



# САПР НА БАЗЕ **AutoCAD** — **КАК ЭТО ДЕЛАЕТСЯ**

Стандарты проектирования  
и разработки

COM-технологии

Связь AutoCAD и Delphi

AutoCAD и базы данных

Иллюстрированные  
XML-меню

Интеграция  
всех разделов проекта

Инженерные расчеты

Элементы  
документооборота

Интеграция САПР и ГИС



УДК 681.3.06  
ББК 32.973.26-018.2  
3-93

**Зуев С. А., Полещук Н. Н.**

3-93 САПР на базе AutoCAD — как это делается. — СПб.:  
БХВ-Петербург, 2004. — 1168 с.: ил.

ISBN 5-94157-344-8

В книге, основанной на собственном опыте авторов, их многочисленных ошибках и находках, раскрывается технология создания комплексной САПР, предназначенной для разработки различных чертежей — от идеи и концепции до выпуска продукта в свет. Разбирается, как и что можно и нужно делать без программирования, какие инструменты разработчика в системе AutoCAD (AutoLISP, Visual LISP, ObjectARX, VBA, Delphi) могут быть использованы, вырабатываются рациональные приемы программирования. Приводятся примеры прикладных программ различного назначения для конечных пользователей и для разработчиков САПР. Рассматриваются элементы документооборота и сочетания геоинформационных технологий с автоматизированным проектированием. Особое внимание уделяется нетрадиционным для AutoCAD технологиям — Delphi, COM-серверам, XML-меню. Основной упор сделан на то, чтобы показать, как это делается практически.

*Для инженеров и программистов, занимающихся вопросами САПР*

УДК 681.3.06  
ББК 32.973.26-018.2

#### **Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Владимир Шабалин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Юрий Рожко</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Игоря Цырульниковой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 02.07.04.

Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 94,17.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02  
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов

в ОАО "Техническая книга"

190005, Санкт-Петербург, Измайловский пр., 29

ISBN 5-94157-344-8

© Зуев С. А., Полещук Н. Н., 2004

© Оформление, издательство "БХВ-Петербург", 2004

# Содержание

<b>Введение.....</b>	<b>21</b>
Особенности книги.....	22
Для кого предназначена эта книга.....	23
Структура книги.....	23
Предупреждения.....	24
Об авторах.....	25
<b>ЧАСТЬ I. РАЗРАБОТКА КОНЦЕПЦИИ САПР .....</b>	<b>27</b>
<b>Глава 1. Подготовка к разработке.....</b>	<b>29</b>
Требуется ли техническое задание.....	30
Что нужно знать об авторских правах.....	30
Как собрать команду разработчиков.....	32
Как организовать работу над проектом.....	33
Хранение информации.....	33
Документирование работ.....	34
Планирование.....	34
Программирование.....	35
Как строить отношения с руководителями.....	35
Как нормировать разработку программных средств.....	36
Пример норм времени на разработку программ.....	39
Пример расчета трудозатрат.....	40
Как определить стоимость работы.....	43
<b>Глава 2. Анализ и постановка задачи .....</b>	<b>44</b>
Какие САПР существуют.....	44
Какая САПР нам нужна.....	46
Какие разделы проектов мы будем автоматизировать.....	46
Кто будет пользователем нашей системы.....	47
Какие версии Windows и AutoCAD мы будем использовать.....	47
Как узнать, что требуется пользователям для работы с AutoCAD.....	48
Анализ проектов.....	48
Общие данные по рабочим чертежам.....	48

Планы зданий.....	49
Трубопроводы и проводники.....	49
Схемы систем.....	50
Рабочее и монтажное проектирование.....	50
Генеральные планы и наружные сети.....	52
Топографические планы.....	52
Условные обозначения в топографии и на генпланах.....	53
Планы коммуникаций.....	53
Профили.....	54
Схемы, узлы, разрезы.....	54
Изучение специфики работы в AutoCAD.....	54
Классификация пользователей.....	55
"Чайники".....	55
"Тетки".....	55
"Обезьяна с гранатой".....	55
"Нормальные пользователи".....	55
"Профи".....	56
"Ламеры".....	56
"Крутые".....	56
Как должна работать наша система.....	57
Просто, это как?.....	57
Удобно, это как?.....	57
Надежно, это как?.....	58
Чем наша система должна отличаться от других.....	58
Какие задачи должна решать наша система.....	59
Составление перечня общих задач.....	59
Составление перечня специальных задач.....	61
Как превратить специальные задачи в общие.....	62
Как определить методы решения задач.....	62
Как придумать "имя собственное" для нашей системы.....	64
Превращение "нашей системы" в "ruCAD".....	64
Логотип.....	65
Стадийность разработки.....	65
Техническое задание.....	65
Эскизный проект.....	66
Технический проект.....	66
Рабочий проект.....	66
Внедрение.....	67
Подготовка окончательного текста технического задания.....	67
Титульная часть.....	67
Информационная часть.....	67
Наименование.....	68
Область применения.....	68
Основание для разработки.....	68
Назначение разработки.....	68
Заказчик.....	69
Исполнитель.....	69
Технические требования к программному изделию.....	69

Требования к функциональным характеристикам .....	70
Требования к надежности.....	70
Условия эксплуатации .....	70
Требования к составу и параметрам технических средств .....	70
Требования к информационной и программной совместимости .....	71
Требования к маркировке и упаковке .....	71
Требования к транспортированию и хранению .....	71
Специальные требования.....	71
Технико-экономические показатели.....	72
Стадии и этапы разработки.....	72
Порядок контроля и приемки .....	72
Приложения.....	73
Подписи и согласования.....	73
<b>Глава 3. Формирование базовых принципов.....</b>	<b>74</b>
Как запускать AutoCAD .....	74
Как использовать шаблоны рисунков .....	76
Стандарты .....	77
О соответствии чертежей на бумаге стандартам СПДС.....	77
Форматы и размеры листа .....	78
Вид основной надписи .....	79
Начертание и размер надписей.....	79
Ширина линий .....	79
Избыточная детализация .....	80
Стандартизация приемов работы с AutoCAD .....	81
Стандарт приемов работы ruCAD для строительного проектирования.....	82
Дополнительные комментарии к положениям стандарта .....	94
Масштаб рисунка .....	94
Управление шириной линий .....	95
Как учитывать работу в пространствах листа и модели.....	96
Нанесение размеров .....	97
Как сохранять и восстанавливать настройки ruCAD.....	98
Что хранить в настройках ruCAD.....	98
О специальных папках Windows.....	99
Где можно хранить настройки.....	100
Системный реестр .....	101
Домашний каталог пользователя .....	102
Общий каталог пользователей.....	102
Где мы будем хранить настройки.....	102
О временных файлах .....	104
Как сохранять и читать настройки .....	104
INI-файлы .....	104
Использование XML .....	105
Политика работы с пользователями в ruCAD.....	106
Как организовать систему папок и документов .....	106
Корневой каталог системы.....	106
Соглашение об именах файлов.....	107

Каталоги AutoCAD.....	107
Каталог Bin.....	108
Каталог Temp.....	108
Каталог Source.....	108
Каталог Samples.....	108
Каталог All Users.....	108
Каталог Current User.....	109
Каталог Application Data.....	109
Регистрация каталогов.....	109
Какие программы потребуются для реализации базовых принципов.....	110
Программа-стартер.....	110
ARX-библиотека.....	110
COM-серверы.....	111
LISP-библиотеки.....	111
<b>Глава 4. Миграция из ранних версий AutoCAD.....</b>	<b>113</b>
Как осуществлять переход под Windows.....	113
Миграция приложений, написанных на AutoLISP.....	114
Миграция внешних приложений, работающих в DOS.....	114
Миграция библиотек блоков.....	114
Шрифты.....	114
Штриховки и типы линий.....	114
Файл acad.pgp.....	115
Меню.....	115
Базы данных.....	115
Как выполнять перекодировку рисунков.....	115
Файлы AutoCAD R10.....	116
Файлы AutoCAD R12.....	116
Что придумано Autodesk.....	117
Утилита Amethyst CAD Converter.....	119
Какие изменения требуется вносить в программы.....	119
Как учитывать особенности локализованных версий AutoCAD.....	119
Как использовать Migration Assistance.....	120
Переход на работу под AutoCAD 2004.....	121
Учет особенностей AutoCAD 2004.....	121
Новинки AutoCAD 2004.....	121
О совместимости приложений.....	123
Не выбрасывайте старую систему AutoCAD.....	124
Какие программы требуется разработать нам.....	124
<b>Глава 5. Итоги части I.....</b>	<b>125</b>
<b>ЧАСТЬ II. АДАПТАЦИЯ AUTOCAD БЕЗ ПРОГРАММИРОВАНИЯ.....</b>	<b>127</b>
<b>Глава 6. Использование блоков.....</b>	<b>129</b>
Какие виды блоков требуется использовать.....	129
Блоки-чертежи.....	129
Блоки-изделия.....	130

Единичные блоки.....	130
Блоки-символы.....	131
Блоки-таблицы .....	131
Какие блоки нам не нужны в системе .....	132
Как правильно создавать блоки .....	132
Как присваивать имена блокам .....	132
Как устанавливать цвет и слой примитивам, входящим в блок .....	133
Как назначать ширину, тип и вес линий .....	133
Как устанавливать точку вставки блока .....	134
Как правильно включать в блоки атрибуты .....	134
Как хранить блоки в каталогах ruCAD .....	134
Как формировать библиотеки блоков .....	135
Какие программы потребуются для работы с блоками .....	136
<b>Глава 7. Использование штриховок и типов линий.....</b>	<b>137</b>
Средства для работы со штриховками.....	137
Средства для работы с типами линий .....	138
Как создавать "лохматые" линии .....	139
Как создавать линии с текстовыми символами.....	141
Особенности использования специальных типов линий .....	141
Какие программы потребуются для работы с типами линий.....	142
<b>Глава 8. Интерфейс пользователя для работы с AutoCAD .....</b>	<b>143</b>
Что хорошо в стандартном интерфейсе AutoCAD .....	143
Командная строка .....	143
Меню .....	143
Панели инструментов.....	144
Центр управления .....	144
Tool Palettes в AutoCAD 2004.....	144
Что плохо в стандартном интерфейсе AutoCAD .....	146
Как мы будем писать главное меню ruCAD .....	146
Какое хотелось бы иметь меню .....	146
Типы файлов меню.....	147
Какие изменения будем вносить в стандартное меню.....	147
Как преодолеть недостатки каскадных меню .....	149
Почему мы отказываемся от слайдовых меню.....	150
Как управлять доступом к пунктам меню .....	151
Использование выражений языка DIESEL .....	151
Использование языка LISP для управления доступом.....	152
Как ставить "галочки" в меню .....	153
Как правильно использовать экранное меню.....	153
Как мы будем писать пункты меню ruCAD.....	155
Правила формирования тегов пунктов меню ruCAD .....	156
Правила формирования текстов пунктов меню ruCAD .....	156
Правила формирования макросов пунктов меню ruCAD .....	157
О сочетании MENUЕCHO и ^P.....	158
Синтаксис наших макросов.....	159

Как создавать панели инструментов.....	161
Синтаксис описаний панелей инструментов .....	161
Как сохраняется конфигурация панелей инструментов.....	163
Как хранить значки для панелей инструментов .....	163
А где брать пиктограммы? .....	165
Как организовать ввод данных.....	166
Как создать иллюстрированное дерево меню .....	166
Какие программы потребуются для реализации интерфейса.....	172

### **ЧАСТЬ III. РАЗРАБОТКА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ДЛЯ ПРИКЛАДНОГО ПРОГРАММИРОВАНИЯ .....173**

#### **Глава 9. Инструменты разработчика в AutoCAD.....175**

Краткий обзор инструментальных средств программирования.....	175
Сравнение AutoLISP и Visual LISP .....	179
Когда можно и нужно использовать VBA.....	180
Преимущества VBA.....	181
Недостатки VBA .....	181
Как мы будем использовать ObjectARX .....	183
Как мы будем использовать Delphi.....	183
Соглашения о кодировании.....	184
Как добиться единообразного внешнего вида программ .....	185
Имена функций .....	185
Вложенные функции.....	186
Имена команд AutoCAD.....	187
Использование комментариев.....	187
Как единообразно именовать файлы .....	189
Как вести дневники проекта .....	191
Как хранить данные.....	191
Редактор меню.....	192
Дополнительные инструменты .....	192

#### **Глава 10. Приемы программирования на Visual LISP.....196**

Как окончательно перейти на Visual LISP .....	196
Обзор стилей программирования.....	197
"Инженерный" стиль .....	197
Особенности функции <i>command</i> .....	198
"Программистский" стиль .....	202
"Объектный" стиль.....	203
Сравнение скорости работы .....	206
Почему бы окончательно не избавиться от <i>command</i> .....	207
Промежуточный диагноз.....	207
Использование объектной модели AutoCAD.....	208
Пример исследования объектной модели .....	209
Как использовать объектную модель .....	212
Особенности многодокументного режима .....	214
Словари .....	216
Реакторы .....	216



Как использовать библиотеки функций и "конечные" программы.....	216
Как использовать библиотеки сторонних авторов.....	219
Как организовать загрузку библиотек и программ.....	220
Советы по предотвращению ошибок.....	222
Ошибки этапа ввода данных.....	222
Контроль ввода.....	222
Опасайтесь двусмысленных предложений.....	224
Выбор объектов и создание наборов примитивов.....	225
Выбор одного примитива.....	228
Ошибки этапа обработки данных и рисования.....	232
Переменные и значения переменных.....	233
Зачем введена функция <i>quote</i> ?.....	237
Контроль типов данных.....	238
Глобальные и локальные переменные.....	242
Откуда берется мусор и как с ним бороться.....	244
Ошибки вычислений.....	245
Что применять: =, <i>eq</i> или <i>equal</i> ?.....	245
<i>NOT</i> и <i>NULL</i> .....	247
Ошибки вызова команд.....	247
Неправильный вызов команды <i>TEXT</i> .....	247
Забываем отключить объектную привязку.....	248
Структура программы.....	250
Варианты структуры программы.....	250
Вариант "один вход — один выход".....	251
Вариант "один вход — несколько выходов".....	251
Программы, использующие диалоговые окна.....	255
Использование диалоговых окон.....	255
Ненаучная классификация диалоговых окон.....	255
О выходе из диалогов.....	260
О цветовой гамме.....	261
Разработка диалогов с использованием <i>DCL</i> .....	261
Пример диалоговой функции.....	261
Управляющие конструкции.....	269
Функция <i>cond</i> .....	269
О лишних функциях <i>progn</i> .....	270
Циклическая обработка списков.....	271
Рекурсия.....	273
Расследование Петра Лоскутова.....	274
Выводы.....	285
Резюме авторов.....	286
Обработка ошибок.....	286
Ошибки прерывания.....	286
Функция <i>*error*</i> .....	287
Как писать функцию <i>*error*</i> .....	288
Функции начала и завершения приложений.....	289
Типовой обработчик ошибок <i>tuCAD</i> .....	292
Проблемы с откатом.....	294
Ошибки свойств.....	296
Ловушки для ошибок.....	297

"Ловля блох" в ActiveX.....	299
Отлов ошибок при отладке .....	300
Использование отладочных сообщений.....	300
Встроенные средства отладки Visual LISP .....	303
Как не стать параноиком.....	303
Создание приложений Visual LISP .....	304
Создание FAS-приложения.....	305
Создание VLX-приложений.....	307
Резюме.....	310
<b>Глава 11. План программирования .....</b>	<b>311</b>
Как установить очередность разработки программ и библиотек.....	311
Какие библиотеки будем создавать в первую очередь.....	312
Какие программы нам понадобятся в первую очередь.....	312
Как выполнять тестирование.....	314
<b>Глава 12. Формирование каркаса ruCAD .....</b>	<b>316</b>
Создание системы каталогов .....	316
Запись в реестр Windows.....	318
Создание временного ярлыка.....	319
Пробный запуск системы и ручная настройка профиля ruCAD .....	320
Создание временного файла acadoc.lsp.....	320
Разработка временного меню для тестирования системы.....	320
Установка библиотеки DOSLib .....	320
Командный файл для регистрации COM-серверов .....	321
<b>Глава 13. Разработка первоочередных библиотечных функций.....</b>	<b>322</b>
Расположение компонентов системы.....	322
Установка свойств рисунка.....	326
Чтение свойств .....	327
Запись свойств.....	327
Особенности AutoCAD 2004 .....	328
Как решить проблему версии 2004 .....	330
Вспомогательные функции .....	331
Сохранение свойств в AutoCAD 2002 .....	332
Основная функция для AutoCAD 2002 .....	334
Работа со свойствами через ActiveX в AutoCAD 2002.....	336
Получение свойств постороннего файла.....	338
Работа с <i>SummaryInfo</i> в AutoCAD 2004 .....	341
Кто виноват?.....	343
Что делать?.....	343
Использование свойств рисунка при инициализации системы.....	347
Командный реактор DWGPROPS.....	349
Обеспечение пропорций .....	352
Настройки системы на масштаб.....	357
Функции для работы с объектной моделью AutoCAD .....	362
Как получить объект.....	362
Безопасные манипуляции .....	366

Как сохранять и восстанавливать собственные данные в файле рисунка.....	367
Работа с меню .....	381
Вывод специальных меню.....	383
Загрузка и выгрузка фрагментных меню.....	383
Функции для управления доступом к меню .....	384
Кошмарная пятница .....	392
Как придется выкручиваться .....	392
Ввод данных .....	394
Традиционные функции ввода данных .....	396
Усовершенствованные функции ввода данных .....	397
Функции выбора примитивов.....	403
Резюме.....	413
<b>Глава 14. Разработка библиотеки функций с использованием ObjectARX .....</b>	<b>414</b>
Как установить ObjectARX 2002 .....	414
Как настроить Microsoft Visual C++ 6.0 .....	415
Постановка задачи .....	415
Создание заготовки библиотеки с помощью ObjectARX AppWizard.....	416
Анализ полученного кода.....	419
Как устроен ARX .....	426
Как работает функция <i>acrEntryPoint</i> .....	426
Как регистрируются функции для Visual LISP .....	427
Как используется буфер результатов.....	428
Работа с ARX-функциями в Visual LISP .....	430
Варианты работы с INI-файлами .....	433
Запуск приложений .....	435
Создание безопасной оболочки для окна сообщений .....	435
Разработка группы функций для вывода диалоговых окон.....	440
Любителям простоты .....	443
<b>Глава 15. Разработка библиотечных функций с использованием Delphi .....</b>	<b>444</b>
Какие библиотеки компонентов мы будем использовать .....	444
Краткое знакомство с COM-технологиями.....	445
Что такое автоматизация в COM.....	446
Как создать внутренний сервер автоматизации.....	447
Разработка первого COM-сервера.....	449
Изменения LISP-библиотеки .....	459
Другие COM-серверы в tuCAD .....	462
Заставка с сообщением для длительных операций .....	462
Диалоговое окно выбора из одинарного списка .....	463
Список из двух колонок с возможностью редактирования .....	464
Двухоконный список.....	466
Диалог с пометкой элементов списка.....	468
Диалоговое окно просмотра текстового файла.....	470
Визуальное редактирование координат полилинии.....	471
Диалоговое окно "Советы дня".....	474
Выбор файла из виртуального дерева .....	476

Системные папки Windows.....	477
Системные папки ruCAD.....	478
Диалоговые окна в EXE-файлах.....	481
Зачем нужно так делать.....	482
Как запустить внешнее приложение в модальном режиме.....	483
Как передать параметры.....	483
Как получить результаты.....	483
Функция ввода строки.....	483
Резюме.....	488
<b>Глава 16. Работа с XML.....</b>	<b>490</b>
Минимальные сведения по XML.....	490
Как создаются XML-документы.....	495
Как работают программы обработки XML.....	498
Что должно делать наше приложение.....	498
Реализация приложения для просмотра XML-меню.....	499
Разработка автономного приложения.....	499
Происхождение и назначение некоторых компонентов, влияющих на логику работы приложения.....	499
Исходный текст модуля.....	502
Разработка COM-сервера в виде DLL.....	519
Тестовое приложение для проверки COM-сервера.....	525
Напишем функции для вызова приложения из Visual LISP.....	527
Работа с XML-таблицами.....	533
Постановка задачи.....	534
XML-таблицы.....	536
Создание XML-таблицы.....	536
Программа редактирования XML-таблиц.....	538
LISP-функции для обработки таблиц.....	550
<b>Глава 17. Разработка диалогового окна выбора файлов.....</b>	<b>559</b>
Чем нас не устраивают стандартные диалоговые окна.....	559
Ограничение навигации.....	560
Предварительный просмотр DWG-файлов.....	560
Комментирование файлов и папок.....	561
Логика получения аннотирующей информации.....	561
Просмотр и редактирование свойств DWG-файлов без AutoCAD.....	562
Разработка диалогового окна.....	563
Просмотр и редактирование расширенного набора свойств файлов не только в файловой системе NTFS.....	575
Создание COM-сервера.....	575
Формирование библиотеки функций для работы с файлами.....	579
<b>Глава 18. Разработка классификатора слоев.....</b>	<b>585</b>
Как реализовать классификатор слоев с использованием XML.....	585
Хранение классификатора в базе данных.....	586
Классификатор слоев в файловой системе.....	586
Какие дополнительные данные можно хранить в классификаторе.....	587

Работа с классификатором.....	589
Разработка программы.....	591
Запуск программы-классификатора.....	615
Формирование библиотеки функций для работы с классификатором.....	615
<b>Глава 19. Работа с базами данных .....</b>	<b>620</b>
Немного об ADO .....	622
Работа с базами данных из Visual LISP .....	622
Подробности технологии ADO.....	623
Нужно ли импортировать библиотеку типов.....	626
Функции для работы с ADO.....	627
Разработка конструктора строки соединения.....	637
Полный пример работы с БД.....	641
Применимость технологии работы с базами данных.....	647
Резюме.....	648
<b>Глава 20. Разработка программы-стартера .....</b>	<b>649</b>
Работа в стартере.....	649
Выбор рабочей версии AutoCAD .....	651
Запуск AutoCAD.....	656
Проблемы с файлом автозагрузки.....	659
Реализация программы-стартера .....	660
Меню приложений .....	674
<b>Глава 21. Рисование формата .....</b>	<b>676</b>
Формирование окружения программы.....	676
Мастер рисования формата.....	677
Выбор размера листа.....	678
Выбор формы основной надписи .....	680
Формирование обозначения документа.....	681
Сохранение и восстановление данных многострочных граф основной надписи.....	683
Заполнение стадии проектирования и количества листов .....	687
Формирование набора подписей.....	687
Формирование наименования организации.....	688
Ввод граф для чертежей изделий.....	693
Завершение работы Мастера.....	693
Реализация COM-сервера .....	695
Разработка LISP-программы.....	697
Создание блоков основных надписей.....	697
Программа рисования формата .....	698
Включение программы в меню .....	707
<b>Глава 22. Завершение разработки главной библиотеки.....</b>	<b>708</b>
Ширина и вес линий.....	708
Функции для геометрических построений.....	715
Учет особенностей систем координат.....	716

Как преобразовывать координаты.....	717
Где трансформировать координаты .....	719
Результат работы функций рисования.....	721
Реализация координатных функций .....	723
Извлечение списка координат вершин .....	724
Как эффективно использовать блоки.....	725
Вставка блока.....	727
Множественная вставка блока .....	728
Различные способы вставки блоков.....	733
Единичная вставка блоков .....	734
Вставка блоков из файлов.....	735
Изменение ширины линий в блоке .....	737
Врезка блоков и текстов в линии.....	746
Рисование объектов .....	751
Создание отрезков .....	751
Создание полилиний .....	752
Создание текстов .....	754
Изменение веса линий .....	756
Создание кругов.....	756
Семейство функций для рисования трасс и линий .....	757
Программная работа с типами линий.....	767
Загрузка программ .....	771
Резюме.....	772
<b>Глава 23. Итоги разработки библиотек .....</b>	<b>773</b>
<b>ЧАСТЬ IV. РАЗРАБОТКА ПРИКЛАДНЫХ ПРОГРАММ .....</b>	<b>775</b>
<b>Глава 24. Разработка набора инструментов для программистов.....</b>	<b>777</b>
Редактирование меню.....	777
Переключатели режимов в меню .....	778
Массовые операции с файлами.....	779
Методика обработки списка файлов.....	780
Нормализация файлов .....	781
Объектный доступ к другому документу .....	784
Конвертирование файлов.....	787
Ну, тупые! .....	790
Как избежать переименования .....	792
Приведение текстового стиля для всех примитивов к определению .....	800
Обработка штриховок.....	803
Настройка оптимальных масштабов штриховок.....	807
Обработка блоков.....	809
Расстановка всех блоков .....	811
Запись списка блоков в файл.....	812
Переопределение указанного блока.....	813
Просмотр информации об объектах рисунка .....	814
Резюме.....	817

<b>Глава 25. Программы общего назначения.....</b>	<b>818</b>
Переключение компоновок .....	818
Быстрое стирание.....	820
Текстовая "лупа".....	822
Быстрое рисование "такого же" объекта.....	823
Программы для быстрого штрихования .....	827
Псевдоштриховка с помощью блока .....	829
Работа со слоями .....	831
Установка слоя по образцу .....	832
Отключение слоя .....	832
Включение всех слоев.....	832
Оставить видимыми указанные слои.....	832
Перелистывание слоев.....	833
Стирание слоя .....	834
Общие средства рисования .....	836
Рисование специальных линий .....	836
Врезка и привязки текстов к линиям .....	838
Рисование контуров.....	838
Рисование прямоугольников.....	839
Рисование текстов различными способами .....	846
Создание и выбор типовых текстов .....	846
Конструктор таблиц.....	848
Заполнение таблиц.....	852
<b>Глава 26. Формирование специализированных программ из универсальных функций.....</b>	<b>855</b>
Эффективные программы длиной в одну строку.....	855
Создание "специального" меню.....	857
Группы команд, использующих одну функцию .....	859
Вставка блоков различных видов .....	859
Рисование трасс .....	860
Рисование любых таблиц .....	860
Как научить пользователя добавлять свои команды.....	862
<b>Глава 27. Примеры программ для архитектурно-строительной части .....</b>	<b>864</b>
Рисование координационных осей .....	864
Рисование стен и перегородок .....	872
Рисование колонн и опор.....	877
Отверстия в стенах и перекрытиях .....	880
Рисование отверстий в стенах .....	881
Составление ведомости отверстий в стенах .....	893
Резюме.....	896
<b>Глава 28. Программы для "генпланистов" и топографов .....</b>	<b>897</b>
Использование городской системы координат .....	897
Названия осей координат .....	898
Работа с координатами объектов .....	899

Ведомость координат.....	901
Несколько способов рисования откосов.....	905
Разбивка кромок .....	906
Рисование трасс .....	911
Рисование дорог.....	915
<b>Глава 29. Несколько программ для сантехников.....</b>	<b>917</b>
Профили .....	917
Технология рисования "аксонометрии" .....	919
Средства рисования аксонометрии в системе tuCAD.....	922
Узлы схем.....	924
Рисование линий схем.....	926
Пересечение линий.....	927
Выноски диаметров .....	928
Рисование элементов трубопроводов в три линии.....	931
Программирование рисования трубопроводов с детализацией .....	934
Организация ввода данных .....	937
Универсальная функция рисования элементов труб.....	941
Программы для рисования вентиляции .....	944
Резюме.....	945
<b>Глава 30. Вывод чертежа на бумагу .....</b>	<b>946</b>
Алгоритм печати .....	947
Шаг 1. Выбор компоновки .....	947
Шаг 2. Выбор устройства печати.....	948
Шаг 3. Выбор формата бумаги .....	948
Шаг 4. Выбор стиля печати .....	949
Шаг 5. Выбор зоны печати .....	949
Шаг 6. Формирование массива листов.....	949
Шаг 7. Печать.....	950
Реализация алгоритма печати.....	950
Как выполняется печать.....	951
Основная функция печати.....	952
Резюме.....	969
<b>Глава 31. Несколько примеров расчетных программ.....</b>	<b>971</b>
Расчеты объемов работ.....	974
Измерение расстояний .....	974
Определение площадей .....	977
Спецкалькуляторы различного назначения .....	978
Математика с числовыми текстами .....	980
Быстрый прикидочный расчет диаметров трубопроводов и воздухопроводов.....	981
Особенности программирования расчетов.....	985
Где брать формулы?.....	987
Алгоритм определения диаметров трубопроводов.....	989
Резюме.....	991



<b>Глава 32. Спецификации оборудования .....</b>	<b>993</b>
Что такое спецификация оборудования .....	996
Обзор методик работы со Спецификациями .....	1000
Ручное составление .....	1000
Использование текстовых процессоров .....	1001
Рисование в AutoCAD .....	1001
Использование электронных таблиц .....	1001
Использование настольных СУБД .....	1002
Специализированные программы .....	1002
Автоматическое определение объемов работ .....	1002
Организация банка данных по оборудованию, изделиям и материалам .....	1004
Структура базы данных .....	1005
Разработка программы .....	1008
Формирование спецификации для конкретного объекта .....	1010
Формирование рабочей спецификации .....	1013
Пополнение и обслуживание банка данных .....	1015
Вывод спецификации на бумагу .....	1015
Рисование спецификации в AutoCAD .....	1016
Подготовка форм .....	1016
Общая схема работы .....	1016
Резюме .....	1022
<b>Глава 33. Элементы документооборота .....</b>	<b>1023</b>
Что требуется исполнителю .....	1023
Что требуется руководителю .....	1024
Что мы уже нечаянно сделали .....	1025
Организация архива электронных калек .....	1026
Организация архива типовых проектных решений .....	1027
Архив топографических планшетов .....	1029
Функции для работы с электронными архивами .....	1030
Как записать кальки в архив .....	1030
Как взять кальки из архива .....	1035
Как использовать наборы калек .....	1037
Работа с типовыми проектными решениями .....	1044
Выбор и вставка типового проектного решения .....	1048
Работа с DWF-файлами .....	1050
Создание DWF .....	1051
Просмотр DWF .....	1052
Использование DWF .....	1052
Как создать DWF на LISP .....	1052
DWF или PDF? .....	1053
Резюме .....	1054
<b>Глава 34. Интеграция САПР и ГИС .....</b>	<b>1055</b>
Возможности ГИС .....	1056
Составные части ГИС .....	1057
Инструментальные средства ГИС .....	1058
Область применения ГИС на базе AutoCAD .....	1060

Об электронных картах .....	1061
Откуда берутся электронные карты?.....	1063
У вас хорошая крыша? .....	1063
Где брать исходные материалы.....	1063
Привязка к системе координат.....	1064
Процесс создания электронной карты .....	1065
Векторизация планшетов.....	1066
Формирование сводного плана .....	1067
Связь изображений с базами данных .....	1067
Вариант пространственной информации в графическом файле.....	1068
Вариант пространственной информации в базе данных .....	1069
Как это сделано.....	1071
Как хранить координаты в базе данных.....	1072
Как создать собственную ГИС-систему .....	1073
Резюме.....	1074
<b>ЧАСТЬ V. ВЫПУСК СИСТЕМЫ В СВЕТ .....</b>	<b>1077</b>
<b>Глава 35. Создание справочной системы .....</b>	<b>1079</b>
Что уже сделано .....	1080
HLP, СНМ или HTML.....	1080
Инструмент для создания справок.....	1081
Публикация в Интернете .....	1081
Дополнительные справочники .....	1082
DWF-справка.....	1082
Растровые справочники.....	1083
Резюме.....	1083
<b>Глава 36. Разработка инсталлятора.....</b>	<b>1084</b>
Коротко о безопасности.....	1085
Как установить систему giCAD вручную .....	1086
Присядем "на дорожку" и подумаем .....	1087
Кто должен устанавливать giCAD.....	1087
Куда копировать файлы .....	1088
Что и куда писать в реестр.....	1088
Разборки с системой AutoCAD.....	1088
Не надо унывать!.....	1089
Как делаются установочные комплекты.....	1089
Подготовка к созданию дистрибутива .....	1089
Уменьшение размеров приложений.....	1090
Перекомпиляция приложений.....	1091
Упаковка файлов.....	1091
Упаковка DWG-файлов .....	1092
Второй этап чистки .....	1092
Установка даты и времени.....	1092
Выбор инструмента.....	1093
Работа с программой Inno Setup .....	1093

Сценарий инсталляции .....	1095
Как найти систему AutoCAD .....	1103
Сборка установочного комплекта .....	1105
Испытания установочного комплекта .....	1105
Испытания в работе .....	1107
Корректировка программ .....	1108
Новая стратегия работы с пользователями ruCAD .....	1108
Корректировка сценария установки .....	1119
Резюме .....	1126
<b>Глава 37. Подготовка к распространению .....</b>	<b>1127</b>
Особенности приложений для AutoCAD .....	1127
Составление бизнес-плана .....	1129
Цель вашего предприятия .....	1129
Кому нужна ваша программа .....	1129
Зачем нужна ваша программа .....	1130
Что вы будете продавать .....	1131
Сколько стоит ваша программа .....	1131
Юридические вопросы .....	1132
Авторские права .....	1133
Регистрация программ .....	1133
Сертификация и лицензирование .....	1133
Юридическое лицо .....	1134
Проверка собственных прав .....	1134
Лицензионные соглашения .....	1134
Подведем итоги .....	1135
<b>Глава 38. Особенности AutoCAD 2005 .....</b>	<b>1137</b>
Новинки для разработчиков .....	1137
Хорошие новости .....	1138
Плохие новости .....	1138
Что делать? .....	1141
Резюме .....	1141
<b>Послесловие .....</b>	<b>1143</b>
Сроки разработки .....	1143
Результаты разработки .....	1144
Параметры книги .....	1144
<b>Приложение. Описание компакт-диска .....</b>	<b>1145</b>
Как устанавливать .....	1145
Что будет установлено .....	1146
Что не входит в установочный комплект .....	1146
<b>Источники информации .....</b>	<b>1147</b>
Список литературы .....	1147
Ресурсы в Интернете .....	1149
<b>Предметный указатель .....</b>	<b>1152</b>



## ГЛАВА 1

# Подготовка к разработке

В *части I* нашей книги мы рассмотрим казалось бы скучные, но необходимые вопросы — общую концепцию будущей САПР. Образно говоря, мы вначале должны решить, как будем строить наш "мост" — вдоль реки или поперек. Очень часто эти важные вопросы оставляются "на потом" ("будем строить вверх, куда надо будет — туда и положим") или вообще не задаются.

Бывали и другие крайности (особенно в "советские" времена) — дальше разработки концепций, технических заданий и проектов дело не двигалось. На работу с ненужными бумагами зачастую тратились все силы и средства, а до реализации дело и не доходило.

Теперь другое время, бесполезную работу никто не делает (надо верить в лучшее, надо), но пренебрегать полезным опытом прошлого не следует. Хотя *стандарты* Единой системы программной документации (ЕСПД), введенные в СССР в начале 80-х годов прошлого века, безнадежно устарели, из них можно почерпнуть много полезного. Разумеется, с учетом нынешнего состояния технологии. Приведем перечень некоторых стандартов.

- ГОСТ 19.001-77 "Общие положения".
- ГОСТ 19.101-77 "Виды программ и программных документов".
- ГОСТ 19.102-77 "Стадии разработки".
- ГОСТ 19.104-78 "Основные надписи".
- ГОСТ 19.105-78 "Общие требования к программным документам".
- ГОСТ 19.106-78 "Требования к программным документам, выполненным печатным способом".
- ГОСТ 19.201-78 "Техническое задание, требования к содержанию и оформлению".
- ГОСТ 19.202-78 "Спецификация, требования к содержанию и оформлению".
- ГОСТ 19.401-78 "Текст программы. Требования к содержанию и оформлению".
- ГОСТ 19.402-78 "Описание программы".
- ГОСТ 19.501-78 "Формуляр. Требования к содержанию и оформлению".
- ГОСТ 19.502-78 "Общее описание. Требования к содержанию и оформлению".

- ГОСТ 24.201-79 "Требования к содержанию документа "Техническое задание".
- ГОСТ 24.202-80 "Требования к содержанию документа "Технико-экономическое обоснование создания АСУ".
- ГОСТ 24.204-80 "Требования к содержанию документа "Описание постановки задачи".
- ГОСТ 24.207-80 "Требования к содержанию документов по программному обеспечению".
- ГОСТ 34.201-89 "Виды, комплектность и обозначение документов при создании автоматизированных систем".
- ГОСТ 34.601-90 "Автоматизированные системы. Стадии создания".
- ГОСТ 34.602-89 "Техническое задание на создание автоматизированной системы".
- ГОСТ 34.603-92 "Виды испытаний автоматизированных систем".

Тексты этих документов, а также других нормативных правовых актов в области информатизации можно найти на интересном сайте [www.internet-law.ru](http://www.internet-law.ru).

## Требуется ли техническое задание

Важнейшим документом, без которого нельзя начинать работу, является *техническое задание*. Даже если вы разрабатываете программные средства по собственной инициативе, "для себя" или "ради интереса" (зарубежные читатели причин этого просто не поймут) — обязательно подготовьте техническое задание. Хотя бы для себя. Даже если вы в одном лице и заказчик, и разработчик.

Техническое задание содержит совокупность требований к программному средству и может использоваться как критерий проверки и приемки разработанной программы. Достаточно полно (и хитро) составленное (с учетом возможности внесения дополнительных разделов) и принятое заказчиком и разработчиком техническое задание очень облегчит работу над проектом.

Стандарты ЕСПД носят рекомендательный характер, но если в договоре на разработку программных средств на них будут сделаны ссылки, то для разработчиков они станут обязательными. Разработчикам при подготовке или рассмотрении договора следует быть очень внимательными, т. к. "невинная" ссылка в договоре на устаревший стандарт может потребовать много дополнительной работы (или послужить причиной для формального отклонения выполненной работы). Требования стандартов лучше придерживаться, хотя бы в структурном плане.

Раз уж мы взялись за разработку собственной системы, мы и приведем пример технического задания. Но прежде чем написать задание, необходимо продумать и задокументировать общую концепцию системы. Рассмотрением концепции системы мы и займемся в ближайших главах.

## Что нужно знать об авторских правах

Может быть не в самый первый день работы, но как можно раньше решите вопрос с *авторскими правами* на будущий программный продукт. В соответствии с законо-

дательством РФ программы для электронно-вычислительных машин и базы данных, созданием которых мы намерены заняться, относятся к "произведениям, являющимся объектами авторского права". Этот факт, о котором забывают или не знают в начале работы, может оказаться очень серьезным при переходе к распространению программного продукта.

Если вы являетесь "программистом по найму", то есть работаете "на дядю" за зарплату, то вопрос с авторскими правами или решен в контракте, или будет решаться по закону. В России действуют Закон РФ № 3523-1 от 23 сентября 1992 года "О правовой охране программ для электронно-вычислительных машин и баз данных", Закон РФ № 5351-1 от 9 июля 1993 года "Об авторском праве и смежных правах" и другие правовые акты.

В этой книге мы не будем цитировать законы. Внимательно изучите их, и вы узнаете массу полезных для себя вещей. Просто удивительно, сколько российских программистов не имеют понятия о правовых актах, непосредственно регулирующих их деятельность!

В России разработка программ, особенно относящихся к САПР, часто производится не в порядке выполнения служебного задания, а по собственной инициативе авторов. Нередко разработчик вообще не является "штатным" программистом, а разрабатывает программы в дополнение к основной проектной или конструкторской работе и в рабочее, и в нерабочее время, в том числе на собственной технике. Дополнительного вознаграждения он не получает, разве что иногда начальник похлопает по плечу и скажет, к примеру, — "Молодец, Вася!"

Неприменно постарайтесь "узаконить" свою программистскую деятельность или придать ей определенный статус. Если никто не собирается вам оплачивать разработку программ, то добейтесь того, чтобы в вашем контракте было упоминание о том, что разработка программ не входит в ваши служебные обязанности. А если все-таки входит, то только тех, которые включены в производственные планы и задания. Постарайтесь оговорить условие, чтобы с вами по каждой теме или на работу в целом заключался договор по какой-то из следующих существующих форм:

- авторский договор о передаче прав на использование произведений (программных продуктов), созданных в порядке выполнения служебного задания;
- авторский лицензионный договор на коммерческое использование программного средства;
- договор на создание и использование программы для ЭВМ или БД (базы данных) в порядке выполнения служебных обязанностей или по заданию работодателя;
- трудовой контракт с разработчиком программ для ЭВМ или БД при приеме на постоянную работу;
- трудовой контракт с разработчиком программ для ЭВМ или БД при приеме на работу, определенную целью задания.

Если такие договоры будут заключаться, то за работу вы будете получать оплату, а если не будут заключаться (что более вероятно), то у вас будут развязаны руки — и личные, и имущественные права на создаваемые программы будут принадлежать вам.

Если работа выполняется творческим коллективом, необходимо знать, что непременно возникнут и такие вопросы:

- выделение частей программного продукта, имеющих самостоятельное значение, их авторов и определение долей этих частей в коллективном произведении;
- определение вклада соавторов в часть произведения, составляющую неразрывное целое;
- рано или поздно выяснится, что если в числе соавторов имеется лицо, обладающее по отношению к другим соавторам властными полномочиями, то потребуются указать конкретный вклад такого лица в произведение;
- возможно, что для выполнения некоторых работ вам придется привлекать исполнителей со стороны. Непременен четко и недвусмысленно определите условия их участия в проекте в части возможных претензий на соавторство.

## Как собрать команду разработчиков

Наверняка вам придется работать над проектом в составе команды. Пусть это будет очень маленькая команда (2—3 человека), но работать придется совместно. В идеале это должна быть команда друзей и единомышленников.

Сразу необходимо решить вопрос о руководителе проекта, чтобы потом, подобно известным литературным героям, не задавать друг другу вопрос: "А ты кто такой?"

Не обязательно, чтобы все в команде были профессионалами высокого класса (где их взять?), но руководить проектом непременно должен профессионал-проектировщик (напоминаем, что мы рассматриваем в книге САПР и только САПР на базе AutoCAD). Нештатными членами команды обязательно должны быть несколько конечных пользователей-энтузиастов, которым можно поручать работы, не связанные с программированием (например, формирование библиотек блоков) и тестирование готовых программ. В качестве вознаграждения за дополнительную нагрузку они должны иметь повышение производительности своего труда.

Очень важно, чтобы все члены творческого коллектива, даже если они выполняют только мелкие "черные" работы, ощущали себя участниками большого общего дела, знали его состояние, результаты и проблемы. Вспомните старую притчу, в которой на вопрос к двум рабочим, несущим носилки, о том, что они делают, один отвечает: "Я таскаю камни", а другой: "Я строю Храм". Постарайтесь создать у всех участников проекта ощущение, что они именно "строят Храм".

Необходимо периодически проводить общие совещания (даже если они не нужны технологически), специально организуя их с некоторой, возможно излишней, официальностью, а иногда и с торжественностью. Постарайтесь как можно чаще и публичней отмечать достижения конкретных людей. Можете даже придумать шуточные награды, медали и грамоты. Поверьте нашему опыту, людям это нравится (даже если они ворчат: "лучше бы материально") и такие награды могут храниться много лет.

Очень важно, чтобы молодые и малоопытные члены команды во время работы над проектом ощущали реальную пользу для себя в виде повышения собственной квалификации. Руководитель проекта просто обязан этим заниматься. Лучше всего, если он будет поручать выполнение все более сложных задач. Возможно, опытный

работник решит такую задачу за несколько минут, но в долгосрочном плане важнее обеспечение роста квалификации молодежи.

Заранее договоритесь со своим коллективом о порядке использования главного вашего достояния — исходных текстов. Возможно, что кому-то потребуется использовать ваши совместные решения в своих "левых" разработках. Совместно решите, можно ли это делать, а если можно, то как. Позаботьтесь о сохранности ваших совместных архивов. Больше всего опасайтесь доступа к ним вездесущих "приятных во всех отношениях ребят", особенно тихих и скромных. Помните: никто не ведет себя тише комара, сосущего кровь.

Задумайтесь и о возможных конфликтных ситуациях.

- Как быть, если часть коллектива перейдет на другую работу? Как придется при этом "делить" исходные тексты?
- Как быть, если кто-то перейдет в "конкурирующую фирму"? А если это будет "главный идеолог" проекта?
- Как быть, если коллектив вообще перессорится?

Мы не даем рекомендаций о том, как поступать в таких случаях. Важно, чтобы вы сами были готовы к решению этих вопросов.

## Как организовать работу над проектом

Помимо согласованных в техническом задании требований к конечному продукту (что сделать) необходимо сразу же выработать и стандартные правила работы (как делать). Не обязательно, чтобы они были изложены в виде документа, хотя, разумеется, лучше закрепить их на бумаге.

Существует немало стандартов организации программирования. Например, широко известна концепция "агрессивного" и высокоэффективного процесса разработки программ Extreme Programming (сокращенно XP) — небольшой набор конкретных правил, позволяющих максимально эффективно выполнять требования современной *теории* управления программными проектами<sup>1</sup>. К сожалению, для тех, кто занимается разработками приложений к системе AutoCAD, особенно в России, эти правила мало пригодны, прежде всего потому, что даже теоретически нельзя разрабатывать такие серьезные проекты с привлечением столь несерьезных людских и финансовых ресурсов. Для *нашего* разработчика, который часто является и руководителем, и постановщиком, и программистом в одном лице (да еще одновременно и чертежником), концепция XP может показаться утопией, хотя отдельные элементы XP можно с успехом использовать.

## Хранение информации

Непременным условием является обязанность всех соразработчиков хранить всю свою информацию на сервере (в крайнем случае, при отсутствии мощной сети, должно быть ежедневное "складирование" всех работ в специальные каталоги).

---

<sup>1</sup> Материалы по Extreme Programming на русском языке можно найти на сайте [www.xprogramming.ru](http://www.xprogramming.ru), оригинальное описание — на [www.xprogramming.com](http://www.xprogramming.com).



Для каждого дня создавайте специальный архив с именем по текущей дате, например, "07\_02\_03". При этом никогда не удаляйте "ненужные" архивы. Поверьте, обязательно будут возникать ситуации, когда понадобится вернуться к предыдущей версии с минимальными временными потерями. Не забывайте об известном законе: "То, что вы храните достаточно долго, можно выбросить. Как только вы что-то выбросите, оно вам понадобится".

## Документирование работ

В общедоступном каталоге непременно должен находиться план проекта и TODO-файлы.

### Совет

Непременно записывайте все свои планы и мысли (даже не очень четкие). Проще всего это делать с помощью обычного Блокнота (notepad.exe). Не все знают, что если в первой строке файла, открываемого в notepad.exe, стоит .LOG, то "блокнотик" автоматически вставляет в текст строку со временем и датой. Ведите такие заметки в файлах с расширением .todo, "закрепите" за этим типом файлов notepad.exe. В первый раз записывайте строку .LOG вручную, а далее вам будет помогать программа.

### Листинг 1.1. Пример TODO-файла

```
.LOG
23:38 10.01.03
Начать и кончить!
23:38 10.01.03
Написать совет по использованию NOTEPAD с .LOG
```

Сразу начинайте документировать разрабатываемые программы. Ни в коем случае не оставляйте это "на потом"! В конце работы вы непременно окажетесь в цейтноте, а восстанавливать по памяти прошлогодние мысли гораздо труднее. Во многих программах не используются прекрасные возможности только потому, что разработчики о них уже забыли, а пользователи еще не знают. И могут никогда не узнать.

Лучше всего делать документацию в файле проекта справочной системы. Мы используем программный продукт Help and Manual. Эта (и другие подобные системы) позволяют вести в своем проекте структурированные справочники, из которых впоследствии можно легко сгенерировать файлы формата HLP, CHM и Manual в формате RTF. Заодно можно получить и файлы формата HTML.

Начинайте все документировать, приступая к обдумыванию концепции!

## Планирование

При разработке концепции сразу составьте предварительный *план-график разработки* системы, пока не связывая его с определенными сроками. В плане необходимо установить очередность разработки компонентов системы на принципиальном уровне (детали вы пока просто не знаете). На первом этапе должна быть запланирована разработка общего ядра системы, затем — реализация задач общего характера, и только в последнюю очередь — частные задачи.

Даже если от вас требуют в первую очередь "автоматизировать электрику", убедите заказчика<sup>1</sup>, что начинать с этого нельзя. Надо сделать так, чтобы как можно больше пользователей быстро получали эффект от вашей работы. В решении частной сложной задачи (например, полная автоматизация конструирования монолитных участков) вы можете увязнуть навсегда, и пользы для организации от этого будет немного. Если же вы быстро автоматизируете вычерчивание таблиц, общих данных, вставку блоков и других подобных работ *в общем виде*, то это сразу даст большой эффект.

Концентрируйте усилия на завершении самых важных задач, выбранных заказчиком по вашим рекомендациям, вместо того чтобы иметь несколько незаконченных задач, выбранных разработчиком.

Постоянно корректируйте и детализируйте рабочий план. В идеале он должен быть доведен до отдельных законченных задач (называемых в концепции XP *итерациями*), на решение которых отводится одна неделя. В начале разработки в итерации будут входить скрытые от заказчика внутренние задачи (например, "разработка библиотеки для работы с блоками"), а после некоторой "точки перелома" — решения для конечного пользователя (например, "вычерчивание трубопроводов в три линии"), которые будут пригодны к практическому использованию.

### Замечание

Относитесь к планированию и к контролю за соблюдением графиков достаточно серьезно, но не дайте превратиться этому процессу в самоцель. Планов и графиков у нас всегда много, как бывает с результатами — вы прекрасно знаете (читайте В. С. Черномырдина). Скорость работы программиста не всегда возможно (в отличие от скорости рытья траншей экскаватором) рассчитать, но наличие планов и проверка их выполнения лучше, чем хаотичная работа.

## Программирование

Стандартные приемы работы, связанные с непосредственной разработкой программ (кодирование, дизайн, тестирование) мы рассмотрим в *части III*.

## Как строить отношения с руководителями

Вы должны быть готовы к неизбежным неприятностям с непосредственными формальными руководителями. Большинство российских руководителей "советского" или "нового русского" образца. Им, например, может нравиться "чисто конкретная" работа программиста категории "мальчик по вызову", который сделал себя просто незаменимым, потому, что без него никто не может заменить бумагу или картридж в плоттере, а иногда и просто включить компьютер. Обычно "начальникам" очень не

---

<sup>1</sup> Определитесь сразу, кто же заказчик работы. При работе "на сторону" это тот, с кем заключен договор и кто платит деньги. А если вы работаете внутри своей организации, то тут все сложнее. Официальный заказчик всегда один, а "представителей" найдется много. Командовать вами захотят многие, требования и пожелания будут противоречивые. Сразу договоритесь, кто из "заказчиков" самый главный (например, главный инженер) и будет утверждать принципиальные решения и очередность выполнения крупных разделов, а кто рангом ниже (например, начальники отделов) — с ними надо будет планировать отдельные направления.

по душе, когда целая команда занимается неизвестно чем, пишет какие-то непонятные иероглифы, переговаривается на непонятном языке и "вообще, воображают себя умнее всех".

Постарайтесь овладеть искусством дипломатии. Не забывайте старую истину: "Начальник не всегда прав, но он всегда начальник". Не давайте ему чувствовать себя дураком (свое мнение держите при себе). Постарайтесь вовлечь его в процесс работы над проектом.

- Почаще спрашивайте его мнение по каким-нибудь мелким вопросам.
- Непременно обучите его нескольким жаргонным программистским словам и аббревиатурам, чтобы он мог шегольнуть ими в своем кругу.
- Научите его правильно произносить название вашего проекта (это необходимое, но достаточное условие его участия).
- Как можно чаще давайте пощупать (именно физически пощупать, в виде твердой копии) очередные результаты ваших трудов, например чертежей, сделанных (или якобы сделанных) исключительно с помощью вашей системы.
- Периодически предоставляйте ему письменные отчеты о проделанной работе. Лучше всего сопровождать отчет перечнем новых или обновленных файлов, который вы получаете легко автоматически, а выглядит он очень солидно, и наглядно показывает, "чего и сколько" (помните фразу Аркадия Райкина?) сделано.
- Усыпив бдительность руководителя, постарайтесь юридически оформить вашу деятельность по одной из упоминавшихся ранее форм, разумеется, в выгодной вашему коллективу формулировке. Идеально, если там будет указано что-то вроде "все права на исполняемый код принадлежат работодателю, на исходные тексты — работнику" (известный принцип заключения договоров с медведем — "вершки и корешки").
- Ни в коем случае не ввязывайтесь в открытые конфликты и "нетехнические" интриги. Любой начальник сильнее вас в этих вопросах, это "его поле".
- Не пытайтесь угрожать ему своим уходом — как правило, это бесполезно. Даже если на вас держится вся работа, даже если она развалится вообще — начальнику на это наплевать.

В случае если разрыв произойдет, непременно зафиксируйте документально, что и в каком состоянии остается после вашего ухода. Скорее всего, с вас могут долго требовать какие-нибудь книги, которые вы брали несколько лет назад, вспомнят про какую-нибудь логарифмическую линейку, числящуюся за вами с незапамятных времен, но никто не вспомнит про главное богатство — *информационные ресурсы* (официальный термин по закону). Для того чтобы потом на вас не "вешали всех собак", сделайте списки всех файлов во всех каталогах, представляющих интеллектуальную ценность, командой `dir /s` (или иным способом) и приложите копию этого списка к акту сдачи-приемки.

## Как нормировать разработку программных средств

В СССР нормирование труда было поставлено очень хорошо. Нормировалось, контролировалось и регламентировалось все. Разумеется, за исключением труда управ-

ленцев. Даже для проектных организаций существовали "Единые нормы времени и расценки" (ЕНВиР) на работы, выполняемые сдельно. Однако применялись эти нормы на практике очень редко по множеству причин, которые выходят за рамки нашей книги.

Предпринимались попытки нормирования и в области разработки программного обеспечения, например:

- Укрупненные нормы времени на разработку программных средств вычислительной техники, издание 1988 г.
- Укрупненные нормы времени на изготовление и сопровождение программных средств вычислительной техники, издание 1988 г.
- Типовые нормы времени на программирование задач для ЭВМ, издание 1989 г.
- Временная методика определения объемов программных средств вычислительной техники, издание 1989 г.
- ОСТ 4.071.030. Автоматизированная система управления предприятием. Создание системы. Нормативы трудоемкости.

Сейчас эти нормы выглядят просто курьезами. Главным их предназначением, на наш взгляд, было оправдание существования многочисленного "способствующего" персонала и руководства. На работу настоящих программистов они никакого влияния не оказывали.

Теперь совершенно другие социально-экономические и технические условия для программистов. Вообще-то все хотят видеть какие-то нормативы. Одни — чтобы убедиться, не переработали ли они, другие — чтобы знать, сколько же их работа стоит. Конечно, существуют рыночные отношения, но всегда полезно иметь представление о какой-то "стартовой цене" своего труда, с которой можно начинать "аукцион".

Несколько лет назад мы, на основании собственной статистики, составляли такие нормы. Нормировались работы, выполняемые с использованием системы AutoCAD. Была произведена статистическая обработка следующих наработанных данных:

- 2382 файла общим объемом 208 979 349 байт в банке данных геоинформационной системы г. Кургана, в том числе:
  - 1361 файл топографических планшетов и электронных карт;
  - 741 файл баз данных;
- 1365 файлов объемом 26 065 536 байт интегрированной системы InCAD;
- 2553 файла объемом 7 233 454 байта исходных текстов программ интегрированной системы InCAD.

Нормы были разработаны на такие виды работ:

- подготовка различных документов;
- оцифровка топографических планшетов;
- занесение информации в базы данных;
- определение геодезических координат по почтовым адресам;
- формирование банка данных из электронных калек (слоев, разложенных по отдельным файлам);

- обмен информацией между базами данных и AutoCAD;
- постановка задач для разработки программ;
- разработка программ на AutoLISP и других языках программирования;
- разработка программной документации;
- внедрение программ;
- приемка и освоение программных средств;
- подготовка программных средств к распространению;
- поставка программ;
- сопровождение программных средств в течение одного года;
- обучение пользователей;
- оказание технической помощи в эксплуатации программ;
- разработка меню в формате AutoCAD;
- создание типовых блоков в формате AutoCAD;
- формирование библиотеки из готовых блоков;
- создание слайдов и библиотек слайдов в формате AutoCAD и другие виды чертежных и расчетных работ с использованием AutoCAD.

Нормы времени были разработаны с учетом возможности автоматизированного учета и расчета трудозатрат и ориентированы, как правило, на машинно-определяемые единицы измерения объемов работ:

- 1 Кбайт размера файла текстового формата;
- 1 Кбайт размера файла формата документа текстового процессора Microsoft Word без включенных в файл шрифтов;
- 1 Кбайт размера файла рисунка AutoCAD без включения неиспользуемых объектов (блоков, описаний слоев, видов и т. п.);
- 1 Кбайт размера файла слайда AutoCAD;
- 1 Кбайт размера файла библиотеки слайдов AutoCAD;
- 1 Кбайт размера файла откомпилированного меню AutoCAD;
- 1 Кбайт размера файла откомпилированной LISP-программы для системы AutoCAD;
- 1 Кбайт размера файла базы данных формата DBF с учетом полей примечаний;
- 1 Кбайт суммы размеров файлов различных форматов, входящих в комплект программного средства или комплекса банка данных и т. п.

Объемы работ при использовании машинно-определяемых единиц измерения определялись по размеру файлов соответствующих типов. Конечно же, была разработана специальная программа, которая позволяла быстро определить трудоемкость любого вида работ и составить развернутую и сводную смету стоимости работ. Кроме того, на основные виды работ были выведены средние удельные показатели, относящиеся как к размеру файлов, так и к физическим объемам (лист, планшет).

## Пример норм времени на разработку программ

Для примера приведем некоторые наши нормы времени на разработку LISP-программ (табл. 1.1).

**Таблица 1.1.** Нормы времени (фрагмент) по категориям сложности, отнесенные к 1 Кбайт откомпилированного кода, чел./час

Вид работ	Категория				
	1	2	3	4	5
Постановка задачи	9,524	11,235	12,824	22,291	28,244
Программирование	4,086	4,822	5,504	9,567	12,122
Разработка документации	6,531	7,708	8,798	15,292	19,376
Внедрение	3,018	3,562	4,065	7,066	8,953

Приведенные нормы времени применяются при определении трудозатрат по отдельным программам. Категории сложности функций определяются по табл. 1.2, а нормы времени — по табл. 1.3.

**Таблица 1.2.** Категории сложности LISP-функций

Категория	Описание
1	Вывод сообщений, запуск внешних программ без обработки результатов, работа с единицами измерений
2	Параметрическое рисование по точкам, обработка меню, вставка блоков и форм, управление слоями, обработка имен файлов и каталогов, ввод данных в диалоговом режиме, геометрические функции
3	Работа с внешними файлами, обработка видовых экранов, работа с INI-файлами, преобразование примитивов (расчленение, объединение), программное изменение свойств примитивов
4	Функции сложной обработки строк, списков, графической базы данных; запуск внешних программ с обработкой результатов; резидентные библиотеки функций; экспорт-импорт ИНФО-объектов и графики; модификация ИНФО-объектов; настройки параметров системы; генерирование меню и сценариев
5	Функции рисования сложных объектов с модификацией графической базы данных, расчеты с использованием графики, сложное многовариантное рисование

Сейчас эти нормативы несколько устарели и мы не приводим их в книге полностью ради экономии драгоценного места. "Скачать" их можно с сайта [www.gis.kurgan.ru](http://www.gis.kurgan.ru). Но раз наша книга о том, *как это делается*, кратко опишем применяющуюся методику определения нормативов на разработку программных средств.

Исходные тексты программ (более 2 тыс.) независимо анализировались несколькими экспертами (в том числе и авторами). Каждый эксперт, на основании собственного опыта, устанавливал для программы категорию сложности и вероятное время