Progs\\e25run.lsp". В качестве разделителя каталогов может использоваться косая черта или обратная косая черта, но обратная черта должна обязательно удваиваться (см. замечание к строковым константам в разд. 1.2.1).

Если расширение в имени файла опущено, то AutoCAD пытается выполнить загрузку, поочередно добавляя следующие расширения: *vlx*, *fas*, *lsp*.

Расширение *vlx* соответствует файлам VLX-приложений, *fas* — компилированным LISP-файлам или файлам FAS-приложений, *lsp* — исходным LISP-файлам. Для непосредственного редактирования пользователю доступны только файлы с расширением *lsp*, т. к. файлы с расширениями *vlx* и *fas* не являются текстовыми. О компилированных файлах (с расширением *fas*) и файлах приложений (с расширениями *fas* и *vlx*) см. главу 2.

1.2.5. Функция *vl-load-all*

Система AutoCAD разрешает работу одновременно с несколькими открытыми документами (рисунками). Поэтому возникает проблема, загружен ли файл с расширением `lsp` в конкретный рисунок. Функция `load` выполняет загрузку только в текущий документ. Для того чтобы обеспечить загрузку LISP-файла во все уже открытые и открываемые в данном сеансе AutoCAD рисунки, следует пользоваться функцией `vl-load-all`:

```
(vl-load-all <файл>)
```

Аргумент `<файл>` — текстовая строка с именем файла (расширение файла может быть тем же, что и у первого аргумента функции `load`). Возвращаемого значения нет, если загрузка не выполнялась. Возвращается значение `t` в случае успешной загрузки.

См. также *разд. 1.17.29*.

1.3. Выполнение команд AutoCAD

Очень важной является функция `command`, которая имитирует ввод пользователя в командной строке AutoCAD с клавиатуры. Аналогичную роль играет функция `vl-cmdf` (различия между ними разбираются в *разд. 1.3.2*).

1.3.1. Функция *command*

Синтаксис функции:

```
(command [<параметр1> [<параметр2> ... [<параметрN>] ... ]])
```

Возвращаемое значение — `nil`.

Функция может вызываться без параметров — это равносильно нажатию клавиши `<Esc>`, что прерывает действие активной команды. Если параметры используются, то их количество может быть любым. Каждый из параметров чаще всего является текстовой строкой, имитирующей набор символов строки на клавиатуре с нажатием клавиши `<Enter>`.

Замечание

Если в качестве параметров функции `command` передаются выражения LISP, то среди этих выражений не должны фигурировать функции интерактивного ввода (`getint`, `getreal`, `getpoint` и т. д. — см. *разд. 1.10*). Если указанное ограничение было в программе нарушено, то результат операции непредсказуем. Следует вынести использование функций интерактивного ввода перед выражением с функцией `command`.

Предположим, необходимо нарисовать отрезками рамку формата размером 210 на 297 (левый нижний угол рамки взять в точке (0,0) и верхний правый — в точке

(210,297)). Вспомним, что команда рисования отрезков в AutoCAD называется LINE (ОТРЕЗОК). Если вы не уверены, что ваша программа всегда будет работать на компьютере с русской версией AutoCAD, то используйте английские имена команд с подчеркиванием (в данном случае — `_LINE`). С такими именами команды будут выполняться в любой версии системы. Кроме того, перед или после символа подчеркивания желательно ставить точку (`_.LINE` или `._LINE`), если вы не хотите работать с переопределенной командой (а возможность переопределения стандартных команд в системе есть).

Если бы вы выполняли задачу без программы, то вводили с клавиатуры следующие команды и опции:

- ◆ `_.LINE`
- ◆ `0,0`
- ◆ `210,0`
- ◆ `210,297`
- ◆ `0,297`
- ◆ `_C`

По этим указаниям AutoCAD вызовет команду LINE, возьмет в качестве первой точки (0,0), в качестве второй — (210,0), в качестве третьей — (210,297), в качестве четвертой — (0,297), а `_C` (соответствует опции **Close** (Замкнуть)) замкнет отрезки на первую точку (опции тоже желательно вводить английские и с подчеркиванием). Программа, имитирующая описанный алгоритм, занимает одну строку и выглядит так, как показано в листинге 1.1.

Листинг 1.1. Построение прямоугольника (вариант 1)

```
; Глава 01\Листинг 1.01.lsp  
(command "_.LINE" "0,0" "210,0" "210,297" "0,297" "_C")
```

Можно указанное построение выполнить в несколько строк (в несколько обращений к функции `command`, как в листинге 1.2).

Листинг 1.2. Построение прямоугольника (вариант 2)

```
; Глава 01\Листинг 1.02.lsp  
(command "_.LINE" "0,0" "210,0")  
(command "210,297" "0,297")  
(command "_C")
```

В листинге 1.2 использовано трехкратное обращение к функции `command`, в каждом из которых количество параметров должно быть не меньше одного.

Можно координаты точек задавать не как строку, а как списки из двух чисел целого или вещественного типа (листинг 1.3).

Листинг 1.3. Построение прямоугольника (вариант 3)

```
; Глава 01\Листинг 1.03.lsp
(command "_.LINE" ' (0 0) ' (210 0) ' (210 297) ' (0 297) " _C")
```

Апостроф опускать нельзя, поскольку выражение (210 0) без апострофа будет рассматриваться как обращение к функции 210 с аргументом 0.

Там, где требуется ввод номеров (чисел), можно в качестве аргументов передавать функции `command` именно числа, а не строки с числами, например:

```
(command "_.COLOR" "1")
(command "_.COLOR" 1)
```

Оба выражения выполняют одну и ту же операцию (в качестве текущего устанавливают красный цвет, который имеет номер 1), но в первом случае номер цвета задается как строка, а во втором — как целое число.

Можно в одной функции `command` обратиться сразу к нескольким командам системы AutoCAD, например:

```
(command "_.COLOR" 1 "_.ZOOM" " _E")
```

Эта программная строка выполнит команду `COLOR` (ЦВЕТ) для установки красного цвета и команду `ZOOM` (ПОКАЗАТЬ) с опцией **Extents** (Границы).

Для имитации варианта, когда пользователь вместо ввода опции нажимает клавишу <Enter>, в качестве параметра функции `command` выступает текстовая строка нулевой длины (""). Например, если бы вы в варианте, приведенном в листинге 1.1, захотели завершить команду `LINE` не опцией `_C` (замкнуть прямоугольник), а нажатием клавиши <Enter> (не замыкать прямоугольник), то соответствующая программная строка выглядела бы так:

```
(command "_.LINE" "0,0" "210,0" "210,297" "0,297" "")
```

1.3.2. Функция *vl-cmdf*

Функция `vl-cmdf` имеет следующий синтаксис:

```
(vl-cmdf [<параметр1> [<параметр2> ... [<параметрN>] ... ]])
```

Аргументами могут быть любые выражения, которые нужно передать в командную строку AutoCAD (в том числе команды, опции команд, данные, выражения LISP и т. д.). В отличие от функции `command`, в качестве параметров функции `vl-cmdf` могут фигурировать выражения LISP, содержащие функции интерактивного ввода (`getint` и т. п.).

Отсутствие аргументов функции равносильно нажатию пользователем клавиши <Esc>.

Возвращаемое значение — `t`, если все выражения, переданные в командную строку, выполнились без сбоев, и `nil` — при обнаружении ошибок.

Основное отличие функции `v1-cmdf` от функции `command` заключается в том, что проверка выполнимости команд AutoCAD с заданными параметрами осуществляется до реального начала их исполнения. И если обнаруживается какая-то ошибка, то ни один из параметров функции `v1-cmdf` не выполняется системой AutoCAD, и генерируется сообщение об ошибке приложения. Это исключает возможность частичной работы любых команд AutoCAD (т. е. когда выполняются первые операции, а затем команда выдает сбой).

Функция `command` по одному передает параметры в командную строку, и AutoCAD по одному пытается их выполнить.

В примерах, приведенных в листингах 1.1—1.3, функцию `command` можно было бы заменить на функцию `v1-cmdf`.

1.4. Арифметические функции

Самыми простыми являются *арифметические функции*, они рассматриваются в следующих разделах. Тип результата арифметической функции зависит от типов ее аргументов. Если все аргументы одного типа, то и результат будет того же типа. Если среди аргументов встречаются как целые, так и вещественные, то результат будет вещественным.

Для каждой функции указывается синтаксис, количество и типы аргументов, а также тип возвращаемого значения.

1.4.1. Функция `+`

Вычисляет сумму чисел.

```
(+ [<число1> [<число2> ... [<числоN>] ... ]])
```

Типы аргументов: целые или вещественные числа.

Тип возвращаемого значения: целое, если все аргументы целые; вещественное, если хотя бы один аргумент является вещественным.

Если количество аргументов равно нулю, то возвращается 0 (целое). Если количество аргументов равно 1, то возвращается значение, равное значению аргумента.

Примеры:

- ◆ `(+ 16 30)` возвращает 46
- ◆ `(+)` возвращает 0
- ◆ `(+ -2.0 12.1)` возвращает 10.1
- ◆ `(+ 2 13 1.00)` возвращает 16.0

1.4.2. Функция –

Вычисляет разность чисел.

(- [<число1> [<число2> ... [<числоN>] ...]])

Типы аргументов: целые или вещественные числа.

Тип возвращаемого значения: целое, если все аргументы целые; вещественное, если хотя бы один аргумент является вещественным.

Если аргументы не заданы, то возвращается 0 (целое). Если количество аргументов равно 1, то возвращается значение, вычитаемое из 0 (т. е. <число1> с противоположным знаком). Если количество аргументов равно 2 и более, то из первого числа вычитается сумма всех остальных.

Примеры:

- ◆ (- 3) возвращает -3
- ◆ (-) возвращает 0
- ◆ (- -4 2.1 2.3) возвращает -8.4

1.4.3. Функция *

Вычисляет произведение чисел.

(* [<число1> [<число2> ... [<числоN>] ...]])

Типы аргументов: целые или вещественные числа.

Тип возвращаемого значения: целое, если все аргументы целые; вещественное, если хотя бы один аргумент является вещественным.

Если количество аргументов равно нулю, то возвращается 0 (целое). Если количество аргументов равно 1, то возвращается значение аргумента. Если количество аргументов равно 2 и более, то выполняется перемножение всех аргументов.

Примеры:

- ◆ (* 7) возвращает 7
- ◆ (*) возвращает 0
- ◆ (* 5 1.1 -0.3) возвращает -1.65

Замечание

При перемножении больших целых чисел следует учитывать, что произведение может выйти за границы допустимых значений и функция выдаст непредсказуемый результат. Так, выражение `(* 12345 987654)` вернет значение `-692313258`, что заведомо не может быть правильным, т. к. произведение положительных чисел должно быть положительным. В таких случаях числа следует перемножать в вещественном виде.

1.4.4. Функция /

Вычисляет частное чисел.

```
(/ [<число1> [<число2> ... [<числоN>] ... ]])
```

Типы аргументов: целые или вещественные числа.

Тип возвращаемого значения: целое, если все аргументы целые; вещественное, если хотя бы один аргумент является вещественным.

Если количество аргументов равно нулю, то возвращается 0 (целое). Если количество аргументов равно 1, то возвращается значение аргумента. Если количество аргументов равно 2 и более, то выполняется деление первого аргумента на произведение всех остальных.

Примеры:

- ◆ (/ 81) возвращает 81
- ◆ (/) возвращает 0
- ◆ (/ 5 1.1 -0.3) возвращает -15.1515 (реальный результат будет получен с максимальной точностью, но на экран обычно выводятся шесть значащих цифр)
- ◆ (/ 17 23) возвращает 0
- ◆ (/ -17 23) возвращает 0

Замечание

При делении целых чисел результат должен быть того же типа, что и аргументы, т. е. целым, поэтому он усекается до целого (путем отбрасывания дробной части). Этим объясняется, что выражения (/ 17 23) и (/ -17 23) возвращают значение 0.

1.4.5. Функция 1+

Увеличивает аргумент на 1.

```
(1+ <число>)
```

Тип аргумента: целое или вещественное число.

Тип возвращаемого значения: целое, если аргумент целый; вещественное, если аргумент является вещественным.

Примеры:

- ◆ (1+ 2) возвращает 3
- ◆ (1+ 5.2) возвращает 6.2
- ◆ (1+ -5.2) возвращает -4.2

1.4.6. Функция 1–

Уменьшает аргумент на 1.

(1– <число>)

Тип аргумента: целое или вещественное число.

Тип возвращаемого значения: целое, если аргумент целый; вещественное, если аргумент является вещественным.

Примеры:

- ◆ (1– 11) возвращает 10
- ◆ (1– 1.17) возвращает 0.17

1.4.7. Функция ~

Вычисляет дополнение числа до -1 (т. е. сумма возвращаемого значения функции и аргумента <целое> равна -1).

(~ <целое>)

Тип аргумента: целое число.

Тип возвращаемого значения: целое.

Примеры:

- ◆ (~ 9) возвращает -10
- ◆ (~ 0) возвращает -1

1.4.8. Функция *abs*

Вычисляет абсолютную величину числа.

(abs <число>)

Тип аргумента: целое или вещественное число.

Тип возвращаемого значения: целое, если аргумент целый; вещественное, если аргумент является вещественным.

Примеры:

- ◆ (abs 3) возвращает 3
- ◆ (abs -1.1) возвращает 1.1
- ◆ (abs 0.0) возвращает 0.0

1.5. Логические функции

В следующих разделах рассматриваются функции логических и условных операций и проверок. Здесь же приводятся функции `while` и `repeat`, позволяющие организовывать циклы.

1.5.1. Функция `=`

Проверяет равенство аргументов (рекомендуется использовать для чисел или строк).

```
(= <аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: любые.

Тип возвращаемого значения: логическое (`t`, если значения аргументов совпали, и `nil`, если не совпали).

Если задан только один аргумент, то возвращается `t`. Если количество аргументов 2 или больше, то при совпадении значений всех аргументов возвращается `t`, в противном случае — `nil`. При сравнении строк `t` выдается только в случае, когда длины всех строк равны и все строки посимвольно совпадают. При сравнении чисел `t` выдается также в том случае, когда одни аргументы целые, а другие вещественные, но их значения совпадают.

Примеры:

- ◆ `(= -17.0 -17 -17.000)` возвращает `t`
- ◆ `(= 1)` возвращает `t`
- ◆ `(= "33")` возвращает `t`
- ◆ `(= nil)` возвращает `t` (потому что один аргумент)
- ◆ `(= 33 "33")` возвращает `nil`
- ◆ `(= nil nil)` возвращает `t`
- ◆ `(= nil t)` возвращает `nil`
- ◆ `(= t t)` возвращает `t`
- ◆ `(= t 1)` возвращает `nil`
- ◆ `(= 2 30)` возвращает `nil`
- ◆ `(= "AP8" "Ap8" "AP8")` возвращает `nil`
- ◆ `(= "2abc" "2abc")` возвращает `t`

См. также функции `eq` и `equal`.

1.5.2. Функция /=

Проверяет неравенство аргументов (рекомендуется использовать для чисел или строк).

```
(/= <аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: любые.

Тип возвращаемого значения: логическое (т, если значения каких-то аргументов не совпали, и nil, если совпали). Если задан только один аргумент, то возвращается т.

Примеры:

- ◆ (/= 5) возвращает т
- ◆ (/= -2.0 -2) возвращает nil
- ◆ (/= 32 30) возвращает т
- ◆ (/= "AP8" "Ap8" "AP8") возвращает т
- ◆ (/= "AP8" "AP8" "AP8") возвращает nil
- ◆ (/= "2abc" "2abc") возвращает nil

1.5.3. Функция <

Проверяет расположение аргументов по возрастанию, слева направо.

```
(< <аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: числа (целые, вещественные) или строки.

Тип возвращаемого значения: логическое (т, если значения аргументов расположены по возрастанию, и nil — в противном случае). Для текстовых строк производится посимвольное сравнение, по возрастанию кодов символов¹.

Если задан только один аргумент, то возвращается т.

Примеры:

- ◆ (< -1) возвращает т
- ◆ (< -5.81 0) возвращает т
- ◆ (< 12 12 20) возвращает nil
- ◆ (< "X" "Y" "Z") возвращает т
- ◆ (< "2abc" "2abc") возвращает nil
- ◆ (< "d" "d5") возвращает т
- ◆ (< "d5" "d") возвращает nil

¹ Имеются в виду символы текущей кодовой таблицы. См. функцию *ascii*.

1.5.4. Функция <=

Проверяет расположение аргументов по неубыванию, слева направо.

(<= <аргумент1> [<аргумент2> ... [<аргументN>]...])

Типы аргументов: числа (целые, вещественные) или строки.

Тип возвращаемого значения: логическое (т, если значения аргументов расположены по неубыванию, и nil — в противном случае). Для текстовых строк производится посимвольное сравнение, по неубыванию кодов символов.

Если задан только один аргумент, то возвращается т.

Примеры:

- ◆ (<= -1) возвращает т
- ◆ (<= -5.81 0) возвращает т
- ◆ (<= 12 12 20) возвращает т
- ◆ (<= "X" "Y" "Z") возвращает т
- ◆ (<= "2abc" "2abc") возвращает nil
- ◆ (<= "d" "d5") возвращает т
- ◆ (<= "d5" "d") возвращает nil

1.5.5. Функция >

Проверяет расположение аргументов по убыванию, слева направо.

(> <аргумент1> [<аргумент2> ... [<аргументN>]...])

Типы аргументов: числа (целые, вещественные) или строки.

Тип возвращаемого значения: логическое (т, если значения аргументов расположены по убыванию, и nil — в противном случае). Для текстовых строк производится посимвольное сравнение, по убыванию кодов символов.

Если задан только один аргумент, то возвращается т.

Примеры:

- ◆ (> -1) возвращает т
- ◆ (> -5.81 0) возвращает nil
- ◆ (> 12 12 20) возвращает nil
- ◆ (> "X" "Y" "Z") возвращает nil
- ◆ (> "2abc" "2abc") возвращает nil
- ◆ (> "d" "d5") возвращает nil
- ◆ (> "d5" "d") возвращает т

1.5.6. Функция `>=`

Проверяет расположение аргументов по невозрастанию, слева направо.

```
(>= <аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: числа (целые, вещественные) или строки.

Тип возвращаемого значения: логическое (т, если значения аргументов расположены по невозрастанию, и `nil` — в противном случае). Для текстовых строк производится посимвольное сравнение, по невозрастанию кодов символов.

Если задан только один аргумент, то возвращается т.

Примеры:

- ◆ `(>= -1)` возвращает т
- ◆ `(>= -5.81 0)` возвращает `nil`
- ◆ `(>= 12 12 20)` возвращает `nil`
- ◆ `(>= "x" "y" "z")` возвращает `nil`
- ◆ `(>= "2abc" "2abc")` возвращает т
- ◆ `(>= "d" "d5")` возвращает `nil`
- ◆ `(>= "d5" "d")` возвращает т

1.5.7. Функция `and`

Выполняет операцию логического И.

```
(and [<аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: любые.

Тип возвращаемого значения: логическое (т, если значения всех аргументов истинны, и `nil` — в противном случае). К истинным, помимо т, относятся все значения (числовые, строковые и т. д.), кроме `nil`.

Если аргументы не заданы или задан только один аргумент, то возвращается т.

Примеры:

- ◆ `(and 11)` возвращает т
- ◆ `(and т nil т)` возвращает `nil`
- ◆ `(and т т 4)` возвращает т
- ◆ `(and "x" "y" "z")` возвращает т

1.5.8. Функция `or`

Выполняет операцию логического ИЛИ.

```
(or [<аргумент1> [<аргумент2> ... [<аргументN>]...])
```

Типы аргументов: любые.

Тип возвращаемого значения: логическое (т, если значение хотя бы одного аргумента истинно, и nil — в противном случае). К истинным, помимо т, относятся все значения (числовые, строковые и т. д.), кроме nil.

Если аргументы не заданы или задан только один аргумент, то возвращается т.

Примеры:

- ◆ (or 11) возвращает т
- ◆ (or т nil т) возвращает т
- ◆ (or т т 4) возвращает т
- ◆ (or "x" "y" "z") возвращает т
- ◆ (or nil nil nil) возвращает nil

1.5.9. Функция *not*

Выполняет операцию логического отрицания.

(not <аргумент>)

Типы аргументов: любые.

Тип возвращаемого значения: логическое (т, если значение аргумента равно nil, и nil — в противном случае).

Примеры:

- ◆ (not т) возвращает nil
- ◆ (not nil) возвращает т
- ◆ (not ".NET") возвращает nil
- ◆ (not -64.64) возвращает nil

1.5.10. Функция *null*

Выполняет операцию проверки на nil.

(null <аргумент>)

Типы аргументов: любые.

Тип возвращаемого значения: логическое (т, если значение аргумента равно nil, и nil — в противном случае).

Фактически идентична функции not. Традиционно функция null употребляется для проверки непустоты списков (поскольку пустой список '() идентичен nil) или функций, а not — для аргументов других типов.