Под редакцией
П. В. Конюховского
Д. Н. Колесова

ЭКОНОМИЧЕСКАЯ ИНФОРМАТИКА

Санкт-Петербург
Москва • Харьков • Минск
2001

ПИТЕР®

ББК 32.973+65с11я7
УДК 681.3:33(075)
К65

Рецензенты:

Дуболазов В. А., доктор экономических наук, профессор, заведующий кафедрой «Предпринимательство и коммерция» Санкт-Петербургского государственного технического университета.

Львов Ю. А., доктор экономических наук, профессор, заведующий кафедрой экономической кибернетики и статистики Санкт-Петербургской государственной инженерно-экономической академии.

К65 **Экономическая информатика** / под ред. П. В. Конюховского и Д. Н. Колесова. — СПб: Питер, 2001. — 560 с.: ил.

ISBN 5-272-00219-9

Книга предназначена для студентов экономических специальностей, изучающих современные инфомационные технологии в рамках дисциплины «Экономическая информатика», преподавателей, обеспечивающих учебный процесс по данной дисциплине, и специалистов-экономистов, повышающих свою квалификацию в области применения современных информационных технологий.

В книге рассмотрены основные категории программных средств вычислительной техники, применяемых в экономических расчетах, приведены способы наиболее эффективного подхода к решению возникающих на практике задач и множество контрольных примеров и вопросов, помогающих закреплению полученных знаний.

ББК 32.973+65с11я7
УДК 681.3:33(075)

ISBN 5-272-00219-9

© П. В. Конюховский, Д. Н. Колесов, Г. С. Осипов, М. В. Михайлов, Е. В. Губина, Р. В. Гордиенко, С. В. Евстратчик, Е. С. Володина, 2000
© Серия, оформление, Издательство «Питер», 2001

Краткое содержание

Предисловие 13
Глава 1. Экономическая информация и ее обработка 14
Глава 2. История и направления развития вычислительной техники 32
Глава 3. Технические средства обработки информации 50
Глава 4. Системное и сервисное программное обеспечение 87
Глава 5. Обработка текстовой информации 149
Глава 6. Процессоры электронных таблиц 210
Глава 7. Системы управления базами данных 260
Глава 8. Электронные презентации 312
Глава 9. Программное обеспечение управления проектами 349
Глава 10. Локальные вычислительные сети 394
Глава 11. Глобальные компьютерные сети 436
Глава 12. Информационные базы данных и электронные библиотеки 476
Глава 13. Системы искусственного интеллекта и нейронные сети 509
Заключение 544
Алфавитный указатель 547

Содержание

Предисловие 13

Глава 1. Экономическая информация и ее обработка 14

1.1. Основные категории и понятия информатики 14
1.1.1. Информация: структура, форма, измерение 15
1.1.2. Информатика и информационные технологии 16
1.2. Информационные аспекты управления 17
1.3. Управление и информация в экономике 18
1.3.1. Виды экономической информации 19
1.3.2. Характеристики экономической информации 23
1.4. Место процессов обработки информации в управлении 23
1.5. Задачи экономической информатики на современном этапе 29
Основные выводы 30
Ключевые понятия 30
Контрольные вопросы 31
Литература 31

Глава 2. История и направления развития вычислительной техники 32

2.1. Эволюция средств вычислительной техники 32
2.2. Поколения современных компьютеров 37
2.2.1. Первое поколение компьютеров 38
2.2.2. Второе поколение компьютеров 41
2.2.3. Третье поколение компьютеров 43
2.2.4. Четвертое поколение компьютеров (с 1971 года и по настоящее время) 45
Ключевые понятия 48
Контрольные вопросы 49
Литература 49

Глава 3. Технические средства обработки информации 50

3.1. Понятие вычислительной машины и принципы организации ее работы ... 50
3.2. Классы вычислительных машин и их основные характеристики 51

3.3. Состав аппаратного обеспечения персонального компьютера, характеристика основных периферийных устройств ... 53
3.3.1. Системный блок ... 53
3.3.2. Устройства хранения информации ... 66
3.3.3. Устройства ввода информации ... 71
3.3.4. Устройства вывода информации. ... 73
3.3.5. Периферийные устройства ... 83
Ключевые понятия ... 86
Контрольные вопросы ... 86
Литература ... 86

Глава 4. Системное и сервисное программное обеспечение ... 87

4.1. Операционная система ... 87
4.1.1. Понятие, основные функции и составные части операционной системы ... 87
4.1.2. Классификация операционных систем ... 89
4.2. Файловые системы ... 90
4.2.1. Основные функции файловой системы ... 90
4.2.2. Файлы и каталоги ... 91
4.2.3. Физическая организация данных на носителе ... 93
4.2.4. Права доступа к файлу ... 93
4.2.5. Другие функции файловых систем ... 94
4.2.6. Файловые системы FAT16, FAT32 ... 95
4.3. Операционная система MS DOS ... 99
4.3.1. Основные составные части MS DOS ... 99
4.3.2. Командный процессор Command.com ... 101
4.3.3. Создание командных файлов ... 105
4.3.4. Командный файл автонастройки AUTOEXEC.BAT ... 106
4.3.5. Файл конфигурации CONFIG.SYS ... 107
4.3.6. Программные оболочки ... 109
4.4. Операционные системы Windows 9x ... 110
4.4.1. Общая характеристика и история развития ... 110
4.4.2. Windows 3.1 ... 111
4.4.3. Windows 95 ... 113
4.4.4. Windows 98 ... 114
4.4.5. Особенности операционной системы Windows 2000 ... 116
4.4.6. Основные элементы графической оболочки Windows ... 118
4.5. Сервисные программные средства ... 125
4.5.1. Служебные программы ... 126
4.5.2. Архивация данных ... 132
4.5.3. Антивирусные программные средства ... 136
4.6. Краткий обзор современных операционных систем ... 142
Ключевые понятия ... 147
Контрольные вопросы ... 147

Литература ... 148
Источники в Интернете ... 148

Глава 5. Обработка текстовой информации ... 149

5.1. Задачи обработки текстовой информации ... 149
5.1.1. Ввод текста ... 149
5.1.2. Редактирование ... 155
5.1.3. Сохранение документа ... 155
5.1.4. Публикация ... 158
5.1.5. Поиск и открытие созданного документа ... 163
5.1.6. Перевод ... 164
5.2. Обработка документов средствами текстовых процессоров ... 171
5.2.1. Основные понятия и характеристики текстовых процессоров ... 171
5.2.2. Microsoft Word 2000 и его новые функциональные возможности ... 173
5.3. Работа с текстовыми документами в среде MS Word ... 179
5.3.1. Экранный интерфейс и настройки процессора MS Word ... 179
5.3.2. Создание документа ... 185
5.3.3. Ввод текста и редактирование ... 188
5.3.4. Сохранение документа ... 194
5.3.5. Управление настройками MS Word ... 195
5.4. Решение задач оформления экономической документации средствами MS Word ... 196
5.4.1. Ввод формул ... 196
5.4.2. Расположение и автонумерация формул в тексте ... 197
5.4.3. Оформление финансово-экономической документации ... 201
Ключевые понятия ... 208
Контрольные вопросы ... 209
Литература ... 209

Глава 6. Процессоры электронных таблиц ... 210

6.1. MS Excel: общая характеристика и функциональные возможности ... 210
6.1.1. Понятие электронной таблицы ... 210
6.1.2. Общая характеристика интерфейса MS Excel ... 212
6.1.3. Технология ввода данных в MS Excel ... 215
6.1.4. Формулы, функции, мастер функций ... 218
6.1.5. Графические возможности Excel ... 221
6.2. Экономико-математические приложения MS Excel ... 224
6.2.1. Средства структуризации и первичной обработки данных ... 224
6.2.2. Статистическая обработка данных и прогнозирование ... 227
6.2.3. Финансово-экономические расчеты ... 232
6.2.4. Финансовые функции для расчетов по кредитам, займам и оценкам инвестиций ... 234
6.2.5. Финансовые функции для расчета амортизации ... 240
6.2.6. Финансовые функции для расчетов по ценным бумагам ... 243
6.2.7. Решение уравнений и оптимизационных задач ... 247

Ключевые понятия 257
Контрольные вопросы 258
Литература 258

Глава 7. Системы управления базами данных 260

7.1. Основные понятия теории баз данных 260
7.1.1. Понятие базы данных 260
7.1.2. Модели организации данных 261
7.1.3. Реляционная модель данных 263
7.1.4. Язык SQL 265
7.1.5. Программные системы управления базами данных 267
7.1.6. Применение СУБД в экономике 270
7.2. СУБД MS Access и ее основные возможности 271
7.2.1. Общая характеристика СУБД MS Access 271
7.2.2. Основные этапы разработки базы данных в среде MS Access 273
7.2.3. Экономические приложения СУБД MS Access 273
7.2.4. Создание таблиц и схем данных 275
7.2.5. Создание схемы данных 279
7.2.6. Разработка запросов к базе данных 280
7.2.7. Конструирование экранных форм для работы с данными 285
7.2.8. Конструирование отчетов 291
7.2.9. Средства макропрограммирования в MS Access 294
7.2.10. Разработка программных приложений для MS Access 296
7.3. Организация взаимодействия между системами управления данными 298
7.3.1. Проблема форматнонезависимого доступа к данным и технология ODBC 298
7.3.2. Доступ из MS Access к источникам данных в формате других программных приложений 299
7.3.3. Технологические решения по организации доступа к данным 301
7.4. Организация многопользовательского доступа к данным 302
7.4.1. Проблема многопользовательского доступа и параллельной обработки данных в автоматизированных информационных системах 302
7.4.2. Основные направления развития технологии клиент-сервер 306
7.4.3. Организация защиты данных в СУБД MS Access 307
Ключевые понятия 310
Контрольные вопросы 310
Литература 311

Глава 8. Электронные презентации 312

8.1. Современные способы организации презентаций 312
8.2. Microsoft PowerPoint 2000 и его новые возможности 317
8.3. Организация презентаций средствами MS PowerPoint 320
8.3.1. Экранный интерфейс и настройки PowerPoint 320
8.3.2. Создание новой презентации и оперирование структурой 326
8.3.3. Оформление презентации 331

8.3.4. Показ презентации ... 335
8.3.5. Публикация презентации ... 337
8.4. Презентационное оформление экономической документации средствами MS PowerPoint ... 339
Ключевые понятия ... 347
Контрольные вопросы ... 347
Литература ... 348

Глава 9. Программное обеспечение управления проектами 349

9.1. Понятие о проектах и основных методах управления ими ... 349
9.1.1. Программные средства автоматизации процессов делопроизводства и бизнес-планирования ... 349
9.1.2. Общие проблемы теории управления проектами ... 351
9.1.3. Программное обеспечение MS Project ... 354
9.1.4. Пример задачи управления проектом ... 356
9.2. Создание описания и графика проекта в MS Project ... 357
9.2.1. Ввод данных по проекту ... 357
9.2.2. Формирование и представление графика работ ... 366
9.2.3. Дополнительные возможности ПО MS Project ... 372
9.3. Управление циклом реализации и ресурсами проекта ... 376
9.3.1. Критический путь и методы управления им ... 376
9.3.2. Базовый план и контроль его выполнения ... 381
9.3.3. Управление ресурсами проекта ... 383
9.4. Финансовое управление проектом ... 386
9.4.1. Технология управления финансами проекта в MS Project ... 386
9.4.2. Формирование информационной базы для финансовых расчетов 387
9.4.3. Основные методы решения задач финансового управления проектом ... 390
Ключевые понятия ... 392
Контрольные вопросы ... 392
Литература ... 393

Глава 10. Локальные вычислительные сети ... 394

10.1. Принципы построения вычислительных сетей ... 394
10.1.1. Программные и аппаратные компоненты вычислительной сети ... 394
10.1.2. Основные требования, предъявляемые к современным вычислительным сетям ... 397
10.1.3. Классификации вычислительных сетей ... 400
10.1.4. Эталонная модель OSI ... 403
10.1.5. Методы передачи данных на физическом и канальном уровнях ... 408
10.1.6. Методы коммутации ... 408
10.1.7. Линии связи ... 409
10.2. Базовые технологии локальных сетей ... 411
10.2.1. Технология Ethernet ... 411
10.2.2. Технологии Fast Ethernet и 100VG-AnyLAN ... 413

10.2.3. Технология Gigabit Ethernet 415
10.2.4. Технология Token Ring 415
10.2.5. Технология FDDI 417
10.3. Построение локальных сетей 418
10.3.1. Структуризация локальных сетей средствами канального уровня 418
10.3.2. Построение локальных сетей средствами сетевого уровня 419
10.4. Стандартные стеки коммуникационных протоколов 420
10.4.1. Стек протоколов TCP/IP 420
10.4.2. Стек протоколов IPX/SPX 421
10.5. Системное программное обеспечение вычислительных сетей 422
10.5.1. Windows NT 424
10.5.2. Novell NetWare 427
10.5.3. Сетевая рабочая станция в локальных сетях 430
Ключевые понятия 434
Контрольные вопросы 435
Литература 435

Глава 11. Глобальные компьютерные сети 436

11.1. Основные принципы построения сети Интернет 436
11.1.1. История создания сети Интернет 437
11.1.2. Роль документации в развитии Интернета 440
11.1.3. Организационная структура Интернета 440
11.1.4. Современная структура сети Интернет 441
11.2. Основные протоколы сети Интернет 442
11.2.1. Адресация в сети Интернет 442
11.2.2. Протокол IP 443
11.2.3. Протокол TCP/IP 443
11.2.4. Порты и соединения 444
11.2.5. Система доменных имен DNS 444
11.3. Электронная почта 445
11.3.1. Адрес электронной почты 445
11.3.2. Формат сообщения электронной почты 445
11.4. Система World Wide Web 451
11.4.1. Принципы работы сервера и клиента 453
11.4.2. Универсальный адрес ресурса 456
11.4.3. Язык разметки гипертекста (HTML) 457
11.4.4. Протокол передачи гипертекста (HTTP) 458
11.4.5. Поиск информации в Интернете 458
11.5. Перспективные технологии на основе Интернета 463
11.5.1. IP-телефония 463
11.6. Электронная коммерция 464
11.6.1. Общие сведения об электронной коммерции 464
11.6.2. Электронный рынок и роль информации в электронном бизнесе ... 466
11.6.3. Системы электронных платежей 467
11.6.4. Электронные платежные средства 468

11.7. Защита информации в Интернете ... 469
11.7.1. Принципы защиты информации ... 469
11.7.2. Криптография ... 470
11.7.3. Электронная подпись ... 471
11.7.4. Аутентификация ... 472
11.7.5. Защита сетей ... 473
Ключевые понятия ... 474
Контрольные вопросы ... 474
Литература ... 475

Глава 12. Информационные базы данных и электронные библиотеки ... 476

12.1. Общие принципы построения информационно-поисковых систем ... 476
12.1.1. Модели организации хранения и поиска документов ... 476
12.1.2. Основные принципы информационного поиска ... 477
12.1.3. Предметное индексирование и механизм поиска ... 478
12.1.4. Стратегии поиска ... 480
12.1.5. Интерфейс системы ... 481
12.1.6. ИПС глобальных сетей ... 482
12.2. Справочно-правовые системы ... 484
12.2.1. Понятие, назначение и преимущества ... 484
12.2.2. Обзор рынка СПС в России ... 485
12.2.3. Основные свойства информационных банков СПС ... 486
12.2.4. Основные возможности программных технологий СПС ... 487
12.2.5. Справочная правовая система «Консультант» ... 488
12.2.6. Справочно-правовая система «Гарант» ... 490
12.3. Интегрированная библиотечно-информационная система «Ирбис» ... 493
12.3.1. Описание системы ... 493
12.3.2. АРМ Читатель, общая характеристика ... 494
12.3.3. Поиск ... 494
12.3.4. Просмотр ... 497
12.4. Глобальные информационные базы данных ... 498
12.4.1. История развития рынка информационных услуг ... 498
12.4.2. Рынок информационных продуктов и услуг ... 499
12.4.3. Структура рынка информационных продуктов и услуг ... 500
Ключевые понятия ... 507
Контрольные вопросы ... 507
Литература ... 508

Глава 13. Системы искусственного интеллекта и нейронные сети ... 509

13.1. Искусственный интеллект, основные понятия ... 509
13.2. Знания и модели их представления ... 511
13.2.1. Логические модели ... 514
13.2.2. Продукционные модели ... 515

13.2.3. Семантические сети ... 518
13.2.4. Фреймовые модели ... 519
13.3. Экспертные системы, основные понятия и определения ... 522
13.4. Нейрокомпьютер и основы нейроинформатики ... 524
13.5. Нейрон, нейронные сети, основные понятия ... 526
13.6. Модели нейронных сетей ... 530
13.6.1. Многослойные однонаправленные сети ... 530
13.6.2. Полносвязные сети Хопфилда ... 532
13.6.3. Двунаправленная ассоциативная память ... 535
13.6.4. Самоорганизующиеся сети Кохонена ... 537
13.7. Области применения нейроинформатики ... 539
Ключевые понятия ... 542
Контрольные вопросы ... 542
Литература ... 543

Заключение ... 544

Алфавитный указатель ... 547

Предисловие

Настоящее издание задумано как учебное пособие по курсу **«Экономическая информатика»**, читаемого студентам экономических и экономико-математических специальностей вузов. При его написании в качестве базовой подразумевалась специальность **061800 – «Математические методы в экономике»**, где данный курс читается в достаточно большом объеме. Одновременно при разработке книги были учтены возможности ее использования для проведения занятий со студентами других специальностей.

Авторы ясно осознают, что для учебников по информационным курсам традиционными являются проблемы структуризации и выбора последовательности изложения материала. Последнее прежде всего обусловливается наличием сложных множественных смысловых и терминологических связей между различными разделами информатики. Такое положение является объективным и естественным с точки зрения процесса их развития, которое идет параллельно по многим направлениям. Однако при размещении материала в книге неизбежно приходится останавливаться на том или ином порядке следования глав. В данном случае авторы решили отдать приоритет хронологическому принципу, а также логике изложения «от простого к сложному».

Среди основных отличий настоящей книги от изданий аналогичной тематической направленности следует отметить то, что при рассмотрении как системного, так и прикладного программного обеспечения (например, MS Word, MS Excel и т. п.) делается акцент на его приложения в сфере экономики и финансов.

Авторский коллектив издания составили преподаватели кафедры экономической кибернетики Санкт-Петербургского государственного университета: канд. экон. наук, доц. Д. Н. Колесов (глава 1), канд. экон. наук П. В. Конюховский (главы 7, 9), д-р техн. наук, проф. Г. С. Осипов (глава 13), канд. экон. наук М. В. Михайлов (главы 3, 10, раздел 4.5), Е. С. Володина (глава 12, раздел 4.2), Р. В. Гордиенко (главы 2, 11, раздел 4.6), Е. В. Губина (главы 5, 8), С. В. Евстратчик (глава 6, разделы 4.1, 4.3, 4.4). Отдельную благодарность хочется выразить И. А Шпаковской, оказавшей авторам активную помощь в написании заключения.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу электронной почты comp@piter-press.ru (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

Подробную информацию о наших книгах вы найдете на Web-сайте издательства http://www.piter-press.ru.

Глава 1. Экономическая информация и ее обработка

1.1. Основные категории и понятия информатики

Изучение любой дисциплины так или иначе начинается с формулировки определений ее фундаментальных терминов и категорий. Особенностью термина «информация» является то, что, с одной стороны, он является интуитивно понятным практически для всех, а с другой — общепризнанной его трактовки в научной литературе не существует. Одновременно следует особо отметить, что как научная категория «информация» составляет предмет изучения для самых различных областей знания: философии, информатики, теории систем, кибернетики и т. д.

Как известно, термин «информация» происходит от латинского слова «informatio», изначально означающего изложение или разъяснение. В качестве примера приведем следующее его определение.

Информация — это совокупность сигналов, воспринимаемых нашим сознанием, которые отражают те или иные свойства объектов и явлений окружающей нас действительности. Природа данных сигналов подразумевает наличие принципиальных возможностей по их сохранению, передаче и трансформации (обработке).

Очевидно, что данное определение носит исключительно «пояснительный» характер и ни в коей мере не претендует на научную строгость, так как базируется на достаточно нечетких и расплывчатых категориях: «сигналы», «объекты», «действительность», «восприятие» и т. п.

Другой подход к определению информации отталкивается от схематичного представления процесса ее передачи. На предельном уровне абстрагирования в нем можно выделить два фундаментальных элемента:

- источник (передатчик);
- приемник (потребитель, клиент).

При их взаимодействии, собственно говоря, и возникает информация — некоторое сообщение, которое тем или иным способом уменьшает незнание потребителя (приемника) о некотором объекте, факте или явлении. Основываясь на данном подходе, один из основоположников теории информации Клод Шеннон определил информацию как *снятую неопределенность*.

В одном терминологическом ряду с информацией стоят понятия «данные» и «знания».

Под *данными* обычно понимают информацию, представленную в конкретных формах, которые адекватны возможным (ожидаемым) процессам ее обработки.

Знания — это информация, на основании которой путем логических рассуждений могут быть получены определенные выводы[1]. Сразу оговоримся, что приведенная формулировка является крайне упрощенной, а более подробно соответствующие проблемы будут рассмотрены в главе 13.

Также под знаниями иногда понимают данные, имеющие сложную организацию, обладающие как фактографической, так и семантической составляющими. Под *фактографической* составляющей понимается информация, связанная с регистрацией событий и явлений, а под *семантической* — информация, касающаяся содержательного (смыслового) толкования зарегистрированных фактов.

1.1.1. Информация: структура, форма, измерение

Важнейшими характеристиками информации являются ее структура и форма. *Структура информации* — это то, что определяет взаимосвязи между ее составными элементами. Фундаментальным свойством информации является свойство системности. Как известно, *системой* называют такую совокупность, которая обладает такими свойствами, которыми не обладает ни один из входящих в нее элементов в отдельности. Без труда можно привести массу примеров, демонстрирующих, что соединение разрозненных информационных сигналов порождает систему, обладающую качественно более высокой содержательной ценностью.

Другой стороной информации является *форма ее представления*. Среди основных форм могут быть названы:

- символьно-текстовая (информация, представленная совокупностью букв, цифр, знаков и т. п.);
- графическая (изображение и т. п.);
- звуковая.

Практическое значение научных подходов к определению термина «информация» связано с решением проблемы *измерения информации*. В соответствии с положениями общей теории информации в качестве эталона меры для нее выбирается некоторый абстрактный объект, который может находиться в одном из двух состояний (например, включен/выключен, да/нет, 0/1 и т. п.), или, как еще говорят, бинарный объект. Говорят, что такой объект содержит информацию в 1 бит. Безусловно, данный метод измерения информации во многом был предопределен возможностями ее хранения в различных технических устройствах, где на элементарном уровне информация запоминается с помощью магнитно-электрических устройств, которые могут находиться либо в намагниченном, либо в размагничен-

[1] Представляется также небезынтересным привести определение соответствующей философской категории, а именно: *«Знания — это проверенный практикой результат познания действительности, ее отражение в сознании человека»*.

ном состояниях. От бита как наименьшей меры количества информации происходят производные единицы:

1 байт = 8 бит,

1 килобайт (Кбайт) = 1024 байт = 2^{10} байт,

1 мегабайт (Мбайт) = 1024 килобайт = 2^{20} байт,

1 гигабайт (Гбайт) = 1024 мегабайт = 2^{30} байт и т. д.

Как можно заметить, приставки кило-, мега- и т. д. здесь употребляются в отличающемся от традиционного смысле, означая умножение на $1024 = 2^{10}$, а не на тысячу. Данное решение позволяет гармонично связать методы измерения информации с бинарной (двоичной) организацией системы ее хранения.

1.1.2. Информатика и информационные технологии

Рассуждения о качественном изменении той роли, которую информация стала играть как в жизни общества в целом, так и в жизни отдельных личностей, можно встретить на страницах научной, публицистической и даже художественной литературы. Постараемся в сжатом виде обобщить, в чем наиболее ярко выразились эти изменения:

- во-первых, на протяжении последних десятилетий наблюдается неизменное ускорение темпов роста объема информации. Кратко и емко данная ситуация была сформулирована в виде тезиса: «Информация — это единственный неубывающий ресурс общества». Следствием такого положения дел стал количественный барьер в процессах обработки информации: грубо говоря, иногда информацию нет смысла собирать и хранить, так как все равно отсутствуют возможности ее обработки и рационального использования;
- во-вторых, возрос удельный вес коммуникативных проблем (искажение и потери информации в процессе передачи);
- в-третьих, серьезные трудности порождает наличие межязыковых, географических, административных, терминологических и других барьеров;
- в-четвертых, практическое применение информацию зачастую становится невозможным из-за ее хаотического рассредоточения по различным источникам (так называемое рассеяние информации).

Работа по решению перечисленных проблем в конечном счете привела к выделению самостоятельной научной дисциплины — *информатики*, предметом которой стали свойства информации, ее поведение в техногенных, социальных и биологических системах, а также методы и технологии, ориентированные на сбор, обработку, хранение, передачу и распространение информации, или, кратко, *информационные технологии* (ИТ).

Данная формулировка предмета информатики не является единственной. Достаточно популярным является определение, согласно которому, **«Информатика — это наука об описании, представлении, интерпретации, формализации и применении знаний, накопленных с помощью вычислительной техники, с целью получения новых знаний»**. Дополнительно укажем на то, что само применение

термина «информатика» для обозначения соответствующей области научного знания не является общепризнанным. В частности, в США для названия данной сферы гораздо чаще используется понятие «*computer science*» (или реже «*computing*»).

Особенностью ИТ по сравнению с традиционными технологиями является качественно иная сфера их приложения. Прежде всего это связано с их универсальным характером. Обратной стороной такого «универсализма» являются сложности, возникающие при создании формальных описаний информационных технологий. Подобные попытки далеко не всегда завершались успехом. Традиционно выделяется несколько уровней представления ИТ: концептуальное, на уровне информационных потоков, на уровне инструментальных средств и некоторые другие.

Общая схема цикла информационного процесса, который должен охватываться соответствующими технологиями, приведена на рис. 1.1.

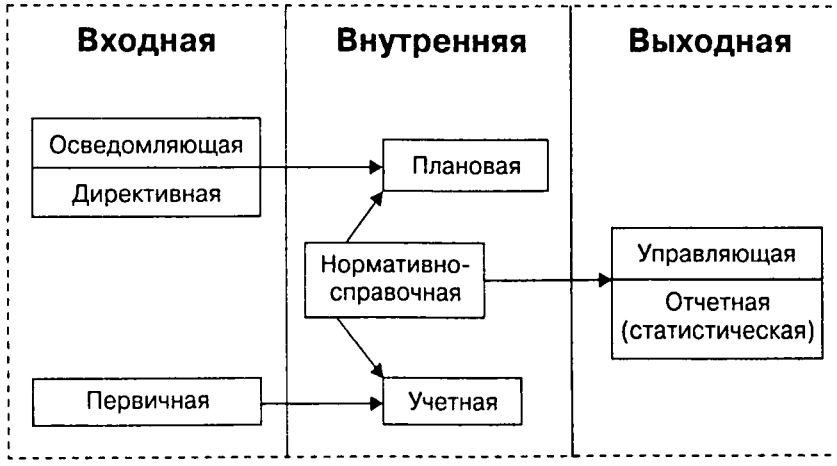


Рис. 1.1. Общая структурная схема цикла информационного процесса

Результатом развития ИТ стали процессы глобальной *информатизации общества*. Последнее означает, что все большая часть его членов оказывается втянутой в профессиональную деятельности в рамках информационных отраслей экономики, удельный вес которых неуклонно возрастает. За последние десять-пятнадцать лет произошли существенные сдвиги в структуре рынка программного и аппаратного обеспечения для вычислительной техники. Он в значительной мере из рынка продуктов и услуг превращается в рынок технологий.

1.2. Информационные аспекты управления

Информация в качестве самостоятельного объекта исследования стала рассматриваться относительно недавно. Важным толчком к таким исследованиям явилось возникновение в конце 40-х годов XX века науки об управлении в сложных динамических системах – *кибернетики*. Основоположником ее явился американский математик Норберт Винер.

Основополагающей предпосылкой, определившей выделение кибернетики в отдельное направление научных исследований, явился тезис об общности процессов управления в объектах разной природы, которая проявляется в механизме действия обратной связи и информационной основе управления. Она обусловливает принципиальную возможность моделирования экономических и социальных процессов, использования ЭВМ для управления экономическими объектами. Кибернетика утверждает существование информационного поля в объективной реальности, формулирует *закон необходимого разнообразия* для регулятора как органа управления. Главным его следствием является вывод о том, что степень сложности системы управления должна соответствовать степени сложности управляемого объекта. В сфере экономики он подтверждает объективную необходимость усложнения управления, увеличения его «мощности» по переработке информации и принятию решений.

Принципиальным моментом для общекибернетических теорий стало определение термина «управление». Приведем следующую его формулировку.

Управление (англ. — *control*) — это функция организованных систем различной природы (технических, биологических или социальных), направленная на реализацию их целевых установок и поддержание внутренне присущей им структуры.

Статистическая теория информации, являясь одним из разделов кибернетики, связывает понятие информации с уменьшением неопределенности состояния (энтропии) объекта. К. Шеннон и Н. Винер предложили математический аппарат для количественного измерения неопределенности и информации. Предложенная мера оказалась плодотворной для технических приложений — оптимизации кодирования, передачи, хранения данных и некоторых других.

Этот подход имел также методологическое значение, он способствовал пониманию того, что нет абсолютной информации об объекте, определение информации зависит от составленной нами модели объекта. Поскольку для разных целей исследования составляются разные модели с различным описанием своих состояний, то и определение информации об объекте зависит от тех целей и задач, которые стоят перед исследователем. Так, в одних и тех же данных содержится разное количество информации для разных задач управления.

В то же время необходимо отметить, что в ходе развития кибернетики помимо достигнутых успехов очень быстро выявились проблемы, носившие принципиальный характер: слишком общие подходы к изучаемым процессам не давали значимых конструктивных результатов. Поэтому в ней достаточно быстро стали выделяться направления, ориентированные на более конкретные классы исследуемых объектов, — техническая кибернетика, экономическая кибернетика и т. д.

1.3. Управление и информация в экономике

Статистическая теория информации и аналогичные ей теории не получили распространения в обработке информации для управления экономическими объектами. Это связано с тем, что такой подход не учитывает специфику экономичес-

ких данных, характер их использования в управлении. Не учитываются смысловые взаимосвязи, степень необходимости для управления, формы представления, уровень и частота использования, достоверность и другие важные характеристики.

Информация, используемая для принятии управленческих решений на всех уровнях, весьма разнообразна по своему содержанию. На уровне фирм (предприятий) к ней относятся:

- технические характеристики средств производства (в первую очередь, оборудования);
- описание технологии и условий производства;
- рыночная конъюнктура (цены, объемы спроса);
- сведения об оборотных средствах;
- сведения о кадровом составе;
- сведения о наличии и потребностях в ресурсах;
- нормативы, плановые задания;
- совокупность расчетных показателей (таких, как фондоемкость, рентабельность, себестоимость);
- различные приказы, инструкции, методики.

Перечисление и конкретизацию можно было бы продолжить, но у всех этих сведений есть общий признак — они относятся к производственно-хозяйственной деятельности людей. По этому признаку их определяют как **экономическую информацию**.

Можно согласиться с определением экономической информации, сформулированным В. Д. Бройдо. Согласно ему, «*Экономическая информация* — это та информация, которая возникает при подготовке и в процессе производственно-хозяйственной деятельности и используется для управления этой деятельностью»[1].

1.3.1. Виды экономической информации

Содержательная классификация циркулирующей в разных объектах информации зависит от отраслевой принадлежности и уровня управления. Тем не менее, в процессе обработки информация проходит аналогичные стадии, общие в управлении разными экономическими объектами. На рис. 1.2 предложена упрощенная схема обработки, в которой отражаются назначение информации и источники ее возникновения. Указанные виды информации различаются:

- формой представления;
- организацией хранения;
- характером обработки.

[1] Бройдо В. Д. Достоверность экономической информации в АСУ. Л.: Изд-во Ленингр. ун-та, 1984.

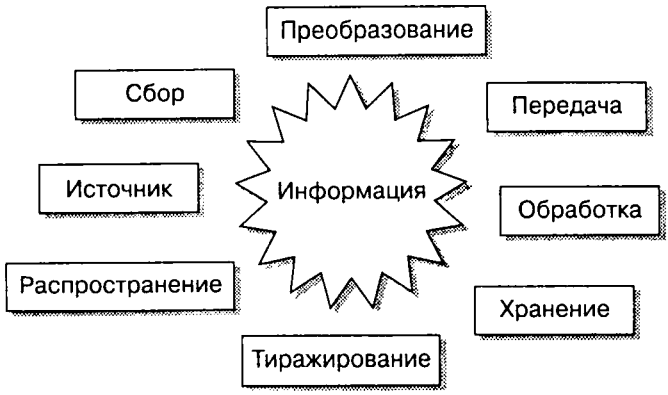


Рис. 1.2. Классификация экономической информации по характеру ее использования в управлении

Безусловно, схема, представленная на рис. 1.2, носит предельно обобщенный характер, однако она охватывает практически всю конкретно-числовую информацию, выделяемую из системы управления в процессе ее функционирования.

Входная информация поступает в орган управления извне. Часть ее, *первичная информация*, поступает от объекта управления и получается в результате непосредственного измерения или подсчета. Допустим, для фирм, занятых в сфере промышленного производства, это объем выпущенной продукции, количество брака, численность рабочих, время простоев, запасы на складе и т. д., для банков и финансовых компаний — ставка рефинансирования, объем активов и пассивов, кредитов и депозитов, действующие нормы выплат по ним и т. д. Первичная информация теснее всего соприкасается с конкретной стороной деятельности управляемых экономических объектов, причем включает как медленно изменяющиеся (условно постоянные), так и оперативные данные.

Особенностью первичной информации является то, что она очень часто содержит объемные характеристики работы объекта. На основании этих объемных характеристик с использованием нормативно-справочных данных в дальнейшем получается подробная оценка фактического состояния и динамики управляемого объекта. С точки зрения процесса управления, первичная информация выполняет роль обратной связи, поступающей от управляемого объекта и показывающей достигнутые результаты, а также отклонения от нормативного функционирования объекта. Остальная часть входной информации — ее иногда называют *внешней* и условно делят на осведомляющую и директивную — поступает из сторонних по отношению к управляющей системе органов. *Директивная информация* исходит из вышестоящих органов и в зависимости от характера подчиненности может включать параметры и условия формирования обязательных (налоговых) выплат, плановые задания и их корректировки, выделенные лимиты. *Осведомляющая информация* поступает от вышестоящих органов, а также от фирм, предприятий и организаций, связанных с объектом управления. Это поставщики, подрядчики, транспортные организации, финансовые институты (банки, пенсионные фонды, страховые компании), территориальные органы государственной власти. Если

директивные данные непосредственно влияют на цели функционирования объекта, то осведомляющие определяют условия его работы.

Директивные и осведомляющие данные, в отличие от первичных, преимущественно имеют стоимостной или безразмерный характер.

По частоте изменения и форме использования выделяют *нормативно-справочную информацию*. Это экономические, технологические, материальные и трудовые нормативы, расчетные коэффициенты, каталоги, номенклатурные справочники. Нормативно-справочная информация отличается большей стабильностью и, как правило, проще организуется в однородные массивы. Нормативно-справочная информация так же, как и входная, является исходным материалом для последующей обработки и поступает в орган управления извне. По этому признаку ее можно было бы отнести к входной информации, но в отличие от последней нормативно-справочная информация используется многократно и после первоначального ввода длительное время хранится в системе управления. Поэтому мы считаем необходимым включить ее в состав внутренней информации, то есть информации, хранимой и обрабатываемой внутри системы управления. Наиболее характерными операциями ее обработки являются выборка из соответствующих массивов и обновление. При создании информационных систем специфика хранения, обновления и использования нормативно-справочной информации обусловливает ее выделение вместе со средствами поиска и обновления в отдельную подсистему.

Из нормативно-справочной и входной информации в результате обработки получается все обилие производной информации. Методы обработки могут быть самыми разными, от простой передачи, агрегирования и группировки до сложных экономических расчетов, граничащих с анализом. Производная информация, полученная в результате таких расчетов, отражает более сложные экономические характеристики производства: себестоимость разных видов продукции, производительность труда, рентабельность. Для сравнения текущей работы объекта с директивными заданиями часть производной информации выражается в тех же показателях, что и директивные задания. Однако производная информация неизмеримо шире по составу и форме представления, чем входная и выходная, поскольку она отражает все разнообразие используемой информации и сложность ее обработки. Она во многом зависит от типа экономико-математических моделей, которые используются при организации процессов управления. Получение производной информации является целью обработки и определяет выбор соответствующих методов.

В обработке как процессе преобразования информации можно выделить этап получения прогнозной, плановой и учетной информации. *Учетная информация* относится к фактическому процессу производства и его конкретным экономическим условиям. В нее входят как первичная информация, так и часть производной. Это данные о фактическом выпуске продукции по стоимости, номенклатуре и качеству, о затратах ресурсов, потерях, запасах, использовании оборудования, начисленной заработной плате, налоговых и премиальных выплатах и др. Они получаются из первичных данных с использованием принятых нормативов, тарифов, цен, ставок и других видов нормативно-справочной информации. Специфика учета как функции управления определяет специфику обработки учетной информации и необходимость ее первоочередной автоматизации.

Учетная информация описывает уже совершившиеся процессы, реально существующие условия. Она определенна и не зависит от последующих действий или принимаемых решений. В отличие от нее *прогнозная и плановая информация* может корректироваться при изменении условий или целей. Расчеты учетных и плановых данных образуют в некотором смысле встречные потоки. В процессе расчетов учетные данные в основном агрегируются, группируются и идут от натуральных показателей к стоимостным, а плановые данные — наоборот, от стоимостных и укрупненных натуральных показателей к детальным данным технической подготовки и обеспечения производства.

Учетная и плановая информация является основой всего процесса управления, включая регулирование, анализ, прогнозирование и другие функции. Так, оценка деятельности фирмы получается при прямом сравнении соответствующих показателей учетных и плановых данных. Вместе с нормативно-справочной учетная и плановая информация оказывается внутренней для органов управления и при создании информационных систем должна храниться в ее информационной базе. Над ней проводятся наиболее сложные виды обработки, обеспечивая решение управленческих задач. Окончательным итогом обработки можно считать получение *выходных данных*. Выходными их называют потому, что подразумевается управление ими (или контроль) со стороны вышестоящих (внешних) органов. Обычно данные при этом физически не удаляются, а участвуют в последующих циклах обработки. Часть из них поступает к объекту управления, а часть — к вышестоящим органам и другим подразделениям.

Информация, поступающая к объекту управления, оказывает нужные воздействия и тем самым влияет на его дальнейшую работу. Эта информация обычно вырабатывается с участием человека, определяется в процессе принятия решений, и поэтому ее называют управляющей[1]. Примером могут служить фонды экономического стимулирования, объем и структура инвестиций, перераспределение ресурсов и т. д.

Информация, поступающая к вышестоящим органам, имеет характер отчетности и может выдаваться в установленных статистическими и налоговыми органами формах. Она значительна по объему. Деление выходной информация на управляющую и отчетную оказывается условным, так как она может пересекаться, содержать одни и те же данные. К выходным относят также данные, поступающие к другим подразделениям и являющиеся для них осведомляющими.

Большая часть экономических данных выражена в форме *показателей*, состоящих из названия переменной величины, представленного набором качественных признаков, и количественного основания. Показатели оказываются теми элементами, с помощью которых описываются экономические объекты и процессы, причем минимальными элементами, достаточными для образования документа. Под обработкой данных понимается получение расчетным или аналитическим путем новых показателей, составление новых данных из имеющихся показателей.

Некоторую часть информации нельзя выразить через показатели. Например, такие, как методические материалы, структура объекта, модель его функционирования,

[1] Ясин Е. Т. Экономическая информация. Что это такое? М., 1976.

технология производства, методы решения задач и т. д. Такая информация не выражена в конкретно-числовой форме, но она может быть представлена в графической форме, с помощью блок-схемы или описана текстом. Она используется в экономическом управлении в неявном виде, задает условия принятия решений, служит для содержательного качественного анализа. При создании информационных систем на ее основе определяются методы и очередность (схема) обработки, а также реализующие их программы.

1.3.2. Характеристики экономической информации

Среди наиболее важных *характеристик* экономической информации, отражающих предъявляемые к ней требования, могут быть названы корректность, ценность, достоверность, точность, актуальность, полнота.

Говорят, что информация является *корректной*, если она обладает такими формой и содержанием, которые обеспечивают ее однозначное восприятие всеми потребителями.

Под *ценностью* понимают свойство информации, отражающее, в какой степени она способствует достижению целей и задач ее потребителя (например, управляющей системы).

Свойство *достоверности* связывает содержательную сторону информации как отражения некоторой объективной реальности с самой реальностью, а *точность* определяется мерой близости (удаленности) их друг от друга.

Понятие актуальности информации неявно подразумевает возможность изменений во времени состояния того объекта, к которому она относится. *Актуальность* информации отражает ее адекватность действительному состоянию референтного объекта.

Полнота информации отражает ее достаточность или недостаточность для принятия управленческого решения.

1.4. Место процессов обработки информации в управлении

С точки зрения кибернетики, управление разделяется на *обработку информации* и *принятие управленческих решений*. Практика подтверждает такое разделение и показывает, что принижение каждой из сторон снижает качество управления. Сам процесс управления является специфическим видом человеческой деятельности, в котором информация выступает как предмет труда и вся работа органов управления внешне представляется как обработка информации. Принятие решений является волевым актом, осуществляемым человеком по результатам обработки информации. Существенность данного деления и важность каждой из сторон процесса управления особенно ярко проявились в ходе массового создания *автоматизированных систем управления* (АСУ) в СССР в период социалистических методов хозяйствования. В то же время, аналогичный опыт накоплен и при автоматизации работы компаний в странах с развитой рыночной экономикой.

Недооценка роли обработки информации ведет к принятию волевых решений, их меньшим обоснованности и взаимной увязке. А это, в свою очередь, ведет к большим потерям и снижению эффективности производства. Как уже отмечалось, в современных условиях повышаются требования к обоснованности и согласованности принятия решений. Поэтому обработка информации требует все больших внимания и затрат. Совершенствование методов обработки является важным фактором улучшения управления. Повышение роли информации, обусловленное научно-техническим прогрессом, характерно для любых управленческих задач, как в управлении предприятиями, так и в государственном и муниципальном управлении. Менеджеры крупных фирм рассматривают автоматизацию обработки информации как ключ к коренным преобразованиям в методах управления фирмой. Информация считается таким же важным ресурсом в фирме, как рабочая сила, производственное оборудование, материалы и денежные средства. Она заслуживает не меньшего внимания и нуждается в не менее обстоятельном планировании, чем остальные ресурсы. Не являясь, собственно, ни одним из этих ресурсов, информация служит инфраструктурой, соединительной тканью, определяющей эффект соединения этих ресурсов.

Однако управление не ограничивается обработкой информации, и усиление роли ее обработки не должно приводить к недооценке второй стороны управления — принятия решения. Так, на первых этапах создания АСУ были попытки формализовать решение управленческих задач и свести его к обработке информации. Практика показала, что такая формализация возможна только для технических систем, а для экономического управления участие человека необходимо. Это можно объяснить сложностью экономических систем, наличием социальных и психологических факторов.

Неотъемлемым атрибутом экономического управления является развитая *система целей*, которая не может быть формализована. Кроме того, система целей должна поддерживаться соответствующей системой интересов и законодательных мер. В отличие от технических систем, каждый человек и отдельный коллектив, являющиеся объектом управления, имеет свои интересы и цели, не совпадающие с интересами и целями других участников. Согласование этих целей, контроль и стимулирование их выполнения входит в процесс управления и может выполняться только человеком.

Необходимость участия человека отражает принципиальную невозможность полной формализации экономического управления и является признаком, отличающим автоматизированные системы от автоматических. Все технические средства в управлении применяются для обработки информации, а само решение принимает всегда человек. В автоматических системах управления обработка информации может заканчиваться выработкой управляющего сигнала без участия человека (например, в автоматизированных системах управления технологическими процессами — АСУ ТП), но это не является собственно экономическим управлением.

Вместе с тем, разделение обработки информации и принятия решений оказывается условным. Представление управления как составленного из двух этапов, из которых первый — обработка информации предшествует второму — принятию

решения, является упрощенным и не раскрывает роли каждого из них. На самом деле эти процессы выполняются параллельно, и управление можно рассматривать как сложный итеративный процесс, в котором этапы обработки информации и принятия решений тесно переплетаются. Так, после некоторой обработки следует анализ того, какая информация должна обрабатываться и как. Такой анализ не является собственно принятием управленческого решения, но он относится к управлению. После него опять обрабатывается информация, затем снова анализ и т. д. Управляющая информация поступает в дальнейшую обработку, и принятие решений в управлении является не конечной точкой, а скорее узловыми моментам процесса управления.

Существуют управленческие задачи, где все процедуры обработки информации известны заранее, анализ по выбору соответствующих процедур может быть формализован. Как правило, это задачи, решаемые постоянно (с некоторым периодом). Их решение автоматизируется, в первую очередь, при создании АСУ. Но и в этих случаях человек не исключается из управления, за ним остаются функции контроля, окончательного принятия решений по всем узловым вопросам, возможность изменения постановки задачи. Те задачи, где потребности в информации заранее не известны, являются наиболее сложными. Разделение всех задач по принципу определенности их информационного обеспечения представляется весьма важным с точки зрения организации обработки информации и особенно ее автоматизации.

Понимание управления как сложного итеративного процесса показывает, что обработка информации не может быть изолирована от принятия решений. С другой стороны, современные тенденции управления, возросшие требования к обработке информации требуют ее дальнейшей специализации, создания специальных подразделений по обработке данных.

Возможность такой специализации, выделения процессов обработки информации из управления обусловлена наличием у информации ее собственных, присущих ей свойств: форм фиксации и представления, смысловых взаимосвязей. Методы ее обработки также имеют относительную независимость от процесса управления. Они в большей степени зависят от указанных свойств информации. По мере усложнения методы обработки все дальше отделяются от управления, требуют своего аппарата, методологии, их могут выполнять люди, не участвующие непосредственно в принятии решений. При этом информация обрабатывается в виде отдельных процедур, среди которых есть процедуры содержательной обработки, а есть обеспечивающие, например, создание и ведение массивов.

В условиях выделения обработки информации из управления вследствие специализации, с одной стороны, и при наличии необходимости соединения процессов обработки и принятия решений в единый процесс управления, с другой стороны, возникает задача органичного включения новых методов обработки информации в систему управления.

При создании автоматизированных систем эта задача всегда находится в центре внимания, поскольку информационное обеспечение проектируется вместе с функциональными подсистемами. Но при создании информационных систем это требование иногда недооценивается, особенно техническими специалистами,

которые основное внимание обращают на сложные методы обработки и организацию информационных массивов, а связь с органами управления стремятся формализовать, жестко фиксируя поступление входных данных в систему и выдачу результатов обработки. Это ведет к резкому сужению области использования и снижения эффективности создаваемой информационной системы. Обособление и выделение технических средств и формальных методов обработки не должно вести к содержательному отделению информационной системы от управления. Система должна иметь множество точек соприкосновения с использующими ее подразделениями органов управления.

Задача органичного включения машинных методов обработки информации должна решаться совместными усилиями работников органов управления и подразделений, отвечающих за программное и компьютерное обеспечение (вычислительных центров, отделов АСУ, управлений автоматизации и т. п.). Органами управления должны быть выделены процедуры обработки с четким заданием входной и выходной информации, особенно структуры и формы представления выходной информации. Специалисты вычислительных центров, в свою очередь, разрабатывают системы с гибкой структурой, способные выдавать информацию с учетом изменяющихся требований работников управления. Для этого основное внимание обращают на организацию системы хранения, передачи и поиска данных, способную выдавать информацию по произвольному набору признаков в режиме ответов на запросы. Разработчики информационных систем определяют формы запросов и обращения к данным, когда нужно, организуют передачу данных по сетям, формулируют требования к входной информации. Тем не менее, проблема согласования методов обработки с работой органов управления остается острой. На наш взгляд, многие трудности внедрения автоматизированных управляющих и информационных систем связаны с неудачным решением этой проблемы.

Совершенствование процессов обработки информации играет одну из ключевых ролей в комплексе мер по совершенствованию управления (хотя, конечно, далеко не исчерпывает их). В данном комплексе можно выделить *три основных* направления:

1. Совершенствование экономического строя государства в целом. К данному направлению относится определение глобальных задач, методов регулирования экономического развития со стороны государства как центрального звена управления, а также оценки и стимулирование деятельности экономических объектов. Решение этих содержательных вопросов требует глубокого качественного анализа, а полученные результаты фиксируются в законодательных актах, правительственных программах и постановлениях.
2. Совершенствование организации процесса управления. Это направление предполагает уже заданные цели и задачи развития. Оно в наибольшей степени затрагивает структуру органов управления и характер их взаимосвязей, изменяет процессы принятия решений. К нему относится стыковка функциональных подразделений и решаемых ими задач, определение содержания и схем

передачи документов в органы управления. Одной из форм такого совершенствования является создание автоматизированных управляющих систем.

3. Совершенствование информационного обеспечения. Данное направление нацелено на улучшение качественных характеристик той информации, которой «снабжаются» управляющие органы. Основной формой его реализации является создание информационных и экспертных систем. Если в АСУ внимание уделяется всему процессу управления, то в информационных системах — обработке информации.

Можно провести аналогию между этими направлениями и уровнями сложности контуров управления, рассматриваемых в общей теорией систем. На рис. 1.3 дана схема управления бизнес-процессом в рамках некоторого абстрактного (обобщенного) экономического объекта.

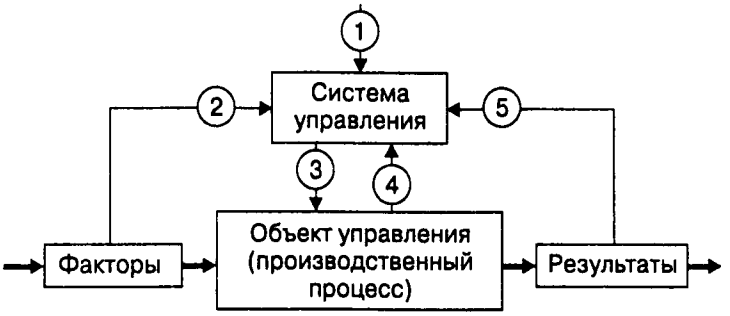


Рис. 1.3. Обобщенная схема управления бизнес-процессом

Цифрами обозначены виды информационных связей:

1. Информационные потоки, поступающие от внешних (управляющих, регулирующих и т. п.) органов.
2. Информация об условиях хозяйственной деятельности (наличных ресурсах, сроках поставок и др.).
3. Собственно управляющие воздействия — доведение принятых решений до объекта управления.
4. Информация о реализации управляющих воздействий.
5. Информация о результатах производства (например, выпущенная продукция, объем продаж и т. п.).

На этой схеме можно выделить три контура управления. *Низший контур регулирования* отражает классическую простейшую схему управления с обратной связью. Он включает получение обратной связи и обработку данных в регуляторе. На схеме регулятором является система управления, обратная связь представлена четвертым и пятым видами информационных связей. Она дополняется предварительной информацией второго вида об имеющихся и возможных факторах (ресурсах). Получение такой информации и ее обработка составляют информационное обеспечение управления. Мероприятия по совершенствованию

управления, проводимые в рамках данного контура, могут быть отнесены к третьему из перечисленных выше направлений.

Способность систем к изменению регулятора характеризует *второй*, более сложный контур самоорганизующихся систем. Применительно к экономике это означает:

- организационное изменение системы управления;
- изменение связанных с управляющей системой процессов принятия решений (методов решения управленческих задач);
- доведение принятых решений до объекта управления (3-й вид информационной связи на рис. 1.3).

Контур самоорганизации (он соответствует второму направлению совершенствования управления) предполагает неизменными цели, критерии их достижения и методы оценки.

Еще более сложные системы, способные изменять критерии и методы оценки, называют *самонастраивающимися*. На схеме этот контур управления представлен первым видом информационных связей. Данному, третьему, уровню сложности контура управления соответствует первое из названных направлений.

Приведенные направления обладают относительной независимостью, которая выражается в различии подходов, постановки проблем и методов их решения. Эта независимость означает возможность организационного совершенствования управления в условиях законодательно определенной хозяйственно-экономической системы, а также отдельное совершенствование информационного обеспечения при заданной организационной структуре управления.

Изменения в более высоких контурах управления практически всегда ведут к изменению низших контуров. Поэтому указанные направления не могут рассматриваться изолированно, их независимость является относительной и в большей мере проявляется в низших контурах управления по отношению к высшим.

Особенно тесно связаны обработка информации с организацией управления. Изменение организационной структуры и состава решаемых задач почти автоматически вызывает необходимость изменений в информационном обеспечении. Одновременно совершенствование методов обработки информации оказывает обратное воздействие, становится эффективным только при определенных условиях, предъявляет требования к организации управления. Иногда высказывается мнение о невозможности создания информационных систем без изменений в структуре управления. Однако практика показывает не только возможность, но и оправданность выделения специальных систем по обработке информации, поскольку они позволяют лучше выявить потребности органов управления, сконцентрировать усилия на методах обработки. Кроме того, по сравнению с «полномасштабными» автоматизированными управляющими системами, разработка информационных систем обычно требует меньших затрат как на предпроектный анализ, так и на само их создание.

1.5. Задачи экономической информатики на современном этапе

Остановимся на ряде задач, которые характерны для экономической информатики на современном этапе ее развития.

Успехи в развитии информатики создали мощный аппарат для семантического анализа и обработки текста, но наиболее эффективным он оказался для информационно-поисковых систем библиотечного типа, обрабатывающих научно-техническую информацию. Методы экономической информатики, построение тезаурусов и специальных языков описания показателей позволяют автоматизировать смысловую обработку экономических документов как в форме показателей, так и в текстовой форме. Можно ожидать, что в будущем они найдут большее применение на высших уровнях управления, где характер решаемых задач в наибольшей степени предполагает использование информации в описательной форме.

Современные программные средства позволяют вести достаточно развернутую обработку визуальной и графической информации. Это нашло широкое применение в системах автоматизации деятельности средств массовой информации, программном обеспечении проектно-конструкторских работ, игровых программах. В то же время, для экономической информации графическая обработка затруднена сложностью смысловых взаимосвязей, и обычно она ограничивается выдачей временных и структурных графиков. Видимо, одно из перспективных направлений развития экономической информатики связано с качественным переломом данной ситуации и переходом к активному использованию аудиовизуальных (мультимедийных) информационных технологий.

Заметное место в современных работах по экономической информатике занимает концепция *электронного офиса*. Она базируется на идеях гибкого встраивания современных технологий обработки информации в деятельность традиционных офисов (органов управления предприятиями и фирмами). Конечной целью реализации концепции электронного офиса является сведение к минимуму внутриорганизационных издержек процесса управления за счет перехода от традиционных (бумажных) представлений документов к их электронным аналогам. Последнее, в конечном счете, позволяет руководству и сотрудникам офисных служб сконцентрироваться на непосредственных задачах их деятельности. Среди основных функций электронного офиса могут быть названы:

- обеспечение жизненного цикла документов (создание, оформление, регистрация, верификация, хранение, удаление);
- обеспечение совместного доступа к документам;
- обеспечение автоматизированного процесса движения документов в соответствии с установленными технологиями их обработки;
- контроль исполнения;

- ведение индивидуальных расписаний деятельности сотрудников, согласованных в рамках единого плана работы офиса;
- служебные функции процесса документооборота (стилистическое и полиграфическое оформление, тиражирование, формирование производных отчетов, служба консультаций и т. д.).

Основные выводы

Кратко сформулируем основные положения данной главы.

- Кибернетика и входящая в нее статистическая теория информации не учитывают наиболее характерные черты обработки данных в экономическом управлении. Поэтому создание информационных систем не может базироваться на статистической теории и необходимо рассмотрение смыслового содержания экономической информации, форм ее выражения и характера использования в управлении. Последнее и обусловило **выделение экономической информатики как самостоятельной научной дисциплины**.
- Существенными признаками классификации экономической информации являются ее назначение и источники получения. По этим признакам выделяется входная, внутренняя и выходная информация, а также учетная и плановая информация, образующие в процессе обработки встречные потоки.
- Обработка информации и принятие решения образуют две стороны единого процесса управления. Выделение обработки данных из управления и передача ее специально организованному подразделению обусловлено дальнейшей специализацией управления и возможно только при условии сохранения тесных связей аппарата управления с системой обработки.

Кардинальным отличием данной главы является то, что в ней фокус внимания был сконцентрирован на целях и задачах процессов управления и обработки экономической информации. Теперь будет уместным задаться вопросом: «Каким способом эти цели могут быть достигнуты?» Ответ на него должны дать последующие двенадцать глав настоящей книги.

Ключевые понятия

- информация;
- информатика;
- система;
- данные;
- знания;
- структура и форма информации;
- бит, байт, килобайт...;
- информационные технологии;

- управление;
- кибернетика;
- экономическая информация;
- электронный офис.

Контрольные вопросы

1. Дайте определение и развернутую характеристику термина «информация».
2. Что такое данные и знания?
3. Какие принципиальные идеи заложены в методах измерения информации?
4. Какой смысл вкладывается в понятие «информационные технологии»? В чем заключается их специфика по сравнению с технологиями индустриальными?
5. Дайте определения термина «экономическая информация».
6. Какие факторы обусловливают выделение экономической информатики как самостоятельного направления?
7. Перечислите свойства, характеризующие качество экономической информации.
8. Перечислите виды экономической информации.
9. Кратко перечислите основные виды информационных связей, присутствующих в обобщенной схеме управления бизнес-процессом.
10. Перечислите и охарактеризуйте основные контуры данной схемы.
11. Дайте характеристику современных задач экономической информатики.

Литература

1. Информатика / Под ред. Н. В. Макаровой. М.: Финансы и статистика, 1997.
2. Основы экономической информатики/ Под ред. А. Н. Морозевича. ООО «Миссанта». Минск, 1998.
3. Экономическая информатика / Под ред. В. В. Евдокимова. СПб.: Питер, 1997.
4. Ясин Е. Т. Экономическая информация. Что это такое? М., 1976.

Глава 2. История и направления развития вычислительной техники

XX век характеризуется необходимостью обрабатывать огромное количество информации. Для сбора, хранения, использования и распространения большого объема информации необходимо специальное устройство. Таким устройством является компьютер. В настоящее время компьютеры представлены практически во всех областях жизни человека. Для того чтобы полно оценить влияние компьютеров на жизнь человека и его будущее, необходимо понять, как проходила их эволюция.

2.1. Эволюция средств вычислительной техники

Современным компьютерам предшествовали механические и электромеханические устройства. В 1642 году французский математик и философ Блез Паскаль в возрасте 18 лет сконструировал суммирующую машину. Машина Паскаля состояла из восьми движущихся дисков с прорезями и могла суммировать числа до восьми знаков. Для своей машины Паскаль использовал десятичную систему исчисления. Например, если первый диск смещался на десять прорезей, что составляло его полный оборот, он перемещал следующий диск на одну позицию и, таким образом, увеличивал количество десятков на один. Когда диск, представляющий десятки, делал полный оборот, он смещал следующий диск, увеличивая количество сотен, и т. д.

Известны и более ранние попытки создания механических суммирующих машин. Описание суммирующей машины, напоминающей по характеристикам машину Паскаля, в 1967 году было обнаружено в записках, принадлежащих Леонардо да Винчи. Подобное устройство также было описано в 1623 году Вильгельмом Шикардом. До наших дней дошли только чертежи Шикарда, обнаруженные в 1956 году.

В 1694 году немецкий математик и философ Готфрид Вильгельм Лейбниц, используя чертежи и рисунки Паскаля, улучшил машину Паскаля, добавив возможность перемножать числа. Вместо обычных шестеренок Лейбниц использовал пошаговый барабан.



Рис. 2.1. Блез Паскаль (1623 — 1662)



Рис. 2.2. Готфрид Вильгельм Лейбниц (1646 — 1716)

Однако широкое распространение вычислительные аппараты получили только в 1820 году, когда француз Чарльз Калмар изобрел машину, которая могла производить четыре основных арифметических действия. Машину Калмара назвали *арифмометр*. Благодаря своей универсальности, арифмометры использовались довольно длительное время. Многие ученые и изобретатели совершенствовали эти устройства. Так, швед, живший в России, Вильгодт Однер в 1880 году создал арифмометр, в котором использовалось переменное число зубцов. Позднее на основе арифмометра Однера был создан арифмометр «Феликс», выпускавшийся в СССР вплоть до 70-х годов.

Начало эры компьютеров в том виде, в котором они существуют сейчас, связано с именем английского математика Чарльза Бэббиджа, который в 30-х годах XIX века предложил идею вычислительной машины, осуществленную лишь в середине XX века. Бэббидж обратил внимание на то, что машина может без ошибок выполнять вычисление больших математических таблиц посредством простого повторения шагов. Работая над этой проблемой, в 1822 году Бэббидж предложил проект машины для решения дифференциальных уравнений. Для повторения операций в машине Бэббиджа должна была использоваться энергия пара. Таким образом, сам процесс вычислений действительно был автоматизирован, то есть проходил

1493213
РашГТУ

без участия человека. В дальнейшем Бэббидж решил создать модель универсальной вычислительной машины, способной выполнять широкий круг задач. Он назвал ее аналитической машиной.

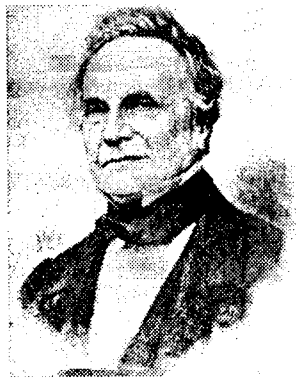


Рис. 2.3. Чарльз Бэббидж (1792 — 1871)

У аналитической машины Бэббиджа были все основные черты современного компьютера. Состоящая более чем из 50 000 компонентов аналитическая машина включала устройство ввода информации, блок управления, запоминающее устройство и устройство вывода результатов. Аналитическая машина могла выполнять определенный набор инструкций, которые записывались на перфокартах. *Перфокарты* представляли собой прямоугольные карточки из картона. Каждой инструкции аналитической машины соответствовала определенная последовательность дырочек, которые пробивались на перфокартах, а затем с помощью устройства ввода поступали в блок управления. Хотя аналитическая машина в том виде, в котором ее задумывал Бэббидж, так и не была создана, идеи, заложенные Бэббиджем, оказали огромное влияние на развитие вычислительной техники. Автоматизация вычислений, универсальность вычислительной машины, набор внутренних инструкций, общая конструктивная схема, организация ввода и вывода информации — все эти элементы впоследствии были использованы при создании компьютера.

В 1889 году американский изобретатель Герман Холлерит сконструировал перфокарточное устройство для решения статистических задач. В отличие от идеи Бэббиджа, хранить на перфокартах инструкции, Холлерит использовал перфокарты для хранения данных. Кроме того, для работы перфокарточного устройства использовалось электричество. Цифры на перфокарте изображались одинарными отверстиями, а буквы алфавита — двойными. Специальный электрический прибор опознавал отверстия на перфокартах и посылал сигналы в обрабатывающее устройство. Вычислительная машина Холлерита оказалась по тем временам очень быстрым устройством обработки данных, а перфокарты — удобным способом хранения данных. Машина Холлерита была использована для обработки результатов переписи населения США. Обработка результатов предыдущей переписи 1880 года заняла около 10 лет. За это время успело вырасти новое поколение американцев. С помощью машины Холлерита те же данные были обработаны всего за шесть недель. В 1896 году Холлерит основал компанию по производству пер-

форирующих устройств — Tabulating Machine Company, которая в 1924 году после серии слияний и поглощений превратилась в знаменитую компанию по производству компьютеров — IBM (International Business Machines).

Кроме механических и электромеханических вычислительных машин появились также аналоговые вычислительные машины, в которых обработка информации происходила с помощью специально подобранного физического процесса, моделирующего вычисляемую закономерность. Простейшей аналоговой вычислительной машиной являются часы. Первыми аналоговыми машинами были устройства, в которых главными элементами были интегрирующие и дифференцирующие устройства, позволяющие мгновенно вычислять интеграл и производную заданной функции, отслеживая ее изменение во времени.

Полезным свойством аналоговой вычислительной машины является практически мгновенное получение решения после задания необходимых параметров задачи и установления моделирующего физического процесса. Однако круг задач, которые может решать аналоговая машина, ограничен теми физическими процессами, которые она в состоянии моделировать. Кроме того, точность решения аналоговой машины часто недостаточна для определенного круга задач, а повышение точности связано со значительным ростом стоимости вычислений.

С другой стороны, механические и электромеханические вычислительные машины, предназначенные для решения сложных задач, требуют наличия огромного количества элементов для представления чисел и связей между ними, что существенно усложняет их работу.

Решая эту проблему, американцы Джон Атанасов и Клиффорд Берри в 1940 году разработали модель полностью электронного компьютера, использующего единую систему представления чисел и связей между ними — *булеву алгебру*. Их подход базировался на работах английского математика XIX века Джорджа Буля, посвященных аппарату символической логики. В основе булевой алгебры лежит интерпретация элементов булевой алгебры как высказываний, принимающих значение «истина» или «ложь». Атанасов и Берри применили эту концепцию для электронных устройств. Истине соответствовало прохождение электрического тока, а лжи — его отсутствие. Для представления чисел Атанасов и Берри предложили использовать двоичную систему исчисления.



Рис. 2.4. Джон Атанасов



Рис. 2.5. Клиффорд Берри

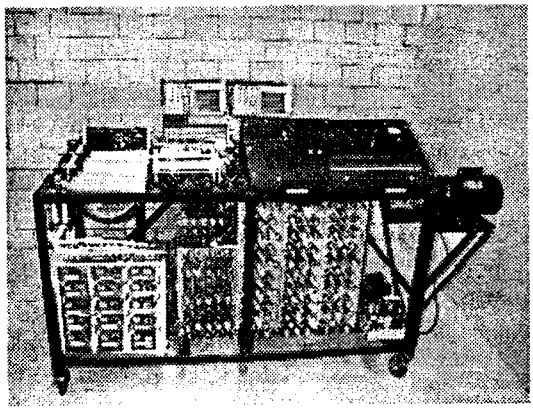


Рис. 2.6. Вычислительная машина Атанасова — Берри

В 1936 году английский математик Алан Тьюринг опубликовал работу «О вычислимых числах», заложив теоретические основы теории алгоритмов. Концепция Тьюринга возникла в результате проведенного им анализа действий человека, выполняющего в соответствии с заранее разработанным планом те или иные вычисления, то есть последовательные преобразования знаковых комплексов. Анализ этот, в свою очередь, был осуществлен им с целью решения проблемы поиска точного математического эквивалента для общего интуитивного представления об алгоритме. Работа Тьюринга стимулировала возникновение абстрактной теории автоматов и во многом определила ее особенности.

В своей работе Тьюринг описал абстрактную вычислительную машину, которая получила название *машины Тьюринга*. Машина Тьюринга представляет собой автоматическое устройство, способное находиться в конечном числе внутренних состояний и снабженное бесконечной внешней памятью — лентой. Среди состояний выделяются два — начальное и конечное. Лента разделена на клетки. В каждую клетку может быть записана любая из букв некоторого алфавита. В пустую клетку записана «пустая буква». В каждый момент времени машина Тьюринга находится в одном из своих состояний и, рассматривая одну из клеток ленты, воспринимает записанный в ней символ.

В неконечном состоянии машина Тьюринга совершает шаг, который определяется ее текущим состоянием и символом на ленте, воспринимаемым в данный момент. Шаг машины Тьюринга заключается в следующем:

1. В рассматриваемой клетке записывается символ, совпадающий со старым, или пустой.
2. Машина переходит в новое состояние, совпадающее со старым, или конечное.
3. Лента сдвигается на одну клетку или остается на месте.

Перечисление всех возможных шагов машины Тьюринга, в зависимости от текущей комбинации неконечного состояния и воспринимаемого символа называется программой данной машины Тьюринга. Конфигурация машины Тьюринга

определяется конкретным заполнением клеток ленты символами и внутренним состоянием, в котором машина находится. Если зафиксировать какую-либо неконечную конфигурацию машины в качестве исходной, то работа машины будет заключаться в последовательном преобразовании исходной конфигурации в соответствии с программой машины до тех пор, пока не будет достигнуто конечное состояние.

Тьюринг не преследовал цели изобрести компьютер. Тем не менее, описанная им абстрактная машина определила некоторые характеристики современных компьютеров. Так, например, бесконечная лента является аналогом оперативной памяти современного компьютера. Впервые подобная модель памяти была использована в компьютере Атанасова и Берри.



Рис. 2.7. Алан Матисон Тьюринг (1912 — 1954)

2.2. Поколения современных компьютеров

Развитие вычислительной техники в современном периоде принято рассматривать с точки зрения смены поколений компьютеров. Каждое поколение компьютеров в начальный момент развития характеризуется качественным скачком в росте основных характеристик компьютера, вызванным обычно переходом на новую элементную базу, а также относительной стабильностью архитектурных и технологических решений.

Разбиение поколений компьютеров по годам весьма условно. В то время, как начиналось активное использование компьютеров одного поколения, создавались предпосылки для возникновения следующего. Кроме элементной базы и временного интервала используются следующие показатели развития компьютеров одного поколения: быстродействие, архитектура, программное обеспечение, уровень развития внешних устройств. Другим важным качественным показателем является широта области применения компьютеров.

2.2.1. Первое поколение компьютеров (1945—1956 годы)

С началом второй мировой войны правительства разных стран начали разрабатывать вычислительные машины, осознавая их стратегическую роль в ведении войны. Увеличение финансирования в значительной степени стимулировало развитие вычислительной техники. В 1941 году немецкий инженер Конрад Цузе разработал вычислительную машину Z2, выполнявшую расчеты, необходимые при проектировании самолетов и баллистических снарядов. В 1943 году английские инженеры завершили создание вычислительной машины для дешифровки сообщений немецкой армии, названной «Колосс». Однако эти устройства не были универсальными вычислительными машинами, они предназначались для решения конкретных задач.

В 1944 году американский инженер Говард Эйкен при поддержке фирмы IBM сконструировал компьютер для выполнения баллистических расчетов. Этот компьютер, названный «Марк I», по площади занимал примерно половину футбольного поля и включал более 600 километров кабеля. В компьютере «Марк I» использовался принцип электромеханического реле, заключающийся в том, что электромагнитные сигналы перемещали механические части. «Марк I» был довольно медленной машиной: для того чтобы произвести одно вычисление требовалось 3-5 с. Однако, несмотря на огромные размеры и медлительность, «Марк I» стал более универсальным вычислительным устройством, чем машина Цузе или «Колосс». «Марк I» управлялся с помощью программы, которая вводилась с перфоленты. Это дало возможность, меняя вводимую программу, решать довольно широкий класс математических задач.

В 1946 году американские ученые Джон Мокли и Дж. Преспер Эккерт сконструировали электронный вычислительный интегратор и калькулятор (ЭНИАК) — компьютер, в котором электромеханические реле были заменены на электронные вакуумные лампы. Применение вакуумных ламп позволило увеличить скорость работы ЭНИАК в 1000 раз по сравнению с «Марк I». ЭНИАК состоял из 18 000 вакуумных ламп, 70 000 резисторов, 5 миллионов соединительных спаек и потреблял 160 кВт электрической энергии, что по тем временам было достаточно для освещения большого города. Между тем, ЭНИАК стал работающим прообразом современного компьютера. Во-первых, ЭНИАК был основан на полностью цифровом принципе обработки информации. Во-вторых, ЭНИАК стал действительно универсальной вычислительной машиной, он использовался для расчета баллистических таблиц, предсказания погоды, расчетов в области атомной энергетики, аэродинамики, изучения космоса.

Следующий важный шаг в совершенствовании вычислительной техники сделал американский математик Джон фон Нейман. Ранние вычислительные машины могли выполнять только команды, поступающие извне, причем команды выполнялись поочередно. Хотя использование перфокарт позволяло упростить процесс ввода команд, тем не менее, часто процесс настройки вычислительной машины и ввода команд занимал больше времени, чем собственно решение поставленной задачи. Фон Нейман предложил включить в состав компьютера для хранения последовательности команд и данных специальное устройство — память. Кроме

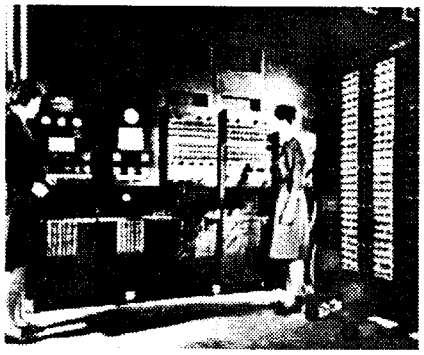


Рис. 2.8. Одна из панелей управления ЭНИАКа (1946)

того, Джон фон Нейман предложил реализовать в компьютере возможность передачи управления от одной программы к другой. Возможность хранить в памяти компьютера разные наборы команд (программы), приостанавливать выполнение одной программы и передавать управление другой, а затем возвращаться к исходной значительно расширяла возможности программирования для вычислительных машин. Другой ключевой идеей, предложенной фон Нейманом, стал процессор (центральное обрабатывающее устройство), который должен был управлять всеми функциями компьютера. В 1945 году Джон фон Нейман подготовил отчет, в котором определил следующие основные принципы работы и элементы архитектуры компьютера:

1. Компьютер состоит из процессора (центрального обрабатывающего устройства), памяти и внешних устройств.
2. Единственным источником активности (не считая стартового или аварийного вмешательства человека) в компьютере является процессор, который, в свою очередь, управляется программой, находящейся в памяти.
3. Память компьютера состоит из ячеек, каждая из которых имеет свой уникальный адрес. Каждая ячейка хранит команду программы или единицу обрабатываемой информации. Причем и команда, и информация имеют одинаковое представление.
4. В любой момент процессор выполняет одну команду программы, адрес которой находится в специальном регистре процессора — счетчике команд.
5. Обработка информации происходит только в регистрах процессора. Информация в процессор поступает из памяти или от внешнего устройства.
6. В каждой команде программы зашифрованы следующие предписания: из каких ячеек взять обрабатываемую информацию; какие операции совершить с этой информацией; в какие ячейки памяти направить результат; как изменить содержимое счетчика команд, чтобы знать, откуда взять следующую команду для выполнения.
7. Процессор исполняет программу команда за командой в соответствии с изменением содержимого счетчика команд до тех пор, пока не получит команду остановиться.



Рис. 2.9. Джон фон Нейман (1903 — 1957)

В дальнейшем *архитектура фон Неймана* незначительно изменялась и дополнялась, но исходные принципы управления работой компьютера с помощью хранящихся в памяти программ остались нетронутыми. Подавляющее большинство современных компьютеров построено именно по архитектуре фон Неймана. В 1951 году был создан первый компьютер, предназначенный для коммерческого использования, — УНИВАК (универсальный автоматический компьютер), в котором были реализованы все принципы архитектуры фон Неймана. В 1952 году с помощью УНИВАК был предсказан результат выборов президента США.

Работы по созданию вычислительных машин велись и в СССР. Так, в 1950 году в Институте электроники Академии наук Украины под руководством академика Сергея Алексеевича Лебедева была разработана и введена в эксплуатацию МЭСМ (малая электронная счетная машина). МЭСМ стала первой отечественной универсальной ламповой вычислительной машиной в СССР. В 1952—1953 годах МЭСМ оставалась самой быстродействующей (50 операций в секунду) вычислительной машиной в Европе. Принципы построения МЭСМ были разработаны С. А. Лебедевым независимо от аналогичных работ на Западе.

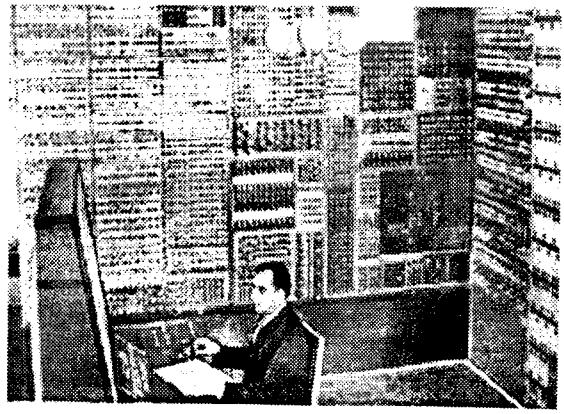


Рис. 2.10. МЭСМ — малая электронная счетная машина (1952)

В компьютерах первого поколения использовался *машинный язык* — способ записи программ, допускающий их непосредственное исполнение на компьютере. Программа на машинном языке представляет собой последовательность машинных команд, допустимых для данного компьютера. Процессор непосредственно воспринимает и выполняет команды, выраженные в виде двоичных кодов. Для каждого компьютера существовал свой собственный машинный язык. Это также ограничивало область применения компьютеров первого поколения.

Появление первого поколения компьютеров стало возможно благодаря трем техническим новшествам: электронным вакуумным лампам, цифровому кодированию информации и созданию устройств искусственной памяти на электростатических трубках. Компьютеры первого поколения имели невысокую производительность: до нескольких тысяч операций в секунду. В компьютерах первого поколения использовалась архитектура фон Неймана. Средства программирования и программное обеспечение еще не были развиты, использовался низкоуровневый машинный язык. Область применения компьютеров была ограничена.

2.2.2. Второе поколение компьютеров (1956—1963 годы)

Электронные вакуумные лампы выделяли большое количество тепла, поглощали много электрической энергии, были громоздкими, дорогими и ненадежными. Как следствие, компьютеры первого поколения, построенные на вакуумных лампах, обладали низким быстродействием и невысокой надежностью. В 1947 году сотрудники американской компании «Белл» Уильям Шокли, Джон Бардин и Уолтер Бреттейн изобрели транзистор. Транзисторы выполняли те же функции, что и электронные лампы, но использовали электрические свойства полупроводников. По сравнению с вакуумными трубками транзисторы занимали в 200 раз меньше места и потребляли в 100 раз меньше электроэнергии. В то же время появляются новые устройства для организации памяти компьютеров — *ферритовые сердечники*. С изобретением транзистора и использованием новых технологий хранения данных в памяти появилась возможность значительно уменьшить размеры компьютеров, сделать их более быстрыми и надежными, а также значительно увеличить емкость памяти компьютеров.

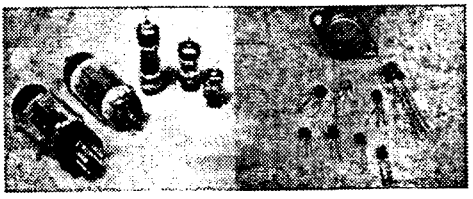


Рис. 2.11. Электронные вакуумные лампы (слева) и транзисторы (справа)

В 1954 году компания Texas Instruments объявила о начале серийного производства транзисторов, а в 1956 году ученые Массачусетского технологического института создали первый полностью построенный на транзисторах компьютер TX-O.

Машинный язык, применявшийся в первом поколении компьютеров, был крайне неудобен для восприятия человеком. Числовая кодировка операций, адресов ячеек и обрабатываемой информации, зависимость вида программы от ее места в памяти не давали возможности следить за смыслом программы. Для преодоления этих неудобств был придуман *язык ассемблер*. Для записи кодов операций и обрабатываемой информации в ассемблере используются стандартные обозначения, позволяющие записывать числа и текст в общепринятой форме, а для кодов команд — принятые мнемонические обозначения. Для обозначения величин, размещаемых в памяти, можно применять любые имена, отвечающие смыслу программы. После ввода программы ассемблер сам заменяет символические имена на адреса памяти, а символические коды команд на числовые. Использование ассемблера сделало процесс написания программ более наглядным.

В конце 50-х — начале 60-х годов компьютеры второго поколения стали интенсивно использоваться государственными организациями и крупными компаниями для решения различных задач. К 1965 году большая часть крупных компаний обрабатывала финансовую информацию с помощью компьютеров. Постепенно они приобретали черты современного нам компьютера. Так, в этот период были сконструированы такие устройства, как графопостроитель и принтер, носители информации на магнитной ленте и магнитных дисках и др.

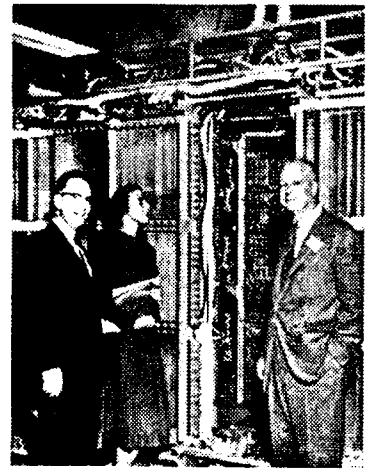


Рис. 2.12. Один из самых мощных компьютеров второго поколения — LARC и его создатели (1958)

Расширение области применения компьютеров потребовало создания новых технологий программирования. Программное обеспечение, написанное на языке ассемблер для одного компьютера, было непригодно для работы на другом компьютере. По этой причине, в частности, не удавалось создать стандартную операционную систему — основную управляющую программу компьютера, так как каждый производитель компьютеров разрабатывал свою операционную систему на своем ассемблере.

Специалисты, использующие в своей деятельности компьютеры, вскоре ощутили потребность в более естественных языках, которые бы упрощали процесс программирования, а также позволяли переносить программы с одного компьютера на другой. Подобные языки программирования получили название языков высокого уровня. Для их использования необходимо иметь компилятор (или интерпретатор), то есть программу, которая преобразует операторы языка в машинный язык данного компьютера.

Одним из первых языков программирования высокого уровня стал Фортран (FORTRAN – FORmula TRANslation), который предназначался для естественного выражения математических алгоритмов и стал необычайно популярен среди ученых. На Фортране можно писать большие программы, разбивая задачу на несколько частей (подпрограммы), которые программируются отдельно, а затем объединяются в единое целое. Так как Фортран предназначен в основном для вычислений, в нем отсутствовали развитые средства работы со структурами данных. Этот недостаток был исправлен в языке Кобол (COBOL – Common Business Oriented Language). Кобол специально предназначался для обработки финансово-экономических данных. Кроме того, разработчики постарались сделать Кобол максимально похожим на естественный английский язык, что позволило писать программы на этом языке даже неспециалистам в программировании. Со вторым поколением компьютеров началось развитие индустрии программного обеспечения.

В целом, данный период развития вычислительной техники характеризуется применением для создания компьютеров транзисторов и памяти на ферритовых сердечниках, увеличением быстродействия компьютеров до нескольких сотен тысяч операций в секунду, возникновением новых технологий программирования, языков программирования высокого уровня, операционных систем. Компьютеры второго поколения получили широкое распространение, они использовались для научных, инженерных и финансовых расчетов, для обработки больших объемов данных на предприятиях, в банках, государственных организациях.

2.2.3. Третье поколение компьютеров (1964—1971 годы)

В 1958 инженер компании Texas Instruments Джек Килби предложил идею *интегральной микросхемы* — кремниевого кристалла, на который монтируются миниатюрные транзисторы и другие элементы. В том же году Килби представил первый образец интегральной микросхемы, содержащий пять транзисторных элементов на кристалле германия. Микросхема Килби занимала чуть больше сантиметра площади и была несколько миллиметров толщиной. Год спустя, независимо от Килби, Роберт Нойс разработал интегральную микросхему на основе кристалла кремния. Впоследствии Роберт Нойс основал компанию «Интел» по производству интегральных микросхем. Микросхемы работали значительно быстрее транзисторов и потребляли значительно меньше энергии.

Первые интегральные микросхемы состояли всего из нескольких элементов. Однако, используя полупроводниковую технологию, ученые довольно быстро на-

учились размещать на одной интегральной микросхеме сначала десятки, а затем сотни и больше транзисторных элементов.

В 1964 году компания IBM выпустила компьютер IMB System 360, построенный на основе интегральных микросхем. Семейство компьютеров IBM System 360 — самое многочисленное семейство компьютеров третьего поколения и одно из самых удачных в истории вычислительной техники. Выпуск этих компьютеров можно считать началом массового производства вычислительной техники. Всего было выпущено более 20 000 экземпляров System 360.

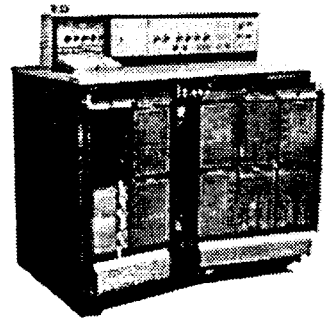


Рис. 2.13. Один из самых популярных компьютеров третьего поколения IBM System 360 (1964)

IMB System 360 относится к классу так называемых *мэйнфреймов*. Компания DEC (Digital Equipment Corporation) представила модель миникомпьютера PDP-8. *Мини-компьютеры*, или компьютеры средней производительности, характеризуются высокой надежностью и сравнительно низкой стоимостью. Низкая по сравнению со стоимостью суперкомпьютеров стоимость мини-компьютеров позволила начать применять их в небольших организациях — исследовательских лабораториях, офисах, на небольших промышленных предприятиях.

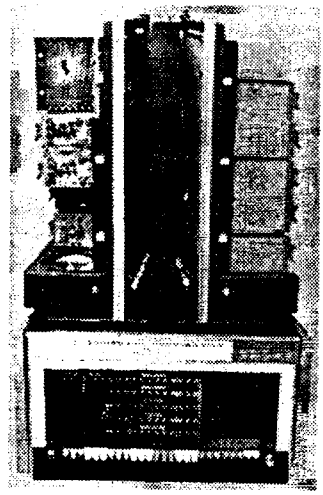


Рис. 2.14. Первый миникомпьютер PDP-8

В то же время проходило совершенствование программного обеспечения. Операционные системы строились таким образом, чтобы поддерживать большее количество внешних устройств, появились первые коммерческие операционные системы и новые прикладные программы. В 1968 году на одной из конференций Дуглас Энгельбарт из Станфордского института продемонстрировал созданную им систему взаимодействия компьютера с пользователем, состоящую из клавиатуры, указателя «мышь» и графического интерфейса, а также некоторые программы, в частности текстовый процессор и систему гипертекста.

В 1964 году появился язык программирования Бейсик (BASIC — Beginner's All-Purpose Symbolic Instruction Code), предназначенный для обучения начинающих программистов. Бейсик обеспечивал быстрый ввод и проверку программ. Бейсик не очень подходил для написания серьезных программ, однако он давал общее представление о программировании и позволял многим далеким от компьютеров людям быстро овладеть основными навыками программирования. В 1970 году швейцарец Никлас Вирт разработал язык программирования Паскаль, также предназначенный для обучения принципам программирования. Создававшийся как язык для обучения, Паскаль оказался очень удобен для решения многих прикладных задач. Он прекрасно обеспечивал применение методов структурного программирования, что стало необходимо при создании больших программных систем.

Основой для компьютеров третьего поколения послужили интегральные микросхемы, что позволило значительно уменьшить стоимость и размеры компьютеров, началось массовое производство компьютеров. В данный период развития вычислительной техники продолжалось увеличение скорости обработки информации. Компьютеры третьего поколения работали со скоростью до одного миллиона операций в секунду. Появились новые внешние устройства, облегчающие взаимодействие человека с компьютером. Увеличение быстродействия компьютеров и области их применения потребовало разработки новых методов создания программного обеспечения. Появились первые коммерческие операционные системы реального времени, специально разработанные для них языки программирования высокого уровня. Область применения компьютеров третьего поколения необычайно широка: системы обработки данных, управления, проектирования, решения различных коммерческих задач.

2.2.4. Четвертое поколение компьютеров (с 1971 года и по настоящее время)

В 1965 году председатель совета директоров компании «Интел» Гордон Мур предположил, что количество элементов на интегральных микросхемах должно удваиваться каждые 18 месяцев. В дальнейшем это правило, известное как *закон Мура*, было применено к скорости микропроцессоров и до сих пор не нарушалось.

В 1969 году компания «Интел» выпустила еще одно важное для развития вычислительной техники устройство — *микропроцессор*. Микропроцессор представляет собой интегральную микросхему, на которой сосредоточено обрабатывающее устройство с собственной системой команд. Конструкция микропроцессора по-

зволяет применять его для решения широкого круга задач, создавая при этом различные функциональные устройства. Использование микропроцессоров значительно упростило конструкцию компьютеров. Практически сразу микропроцессоры получили широкое применение в различных системах управления от космических аппаратов до бытовых приборов.

В течение следующих десятилетий, следуя закону Мура, продолжалось все большее увеличение скорости и интеграции микропроцессоров. Появились сверхбольшие интегральные схемы, включающие сотни тысяч и даже миллионы элементов на один кристалл. Это позволило продолжить уменьшение размеров и стоимости компьютеров и повысить их производительность и надежность.

Практически одновременно с микропроцессорами появились *микрокомпьютеры*, или *персональные компьютеры*, отличительной особенностью которых стали небольшие размеры и низкая стоимость. Благодаря своим характеристикам персональные компьютеры предоставили возможность практически любому человеку познакомиться с вычислительной техникой. Компьютеры перестали быть прерогативой крупных компаний и государственных учреждений, а превратились в товар массового потребления.

Одним из пионеров в производстве персональных компьютеров была компания Apple. Ее основатели Стив Джобс и Стив Возняк собрали первую модель персонального компьютера в 1976 году и назвали ее Apple I. В 1977 году они представили свой компьютер членам компьютерного клуба в Калифорнии и на следующий день получили заказ на 50 подобных компьютеров. Стоимость первого персонального компьютера составляла всего 500 долларов. В том же 1977 году компания Apple представила следующую модель персонального компьютера — Apple II. У новой модели был изящный пластиковый корпус со встроенной клавиатурой. Впервые компьютер приобрел черты бытового прибора. Продажи персональных компьютеров резко возросли. Apple II окончательно сломал представление о компьютере, как об огромном железном монстре, у него был изящный дизайн и дружелюбный интерфейс взаимодействия с пользователем.

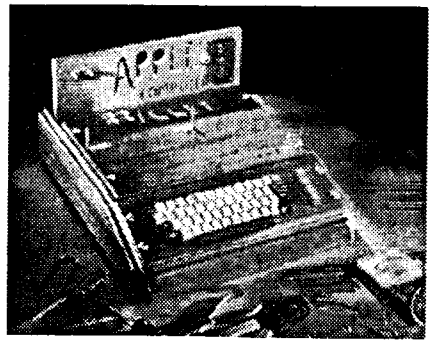


Рис. 2.15. Одни из первых персональных компьютеров — Apple I (1976)

Персональные компьютеры не привлекали крупные компании до 1979 года, когда появился первый процессор электронных таблиц — VisiCalc. Идея VisiCalc

была предложена студентом Гарварда Даном Брисклином, которому пришлось решать сложные финансовые задачи, требующие большого количества вычислений. Со своим другом Бобом Франкстоном они написали VisiCalc для компьютера Apple II. Программа оказалась настолько удобной для финансовых вычислений, что многие компании стали покупать Apple II с VisiCalc для своих сотрудников.



Рис. 2.16. Стив Джобс (справа) и Стив Возняк (слева) (1976)

В 1981 году крупнейшая компьютерная компания IBM представила свой первый персональный компьютер – IBM PC. В течение двух лет было продано более пяти миллионов этих компьютеров. В то же время компания Microsoft начинает выпуск программного обеспечения для IBM PC. Появляются клоны IBM PC, но все они, так или иначе, отражают стандарты, заложенные IBM. Появление клонов IBM PC способствовало росту промышленного производства персональных компьютеров.

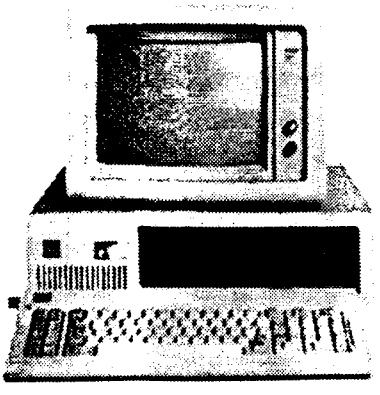


Рис. 2.17. Один из первых персональных компьютеров компании IBM — IBM PC/XT (1983)

В 1984 году компания Apple представила компьютер «Макинтош». Операционная система «Макинтоша» включала в себя графический интерфейс пользователя,

позволявший вводить команды, выбирая их с помощью указателя «мышь». Сами команды были представлены в виде небольших графических изображений — значков. Простота использования в сочетании с большим набором текстовых и графических программ сделала этот компьютер идеальным для небольших офисов, издательств, школ и даже детских садов. С появлением «Макинтоша» персональный компьютер стал еще более доступным. Для работы с ним больше не требовалось никаких специальных навыков, а тем более знания программирования. В 1984 году компания Apple показала на телевидении первый ролик, посвященный рекламе персонального компьютера. Компьютер действительно перестал быть чем-то особенным и превратился в обычный бытовой прибор.

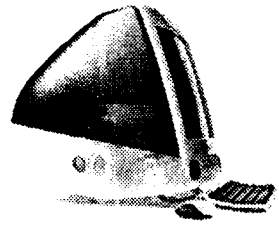


Рис. 2.18. Одна из последних моделей компьютеров Apple — iMac (1999)

На протяжении всего 50 лет компьютеры превратились из неуклюжих диковинных электронных монстров в мощный, гибкий, удобный и доступный инструмент. Компьютеры стали символом прогресса в XX веке. По мере того как человеку понадобится обрабатывать все большее количество информации, будут совершенствоваться и средства ее обработки — компьютеры.

Ключевые понятия

- перфокарта;
- булева алгебра;
- архитектура фон Неймана;
- машина Тьюринга;
- поколения компьютеров;
- язык программирования;
- компьютер;
- интегральная микросхема;
- мэйнфрейм;
- мини-компьютер;
- микропроцессор и микрокомпьютер.

Контрольные вопросы

1. Дайте характеристику основных этапов развития вычислительной техники.
2. Сформулируйте, в чем заключается принципиальное значение вычислительных инструментов, предложенных Паскалем, Бэббиджем.
3. Раскройте содержание термина «булева алгебра».
4. Что такое архитектура фон Неймана?
5. Приведите классификацию компьютеров по функциональному назначению.
6. Представьте классификацию языков программирования.
7. Перечислите основные характеристики персонального компьютера.
8. Охарактеризуйте изменение элементной базы компьютеров при смене их поколений.
9. Какие причины обусловили появление компьютеров четвертого поколения?
10. Какие классы современных персональных компьютеров вы можете перечислить?

Литература

1. Апокин И. А., Майстров Л. Е. История вычислительной техники. М.: Наука,1990.
2. Гутер Р. С., Полунов Ю. Л. От Абака до компьютера. М.: Знание, 1975.
3. Королев Л. Н. Структуры ЭВМ и их математическое обеспечение. М., 1978.

Глава 3. Технические средства обработки информации

Цель данной главы — сформировать у читателя общие представления о тех технических устройствах и аппаратных решениях, которые положены в основу систем обработки информации. В общем плане данный круг вопросов относится к ведению общей информатики и даже технических дисциплин. Однако практический опыт подготовки специалистов по экономике и финансам свидетельствует о том, что современным менеджерам или экономистам приходится принимать решения, касающиеся как программного обеспечения, так и того оборудования, на котором оно должно функционировать. Для этого они должны иметь общее представление об аппаратных средствах, используемых в компьютерных технологиях.

3.1. Понятие вычислительной машины и принципы организации ее работы

Вычислительная машина — это техническое устройство обработки информации. Кроме обработки информации, вычислительная машина выполняет функции ввода, хранения и вывода информации. Общие принципы работы универсальных вычислительных устройств были сформулированы известным математиком Джоном фон Нейманом.

Общая схема работы вычислительной машины, или *компьютера*, может быть представлена следующим образом. Компьютер может работать только по заранее разработанной программе. Программа представляет собой последовательность специальных команд, которые выполняются одна за другой, сначала первая, потом вторая, третья, и т. д., пока не выполнится последняя команда. Последовательность команд может быть изменена при помощи команд, передающих управление на выполнение других команд, не обязательно следующих за текущей командой. Таким образом, некоторые последовательности команд могут выполняться неоднократно в течение нескольких циклов.

Для выполнения программа размещается на устройстве, которое называется памятью, или основной, системной памятью. Каждая команда размещается в отдельной ячейке памяти, имеющей свой адрес, по которому может быть осуществлен прямой непосредственный доступ. Устройство управления, имея адрес ячейки памяти, в которой находится первая команда программы, извлекает эту команду и передает на выполнение в арифметическо-логическое устройство. После выполнения этой команды из следующей ячейки памяти извлекается очередная команда и передается на выполнение. Последовательность команд выполнятся автоматически до выполнения последней.

Набор команд, которые может выполнить арифметическо-логическое устройство, определяется при разработке и совершенствовании этого устройства. В общем случае команды могут задавать выполнение арифметических или логических операций, чтение данных из памяти, запись данных в память, чтение данных из внешних устройств, запись данных на внешние устройства.

В современных компьютерах устройство управления и арифметическо-логическое устройство объединены в одно, которое называется *процессором* (или центральным процессором). В настоящее время процессоры производятся на отдельном кристалле (чипе) по специальным микротехнологиям. Такие устройства называют *центральными микропроцессорами*. Они определяют тип компьютера.

Современные компьютеры являются сложными устройствами, состоящими из многих устройств, работа которых управляется центральным микропроцессором. Выполняющаяся программа может прерываться по специальным сигналам-прерываниям от внешних устройств для выполнения срочных действий.

3.2. Классы вычислительных машин и их основные характеристики

По возможностям, которые предоставляют компьютеры пользователю, их можно разделить на следующие большие группы: мини-компьютеры, мэйнфреймы, суперкомпьютеры.

Суперкомпьютеры обладают самым высоким быстродействием и имеют огромные вычислительные мощности. Производство суперкомпьютеров — это штучное производство, в котором используются самые новейшие достижения во многих отраслях науки и техники. Суперкомпьютеры используются для сложных расчетов в аэродинамике, метеорологии, космических и физических исследованиях, экономике и финансовом управлении.

Мэйнфреймы обладают значительными ресурсами для решения сложных задач в финансовой области, в управлении регионами, отраслями, большими предприятиями, в том числе и предприятиями торговли, в военной области. Мэйнфреймы получили наибольшее развитие в 80-е годы. Однако и сейчас они с успехом выполняют задачи по интеграции больших неоднородных компьютерных комплексов.

Мини-компьютеры используются при управлении предприятиями и организациями. К ним относятся *серверы* старшего уровня, являющиеся центральными компьютерами в компьютерных сетях предприятия, которые выполняют функции управления локальными сетями. В качестве серверов среднего и младшего уровня используются микрокомпьютеры, имеющие меньшие возможности. На рис. 3.1. представлен сервер для рабочих групп.

Микрокомпьютеры — это самые массовые модели вычислительных машин. К ним относятся персональные компьютеры настольного (рис. 3.2) и мобильного исполнения (рис. 3.3). Можно выделить рабочие станции, персональные компьютеры, сетевые компьютеры. *Рабочие станции* используются в офисах, для работы с научными и инженерными приложениями, при моделировании производственных, финансово-экономических процессов, в типографском деле. *Персональные компьютеры* имеют более низкие характеристики и используются для офисных приложений: текстовых процессоров, электронных таблиц, ведения простейших баз данных. *Сетевые* компьютеры используются в локальных сетях как компонент архитектуры клиент—сервер. Сетевые компьютеры могут не иметь достаточных вычислительных мощностей для решения сложных задач. Недостаток мощности восполняется возможностями сетей, которые реализуются при использовании мощных компьютеров-серверов.

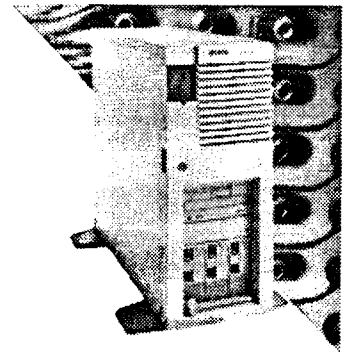


Рис. 3.1. Сервер для рабочих групп



Рис. 3.2. Персональный компьютер настольного исполнения

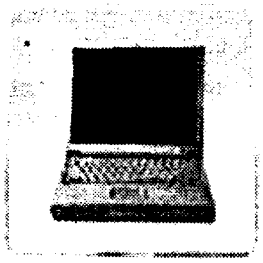


Рис. 3.3. Персональный компьютер в мобильном исполнении

Сервер и персональный компьютер относятся к одному классу микрокомпьютеров. Отличие состоит в надежности работы. От работы сервера зависит работа целого предприятия или подразделения. Поэтому сервер должен обладать достаточной надежностью и устойчивостью к возможным сбоям системы. Для обеспечения надежности работы сервера могут использоваться такие средства, как резервные источники питания, сдвоенная шина, сдвоенные контроллеры, зеркалирование (использование RAID-систем), использование алгоритмов для быстрого и полного восстановления данных.

Сервер не является лишь усиленным (с дублированием основных элементов) вариантом персонального компьютера. Серверные решения предполагают использование новейших достижений в области компьютерной техники, которые позднее находят свое применение в персональных компьютерах.

3.3. Состав аппаратного обеспечения персонального компьютера, характеристика основных периферийных устройств

3.3.1. Системный блок

Системный блок состоит из корпуса с блоком питания и системной платы.

Системная плата

Системная (материнская) плата является основной частью компьютера, при помощи которой части компьютера объединяются в одно целое. Системная плата представляет собой большую печатную плату, на которой располагаются основные электронные элементы компьютера:

- системная и локальные шины;
- микропроцессор;
- оперативная память;
- дополнительные микросхемы;
- разъемы (слоты) для дополнительных устройств.

Материнские платы унифицированы по типоразмерам (форм-факторам). На данный момент преобладают типоразмеры AT, ATX, LPX и NLX.

Форм-фактор AT используется в модификации Baby AT. Это уменьшенный вариант AT. Размеры 8,5"×13". Почти все платы имеют последовательные и параллельные порты, присоединяемые к материнской плате через соединительные планки. Разъем клавиатуры впаян на задней части платы. Гнездо под процессор устанавливается на передней стороне платы.

Для снижения стоимости компьютера был разработан форм-фактор LPX. Карты расширения вставляются в специальную стойку и располагаются параллельно материнской плате. Высота корпуса уменьшается. Но максимальное количество подключаемых карт не превышает 2-3. Размер остается 9"×13", для Mini LPX – 8"×10".

Спецификация ATX, предложенная Intel еще в 1995 году, нацелена на исправление недостатков, которые выявились со временем у форм-фактора AT. Плата Baby AT повернута на 90°. Разъемы портов ввода/вывода расположены на плате. Добавлены порты PS/2, USB. Количество кабелей в комплекте снижено. Удобный доступ к слотам памяти. Разъемы контроллеров FDD и IDE (флоппи-дисководов, дисководов жестких дисков и некоторых других устройств) передвинуты ближе к подсоединяемым устройствам. Процессор и слоты для плат расширения разнесены. Это позволяет устанавливать в слоты расширения полноразмерные платы – процессор им не мешает. Добавлена возможность управления блоком питания для материнской платы. Размер 12"×9,6", для MiniATX – 11,2"×8,2". В декабре 1997 года была представлена спецификация формата microATX, модификация ATX-платы, рассчитанная на 4 слота для плат расширения (в ATX-плате используется 7 слотов расширения). Размер составляет 9,6"×9,6".

Как развитие идеи LPX, учитывающее появление новых технологий, в 1997 году появилась спецификация форм-фактора NLX, который применяется в низкопрофильных корпусах. При разработке учитывалось появление AGP и модулей DIMM, интеграция аудио- и видео компонентов на материнской плате. Платы расширения устанавливаются в специальную стойку. Материнская плата может быть легко отсоединена от стойки и выдвинута из корпуса, например, для замены процессора или памяти. Размеры колеблются от 8"×9,6" до 9"×13,6".

В 1998 году была предложена спецификация WTX для применения в рабочих станциях и серверах, где на передний план выходят обеспечение нормального охлаждения, размещение больших объемов памяти, удобная поддержка многопроцессорных конфигураций, большая мощность блока питания, размещение большего количество портов контроллеров накопителей данных и портов ввода/вывода. Размеры платы 14"×16,75".

Системная и локальные шины

Системная шина предназначена для передачи информации между компонентами компьютерной системы. Шины бывают синхронными, когда данные передаются в соответствии с тактовой частотой, и асинхронными, когда передача данных осу-

ществляется в произвольные моменты времени. В современных компьютерах применяются шины EISA, PCI, PCMCIA и AGP.

ISA

Шина ISA (Industry Standard Architecture) использовалась в компьютерах, использующих еще микропроцессор 80286. Это 16-разрядная шина с 24 адресными линиями, с 16 линиями аппаратных прерываний и с 8 каналами DMA (Direct Memory Access) прямого доступа к памяти. Шина ISA работает асинхронно на частоте 8 МГц. Скорость обмена данными – около 5 Мбит/с.

EISA

Шина EISA (Extended Industry Standard Architecture) является 32-разрядным расширением шины ISA, которая поддерживает 32-разрядную адресацию памяти и передачу данных, в том числе и в режиме DMA. Шина поддерживает арбитраж DMA (когда каналом пользуется несколько устройств), автоматическую конфигурацию системы и плат расширения. Шина работает на частоте 8 МГц.

PCI

Шина PCI (Peripherial Component Interconnect) является процессорно-независимой, так называемой мезонин-шиной. На рис. 3.4 изображены PCI-разъемы на системной плате. Она может работать параллельно с шиной процессора, то есть обмен данными процессор – память и, например, видеоадаптер – память может осуществляться параллельно. Шина PCI является синхронной 32-разрядной или 64-разрядной шиной, работающей на частоте 33 или 66 МГц. Максимальная скорость обмена данными может достигать 528 Мбит/с для реализации шины 64 бит/66 МГц. Шина поддерживает несколько арбитров. При передаче данных поддерживается кэширование и блочная передача. Шина PCI поддерживает автоматическое определение и конфигурирование плат расширения.

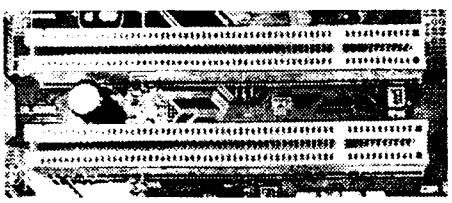


Рис. 3.4. Два слота PCI на системной плате

PCMCIA

Стандарт PCMCIA используется в портативных компьютерах. Стандарт поддерживается Ассоциацией PCMCIA (Personal Computer Memory Card International Association), объединяющей компании, разрабатывающие периферийные устройства для портативных компьютеров. Второе название – PC Card. На шине используется 16 разрядов под данные и 26 разрядов под адрес. Подключение и отключение PC Card может происходить при работающем компьютере. Спецификация PCMCIA определяет типы габаритных размеров для PC Card (Type I, Type II и

Type III), различающиеся по толщине. Размер PC Card не превышает размеры обычной кредитной карточки. Спецификация CardBus является 32-разрядным расширением шины PCI для устройств PC Card.

Шина AGP

Шина AGP (Accelerated Graphic Port) разработана для увеличения быстродействия при обмене между видеоадаптером и оперативной памятью. В переводе эта локальная шина называется ускоренным графическим портом. Порт AGP необходим прежде всего для работы с трехмерной графикой, где необходимо оперировать с большими объемами данных Z-буфера и текстур. Объем этой памяти напрямую определяет качество 3D-изображения и поддерживаемые разрешения. Для хранения текстур используется основная память, а на плате 3D-ускорителя расположена только память кадрового буфера и Z-буфера. Благодаря переносу потока графической информации с шины PCI на шину AGP освободится полоса пропускания шины PCI, которая может быть использована другими устройствами. Порт AGP работает на частоте 66 МГц. Предусмотрены три режима передачи данных — AGP 1x со скоростью обмена 266 Мбит/с, AGP 2x со скоростью обмена 532 Мбит/с и AGP 4x со скоростью обмена 1064 Мбит/с. Преимущества AGP могут быть использованы полностью только в случае, когда плата ускорителя и используемое программное обеспечение поддерживают режим доступа к памяти DIME (Direct Memory Execute).

Микропроцессоры

Микропроцессор представляет собой сверхбольшую интегральную схему, реализованную на одном полупроводниковом кристалле. Это программно управляемое устройство обработки информации. Иначе его называют центральным процессором — Central Processing Unit (CPU). Микропроцессор получает информацию и команды по ее обработке и выдает обработанную информацию. Команды, которые может исполнять микропроцессор, называют инструкциями. В зависимости от типа исполняемых инструкций различают CISC (Complex Instruction Set Computer) и RISC (Reduce Instruction Set Computer) микропроцессоры. Первые микропроцессоры были CISC-процессорами. Инструкции набора X86 имеют длину от 8 до 120 бит. RISC-инструкции имеют одинаковую длину, поэтому они проще, быстрее выполняются. В современных микропроцессорах используются RISC-инструкции.

Первый микропроцессор появился в 1971 году. Это был *4-разрядный* Intel 4004, разработанный с ориентацией на требования изготовителей калькуляторов. (*Разрядность* микропроцессора показывает, сколько двоичных битов информации обрабатывается за один такт.) Он имел невысокое быстродействие и ограниченные возможности адресации памяти. Быстродействие микропроцессора определяется *тактовой частотой*. Измеряется в герцах. Intel 4004 имел тактовую частоту 750 КГц.

В 1972 году фирма Intel выпустила первый *8-разрядный* микропроцессор 8008.

Первые *16-разрядные* микропроцессоры появились в 1977 году. В 1978 году фирма Intel объявила о начале производства микропроцессора 8086, который позже был выбран фирмой IBM в качестве центрального микропроцессора персонального компьютера IBM PC XT (Personal Computer eXtended Technology). PC XT имел огромный успех и положил начало эре персональных компьютеров.

С 1982 года стал выпускаться микропроцессор 80286, который использовался в компьютерах IBM PC AT (Advanced Technology). Этот микропроцессор изготовлялся по 1,5-микрометровой технологии. Тактовая частота от 6 до 12 МГц.

В октябре 1985 года фирма Intel анонсировала первый *32-разрядный* микропроцессор i80386. Этот микропроцессор имел полностью 32-разрядную архитектуру (32-разрядные регистры и 32-разрядная внешняя шина данных) и работал на тактовой частоте 16 МГц. Intel выпустила несколько моделей этого поколения микропроцессоров: 80386DX, упрощенный 80386SX, 80386LX специально для мобильных компьютеров. Тактовая частота от 16 до 33 МГц. Микропроцессоры изготавливались сначала по 1,5-микрометровой, а затем по 1-микрометровой технологии. Этот показатель характеризует размеры и концентрацию элементов микропроцессора на единице площади. Микропроцессор имел завершенную систему поддержки многозадачного режима, мог оперировать оперативной памятью до 64 Мбайт.

Осенью 1989 года Intel объявила о новом 32-разрядном микропроцессоре *i486*. Благодаря использованию конвейерной архитектуры, присущей RISC-процессорам, удалось увеличить производительность обычных 32-разрядных систем в четыре раза. Использование встроенной *кэш-памяти* (8 Кбайт) ускоряет выполнение команд за счет промежуточного хранения часто используемых команд и данных. Кэш-память имеет наименьшее время доступа по сравнению с другими видами памяти.

Развитие микропроцессоров идет не только по пути увеличения разрядности и тактовой частоты, но и по пути улучшения декодирования команд. Операция сложения двух чисел выполняется 386 микропроцессором за 6 тактов, 486 — за 2 такта. Микропроцессоры следующих поколений за один такт выполняют не менее одного сложения.

В марте 1992 года фирма Intel объявила о создании нового микропроцессора 486, названного i486DX2. Скорость работы внутренних блоков микропроцессора больше скорости работы остальных частей компьютерной системы в два раза. Технология *умножения тактовой частоты*, и не только в два раза, нашла свое применение и в следующих моделях микропроцессоров. Последние модели микропроцессоров выпускались уже по 0,8-микрометровой технологии.

Микропроцессоры могут работать только с целыми числами. Для удовлетворения всех требований, предъявляемых к компьютерам, дополнительно устанавливаются *математические сопроцессоры*. Позже, начиная с модели Pentium, математический сопроцессор встраивается в основной микропроцессор.

Pentium

В марте 1993 года фирма Intel объявила о начале промышленных поставок 32-разрядных микропроцессоров нового поколения, названных *Pentium*. Новый микропроцессор имеет 32-разрядную адресную и 64-разрядную внешнюю шины дан-

ных. Микропроцессор выполнен по 0,8-микрометровой технологии, работает на тактовых частотах 60 и 66 МГц. Через год появилась следующая модель микропроцессора, выполенная по 0,5- (позднее по 0,35) микрометровой технологии. Впервые был применен раздельный 16-килобайтный кэш: 8 Кбайт для инструкций и 8 Кбайт для данных. Тактовая частота была в пределах 75—200 МГц, а тактовая частота системной шины 50—66 МГц.

Pentium MMX — это версия Pentium с дополнительными мультимедиа-инструкциями (добавлено 57 новых инструкций). Основа MMX — *технология обработки множественных данных одной инструкцией* (Single Instruction Multiple Data, SIMD). Кэш-память увеличена до 32 Кбайт. Тактовая частота — 166—233 МГц. Частота системной шины — 66 МГц.

Pentium Pro

Разработка Pentium Pro началась в 1991 году. Промышленный выпуск начался в ноябре 1995 года. Это RISC-процессор, разработанный для 32-разрядных операционных систем. Впервые в микропроцессоре вместе с кэш-памятью L1 (здесь объемом 16 Кбайт) стали применять *кэш-память второго уровня* (L2), объединенную в одном корпусе и оперирующую на частоте микропроцессора. Выпускался сначала по 0,5-, позднее по 0,35-микрометровой технологии. Кэш уровня L1 объемом 16 Кбайт. Кэш уровня L2 имел объем 256, 512, 1024 и 2048 Кбайт. Тактовая частота от 150 до 200 МГц. Четырехканальная параллельная обработка данных. Частота системной шины 60—66 МГц. Pentium Pro поддерживал все инструкции процессора Pentium, кроме MMX, а также ряд новых инструкций. Введена архитектура *двойной независимой шины*, обеспечивающая параллельный обмен данными ядро микропроцессора — кэш L2 и ядро микропроцессора — оперативную память. Микропроцессор устанавливается в разъем Socket 8, позволяющий поддерживать до четырех микропроцессоров в симметричной мультипроцессорной системе.

Pentium II

Первая модель микропроцессора Pentium II впервые появилась в мае 1997 года. Под этим общим именем выпускались микропроцессоры, предназначенные для разных сегментов рынка: для недорогих low-end-компьютеров, для массового рынка ПК среднего уровня, для высокопроизводительных серверов и рабочих станций.

Klamath — первый процессор линейки Pentium II, изготовленный по 0,35-микрометровой технологии. Диапазон тактовых частот 233—300 МГц. Частота системной шины — 66 МГц, кэш-память уровня L2 — 256 или 512 Кбайт, которая для уменьшения стоимости продукта размещена в специальном модуле SECC (Single Edge Contact Cartridge) вместе с микропроцессором. На системной плате устанавливается в разъем Slot1, поддерживающий двухпроцессорную систему. Кэш работает на половине частоты процессора, что является шагом назад по сравнению с Pentium Pro, у которого кэш работает на полной частоте микропроцессора. Кэш первого уровня 32 Кбайт. Дополнен MMX-блоком. Таким образом, это быстрый Pentium Pro с MMX-поддержкой и с урезанными возможностями создания мультипроцессорных систем.

Deschutes — дальнейшее развитие линейки Pentium II. Первая модель появилась в январе 1998 года. Технология изготовления усовершенствована с 0,35 до 0,25 мкм. Тактовая частота 333 МГц, частота системной шины 66 МГц. В апреле появилась модель, которая работает на частоте системной шины 100 МГц. Тактовая частота — 350—450 МГц. Микропроцессор и кэш L2 располагаются в картридже SECC2 (более совершенная система охлаждения). Устанавливается в Slot 1.

Микропроцессор *Celeron* впервые появился в апреле 1998 года. Он предназначался для рынка недорогих компьютеров. Celeron выпускался как с кэшем второго уровня (см. рис. 3.5), так и без. Выпускается в вариантах для Socket 370, Slot 1. Эти микропроцессоры хорошо «разгоняются». Процессор Celeron 300, например, хорошо работает на частоте 464 МГц.

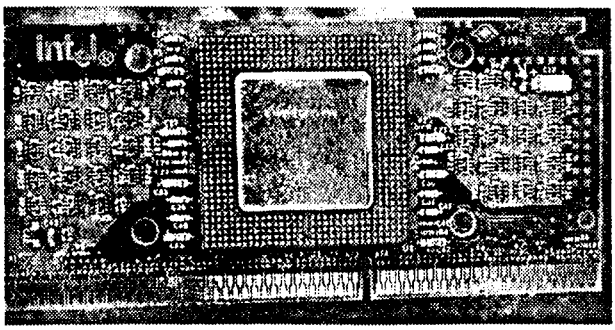


Рис. 3.5. Процессор Celeron 300A

Covington — первый процессор линейки Celeron. Построен на ядре Deschutes и выпускался по 0,25-микрометровой технологии. Тактовая частота 266—300 МГц, частота системной шины 66 МГц, кэш L1 — 32 Кбайт (по 16 Кбайт для данных и инструкций). Для уменьшения себестоимости выпускался без кэш-памяти второго уровня. Размещается в картридже SEPP (Single Edge Processor Package), где нет термопластины и защитной крышки. Устанавливается в Slot 1.

Mendocino — является развитием линейки Celeron. Появился в августе 1998 года. В отличие от своего предшественника, имеет кэш-память второго уровня объемом 128 Кбайт, интегрированную на одном кристалле с ядром и работающую на частоте процессора. Тактовая частота — 300—533 МГц. Частота системной шины — 66 МГц. Технологический процесс — 0,25 мкм, 0,22 мкм. Благодаря тому что кэш L2 оперирует на частоте процессора, имеет весьма неплохую производительность.

С 1999 года микропроцессоры Celeron стали выпускаться в пластиковом корпусе PPGA (Plastic Pin Grid Array), который устанавливается в Socket 370. Это дешевле, чем процессор в полупустом картридже SEPP.

Xeon — микропроцессор Pentium II, разработанный для замены Pentium Pro (см. рис. 3.6). Кэш-память второго уровня работает на частоте процессора. Это первый процессор для Slot 2, и он предназначен в первую очередь для мощных серверов и рабочих станций. Способен работать в мультипроцессорных конфигурациях. Построен на ядре Deschutes и выпускается, как и собственный кэш, по 0,25-мик-

рометровой технологии. Кэш L2 имеет объем 512, 1024, 2048 Кбайт и работает на полной частоте микропроцессора.

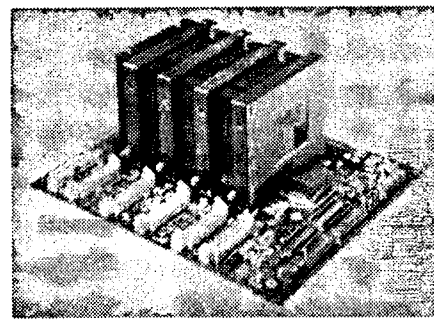


Рис. 3.6. Четыре процессора Xeon на системной плате

Pentium III

Katmai — микропроцессор Pentium III. Развитие Deschutes. Здесь расширен набор MMX (MMX2), в основе которого лежит *технология SSE* (Streaming SIMD Extentions), где технология SIMD расширена на числа с плавающей запятой. Добавлены новые 128-разрядные регистры. Каждый регистр может обрабатывать четыре числа с плавающей запятой. Усовершенствована технология поточного доступа к памяти, улучшающая взаимодействие между кэш-памятью L2 и оперативной памятью. Дополнительные инструкции называются *инструкциями KNI* (Katmai New Instruction). Внедрение KNI предназначено для ускорения работы графических приложений и 3D-игр. Введена уникальная нумерация процессоров. Тактовая частота 450–600 МГц. Кэш-память L2 512 Кбайт размещена на процессорной плате. Частота системной шины 100 или 133 МГц.

Coppermine — микропроцессор Pentium III с ядром Katmai, сделанный на базе 0,18 мкм техпроцесса, с интегрированной кэш-памятью L2 256 Кбайт. Тактовая частота — от 533 МГц и выше. Последний Slot-1-процессор.

Coppermine 128 K — микропроцессор Celeron с процессорным ядром Coppermine с урезанной до 128 Кбайт кэш-памятью L2. Впервые для Celeron будет поддержка SSE.

Cascades — серверный вариант Coppermine. На чипе содержится кэш-память L2 256 Кбайт (позднее до 2 Мбайт), тактовая частота от 600 МГц, частота системной шины — 133 МГц. Форм-фактор — Slot-2.

Tanner — Pentium III Xeon. Предназначен для серверов. Тактовая частота от 500 МГц, частота системной шины 100 МГц. Кэш-память L2, работающая на частоте процессора, объемом 512, 1024 и 2048 Кбайт. Поддержка MMX и SSE. Кэш-память L1 — 32 Кбайт. Слот 2.

Willamette — последний IA-32-процессор Intel для обычных PC. Использует новую системную шину с результирующей частотой 400 МГц. Кэш-память L1 — 256 Кбайт, L2 — 1 Мбайт. 0,18-микрометровый технологический процесс с последующим переходом на 0,13 мкм. Тактовая частота — более 1 ГГц.

Foster — серверный вариант Willamette. Частота системной шины — 400 МГц. Значительно увеличенная кэш-память L1 и L2. Тактовая частота — выше 1 ГГц. Форм-фактор — Slot-M. Последний IA-32-процессор от Intel, своеобразное переходное звено к IA-64.

Merced — первый процессор архитектуры IA-64, аппаратно совместим с архитектурой IA-32, включает трехуровневую кэш-память 2—4 Мбайт. Технология изготовления 0,18 мкм, тактовая частота, начиная с 800 МГц, частота системной шины — 266 МГц. Физический интерфейс: Slot M.

McKinley — второе поколение процессоров архитектуры IA-64, тактовая частота — начиная с 1000 МГц. Увеличенный объем кэш-памяти L2. 0,18-микрометровая технология с последующим переходом на 0,13-микрометровую медную технологию. Результирующая частота системной шины — 400 МГц. Физический интерфейс — Slot M.

В истории развития микропроцессоров конкуренцию Intel на рынке персональных компьютеров составляли многие производители. Коротко остановимся на некоторых.

AMD

K5 — первый процессор AMD, который всерьез предназначался для конкуренции с Pentium. Тактовая частота от 75 до 166 МГц по PR-рейтингу (тактовая частота по PR-рейтингу использовалась для сравнения производительности микропроцессоров конкурентов, фактические значения тактовых частот которых не совпадали; производительность процессора сравнивается со скоростью процессора Pentium, на которой ему пришлось бы работать для достижения той же производительности). Частота системной шины составляла от 50 до 66 МГц. Кэш-память L1 — 24 Кбайт. Кэш-память L2 для уменьшения стоимости размещалась не вместе с микропроцессором, как у Intel, а на материнской плате; работает на частоте системной шины. Технологический процесс 0,6 мкм и 0,35 мкм.

K6 — начал поставляться со 2 апреля 1997 года, на месяц раньше выхода Pentium II, производился на базе 0,35 (позднее 0,25) мкм технологического процесса. Процессор (рис. 3.7) работает на частоте от 166 до 233 МГц. Кэш-память L1 увеличена до 64 Кбайт. Микропроцессор имеет модуль MMX. Микропроцессор имеет внутреннюю RISC-подобную организацию. Все последующие процессоры унаследовали это свойство от AMD.



Рис. 3.7. Процессор AMD K6

- K6-II — следующее поколение K6. Вышел в мае 1998 года. Тактовая частота 266—500 МГц. Кэш-память L2 расположена на материнской плате, работает на частоте

системной шины 100 МГц, имеет объем до 2 Мбайт. Микропроцессор поддерживает дополнительный набор из 21 инструкций *3DNow!*, использование которых обеспечивает повышение производительности 3D-приложений. MMX и 3DNow! несовместимы, но K6-II поддерживает оба набора. Этот микропроцессор может исполнять инструкции либо набора MMX, либо набора 3DNow!. Последние модели микропроцессора имеют новое ядро, поддерживающее модифицированный метод работы с кэш-памятью.

Sharptooth (K6-III) — первый процессор от AMD, имеющий кэш-память L2 объемом 256 Кбайт на ядре. Кэш-память работает на частоте процессора. Кэш-память L1 имеет объем 64 Кбайт (по 32 Кбайт для инструкций и данных), кэш-память L3 находится на материнской плате и может иметь объем от 512 Кбайт до 2 Мбайт, работая на частоте системной шины. Тактовая частота 400 и 450 МГц. Технологический процесс — 0,25 мкм.

K7 (Athlon) — микропроцессор, имеющий кэш-память L1 128 Кбайт (по 64 Кбайт для инструкций и данных), работающую на 1/2 или 2/5 частоты процессора. На рис. 3.8 блок микропроцессора установлен на системной плате. Системная шина та же, что и для процессоров Alpha. Скорость системной шины — 200 МГц.

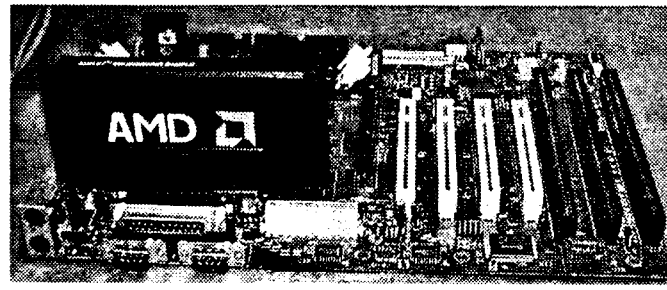


Рис. 3.8. Материнская плата с установленным процессором Athlon

Thunderbird — Socket-вариант Athlon, работающий на частоте 1,1 ГГц. Кэш-память L2 на чипе — 512 Кбайт, работает на полной частоте процессора.

Spitfire — недорогая (за счет уменьшения кэш-памяти L2) версия Athlon. Кэш-память L2 работает на полной частоте процессора.

Mustang — серверный вариант Athlon. Кэш-память L2 объемом 1–2 Мбайт интегрирована в чип. Процессор рассчитан на использование системной шины 266 МГц и DDR SDRAM-памяти.

SledgeHammer — первый 64-битный процессор AMD. Микропроцессор оптимизирован на исполнение 32-битных инструкций. Предполагается, что к этому времени 64-разрядные приложения не занимают значительную часть программного рынка. Тактовая частота — 1,5 ГГц и выше.

Cyrix

6x86 — или M1. PR-рейтинг 6x86 составлял от 120 до 200 МГц. Кэш первого уровня — 16 Кбайт (единый). Частота системной шины — от 50 до 75 МГц.

6x86MX имеет увеличенную кэш-память L1 объемом 64 Кбайт. Добавлена поддержка MMX. Частота системной шины от 60 до 75 МГц. PR-рейтинг от 166 до 266 МГц.

MII — микропроцессор с кэш-памятью L2 на материнской плате объемом от 512 Кбайт до 2 Мбайт. Технологический процесс 0,25 мкм. PR-рейтинг 300—433 МГц.

Системная память

Память *динамического* типа — память с произвольной выборкой (Dynamic Random Access Memory, DRAM). Каждый бит такой памяти представляется в виде наличия (или отсутствия) заряда на конденсаторе, образованном в структуре полупроводникового кристалла. Другой, более дорогой тип памяти — *статический* (Static RAM, SRAM) в качестве элементарной ячейки использует так называемый статический триггер (схема которого состоит из нескольких транзисторов). Статический тип памяти обладает более высоким быстродействием.

Память по способу доступа к данным может быть как асинхронной, так и синхронной. Асинхронным называется доступ к данным, который можно осуществлять в произвольный момент времени. Синхронная память обеспечивает доступ к данным не в произвольные моменты времени, а синхронно с тактовыми импульсами.

Микросхемы динамической памяти исполняются в разных корпусах: SIMM (Single In line Memory Module), DIMM (Dual In line Memory Module). Контакты у DIMM-модулей (рис. 3.9) на разных сторонах платы электрически не связаны между собой, как у SIMM-модулей.



Рис. 3.9. Модуль памяти DIMM 64 Мбайт

EDRAM

Память Enhanced DRAM основана на интеграции небольшого объема памяти SRAM и DRAM, где первая, более быстрая, память используется в качестве кэша, где размещаются более важные, часто используемые данные.

FPM DRAM

Fast Page Mode DRAM имеет страничную организацию ячеек памяти, ускоряющую, по сравнению с произвольной выборкой, доступ к расположенным последовательно в пределах страницы данным.

EDORAM

В Extended Data Out DRAM используются специальные регистры для хранения выходных данных на выходе модуля памяти в течении некоторого дополнительного времени. При этом их можно считывать на более высокой частоте.

SDRAM

Synchronous (синхронная) DRAM синхронизирована с системным таймером, управляющим центральным процессором. Синхронизация делает данные доступными во время каждого такта, в то время как у EDO RAM данные доступны один раз за два такта, а у FPM — один раз за три такта. Технология SDRAM позволяет применять банки памяти, используя их чередование (interleaving) для ускорения работы памяти.

Спецификация PC100

Спецификация PC100 определяет набор требований к модулям SDRAM для работы на частоте 100 МГц.

SDRAM II (DDR)

Synchronous DRAM II, или DDR (Double Data Rate — удвоенная скорость передачи данных) — следующее поколение SDRAM. В DDR используется более точная внутренняя синхронизация, отсутствующая в SDRAM, которая фактически увеличивает скорость доступа вдвое за счет возможности передачи данных на обеих границах сигнала тактовой частоты.

SLDRAM (SyncLink)

SyncLink DRAM, продукт DRAM-консорциума, является ближайшим конкурентом Rambus. Этот консорциум объединяет двенадцать производителей DRAM. SLDRAM продолжает развивать технологии SDRAM, расширяя четырехбанковую архитектуру модуля до шестнадцати банков. Кроме того, добавляется новый интерфейс и управляющая логика. SLDRAM передает данные по каждому такту.

RDRAM

Rambus DRAM представляет собой интегрированную на системном уровне технологию, разработанную Rambus Inc., включающую многофункциональный протокол обмена данными между микросхемами, работающий на высокой частоте до 600 МГц. RDRAM использует 8-битовый интерфейс, в то время как EDO RAM и SDRAM используют 4-битовый, 8-битовый и 16-битовый интерфейсы. Допускается одновременная обработка до 4 запросов. На рис. 3.10 приведен 184 pin модуль Rambus. Технология Direct RDRAM состоит из трех функциональных частей: Rambus Interface, Rambus Channel и собственно чипы RDRAM. RAMBUS запатентована 11 крупнейшими производителями DRAM.

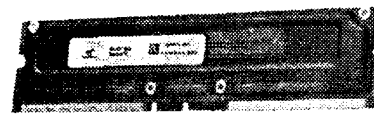


Рис. 3.10. 184 pin модуль Rambus

Concurrent Rambus

Concurrent Rambus использует улучшенный протокол. По сравнению с RDRAM, применен новый синхронный параллельный протокол для чередующихся или перекрывающихся данных, который повышает эффективность на 80%.

Direct Rambus DRAM

Это совместная разработка Intel и Rambus Inc. по доработке существующей технологии Rambus: в ней было обеспечено расширение интерфейса до 16 бит, переход на частоту 800 МГц и внедрение более эффективного протокола.

PC133

Появление PC133 было неизбежно — когда Intel, вдохновитель PC66 и PC100, ушел и место осталось незанятым, просто не мог не появиться кто-то, кто продолжил бы логичный эволюционный процесс. И этим кем-то оказался второй крупнейший производитель чипсетов для компьютеров — VIA Technologies. К тому же индустрия, как правило, без особой радости встречает появление продуктов, за производство которых следует делать лицензионные отчисления, что мы имеем в случае с Direct RDRAM.

VCM133

Другой способ увеличить эффективность работы SDRAM — это использование технологии Virtual Channel Memory. Ее разработчик — NEC. Она обеспечивает достаточно заметный прирост в быстродействии, практически никак не сказываясь на стоимости чипа DRAM. Каждому устройству (из тех, которые конкурируют между собой за доступ к памяти) назначается свой высокоскоростной виртуальный канал, учитывающий специфические характеристики его запросов. Внешние и внутренние операции абсолютно независимы друг от друга и могут исполняться параллельно. По выводам чипы VCM полностью аналогичны обычным чипам SDRAM. VCM полностью независима от типа памяти и с легкостью может быть в дальнейшем встроена, например, в DDR SDRAM.

Видеопамять

В видеопамяти используется динамическая оперативная память, работа которой имеет ряд особенностей: доступ осуществляется достаточно крупными блоками, перезаписывание данных без прерывания процедуры считывания. Для этого используется двухпортовая RAM. Это VRAM (Video RAM) и WRAM (Window RAM). Используются и другие виды памяти.

Вспомогательные устройства

BIOS (Basic Input/Output System)[1], выполненная на базе специальной микросхемы (рис. 3.11), содержит набор программ ввода/вывода, благодаря которым операционная система и прикладные программы могут работать с устройствами компьютера на физическом уровне. В набор программ BIOS входят программа тестирования компьютера и его устройств, которая запускается при включении компьютера, и программа setup, которая позволяет изменять параметры, определяющие конфигурацию компьютерной системы и необходимые для работы программ BIOS. Конфигурационные параметры хранятся в специальной микросхеме, которая в настоящее время исполняется на основе технологии флэш-памяти.

[1] Некоторые вопросы, связанные с назначением и функциями BIOS, рассматриваются в 4.3.1.



Рис. 3.11. Микросхема AMI с BIOS

Большинство устойств компьютера работает в синхронном режиме. *Тактовый генератор* служит для выработки специальных импульсов для синхронизации, например, работы микропроцессора и системной шины.

3.3.2. Устройства хранения информации

Такие устройства называются накопителями. В основе их работы лежат разные принципы (в основном это магнитные или оптические устройства), но используются они для одной цели — для хранения информации для последующего многократного использования. Этот вид памяти, в отличие от оперативной памяти, является энергонезависимым. Объем носителей, используемых в этих устройствах, значительно превосходит объем оперативной памяти. Стоимость хранения единицы информации существенно ниже. Накопители бывают внешними (собственный корпус и источник питания) и внутренними (встраиваются в корпус компьютера), со сменными и несменными носителями, с носителями разной формы (диски, ленты). Накопители имеют разные характеристики: максимально возможный объем хранимой информации, время доступа. Немаловажное значение имеет стоимость хранения информации. Для интеграции накопителей в компьютер разработаны специальные интерфейсы.

Интерфейсы накопителей

В настоящее время преобладают два интерфейса: IDE и SCSI. Большое распространение получает интерфейс USB.

SCSI

Интерфейс SCSI (Small Computer System Interface) был разработан еще в 1970 году. К шине можно подключить до восьми устройств, включая основной контроллер SCSI. Контроллер SCSI имеет собственную BIOS, которая управляет шиной SCSI, освобождая центральный процессор. Шина SCSI — 8-разрядная, тактовая частота — 5 МГц, имеет восемь линий данных, линию четности, девять управляющих линий. Максимальная скорость передачи данных не превышает 5 Мбит/с. Интерфейс SCSI имеет большую чувствительность к качеству изготовления кабелей.

Интерфейс SCSI-2 (Fast SCSI) — развитие стандарта, в котором тактовая частота и скорость передачи данных увеличены в два раза. Добавлены и стандартизованы команды доступа к периферийным устройствам.

В стандарт *SCSI-2* добавлена спецификация *Wide* (широкая), в которой количество информационных линий увеличено до 24. К шине можно подключать как 8-разрядные, так и 16-разрядные устройства.

В модификации *Ultra SCSI* тактовая частота – 20 МГц (другие названия *SCSI-3, SCSI 20*). Добавлены команды для некоторых графических устройств. *Ultra Wide SCSI* обеспечивает 16-разрядную передачу данных со скоростью до 40 Мбит/с.

В *Ultra 2 Wide SCSI* тактовая частота увеличена до 40 МГц, скорость передачи данных – 40 Мбит/с. Существует *Wide*-спецификация этого стандарта, по которой поддерживается до 15 устройств.

IDE

Интерфейс IDE (Integrated Drive Electronics) был предложен в 1988 году как недорогая альтернатива используемым в то время интерфейсам. Отличительной чертой IDE является реализация функций контроллера в электронной части устройства. Второе название интерфейса – ATA (AT Attachment). В качестве подключаемых устройств могли быть только накопители на несъемных жестких дисках. Интерфейс поддерживает PIO (Programmed Input/Output), в соответствии с которым обмен данными осуществляется через центральный процессор. Второй способ обмена – режим DMA (Direct Memory Access), когда обмен данными с оперативной памятью осуществляется непосредственно, без участия центрального процессора. IDE одновременно может поддерживать до двух устройств, режимы PIO Mode 1 и 2, DMA. Максимальная скорость передачи данных 8,3 Мбит/с.

Развитием стандарта стал *интерфейс EIDE* (Enhanced IDE), который поддерживает устройства объемом свыше 504 Мбайт, и не только устройства с жесткими дисками (такая спецификация интерфейса называется ATAPI – ATA Packed Interface). Возможно подключение до четырех устройств. В системе можно устанавливать несколько контроллеров EIDE. Интерфейс поддерживает режимы PIO Mode 3 и 4, DMA Single Word Mode 2 и Multi Word Mode 1 и 2. Существенное увеличение пропускной способности шины дает поддержка режима Bus Mastering DMA. Максимальная скорость передачи данных – до 16,7 Мбит/с.

Интерфейс ATA-3 разработан для повышения надежности передачи данных и повышения ее производительности. Стандарт поддерживает технологию предупреждения отказов жестких дисков SMART (Self-Monitoring Analysis and Reporting Technology).

Интерфейс Ultra ATA поддерживает протокол UltraDMA, по которому за один такт передается в два раза больше информации, чем в предыдущих версиях.

Выпускаются накопители, использующие, кроме рассмотренных стандартов, стандарты USB, PCMCIA, FireWire и др.

Накопители на лентах

Накопители на магнитных лентах называются *стримерами*. Стримеры используются, когда необходимо записать большие объемы информации при создании архивных копий. Современные стримеры используют специальные кассеты (кар-

триджи) с магнитной лентой. Стримеры, как правило, имеют собственные средства сжатия данных. Стримеры имеют разные стандарты, определяющие интерфейс с компьютером, формат магнитной ленты, методы кодирования и сжатия.

QUC-стримеры (Quarter Inch Cartridge) имеют толщину картриджа 1/4". Емкость картриджей различна — от 250 Мбайт до 1,3 Гбайт. *Travan*-стримеры используют магнитные ленты шириной 0,315". Емкость картриджей — от 400 Мбайт до 4 Гбайт. В *DAT*-стримерах (Digital Audio Tape) используется технология спирального сканирования, позволяющая увеличить плотность записи. Емкость картриджа достигает 8 Гбайт. *DLT*-стримеры (Digital Linear Tape) имеют высокую надежность в эксплуатации, позволяют записывать информацию с высокой скоростью. Емкость картриджа до 35 Гбайт.

Накопители на дисках

Отличительной особенностью таких устройств является использование в качестве носителей информации круглых дисков разного диаметра, отличающихся форм-фактором. Выпускаются носители с форм-фактором (размером) 1,8", 2,5", 3,5", 5,25".

Накопители на жестких несъемных дисках

Эти накопители называются винчестерами. Они представляют собой систему, состоящую из механического привода, головок чтения/записи, нескольких носителей и контроллера, обеспечивающего работу всего устройства и передачу данных. На рис. 3.12 изображен винчестер с форм-фактором 3,5".

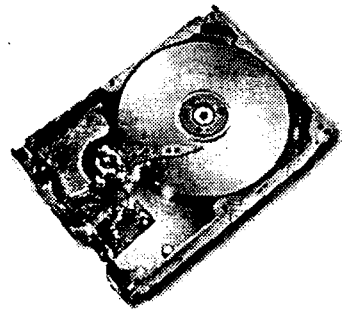


Рис. 3.12. Винчестер с форм-фактором 3,5"

Магнитная головка (несколько магнитных головок в специальном позиционере) является одной из наиболее важных частей устройства. Конструкция магнитных головок постоянно совершенствуется. Различают следующие типы: монолитные, изготовленные из ферритов, композиционные, состоящие из нескольких видов материалов (стекло, сплавы, керамика), тонкопленочные, изготавливаемые методом фотолитографии, магниторезистивные, состоящие из двух — для записи и для чтения.

Носитель информации состоит из нескольких дисков, каждый из которых имеет две рабочих поверхности. При записи информации используются магнитные свойства слоя, нанесенного на поверхность. Диски закреплены на шпинделе

двигателя. Скорость вращения дисков может быть 3600, 4500, 5400, 7200, 10000, 12000 об/мин. С увеличением скорости вращения дисков увеличивается производительность всей системы.

Геометрические размеры устройства определяются форм-фактором и размером по высоте: Full Height (FH) — полная высота 3,25", Half-Height (HH) — половинная высота или Low-Profile (LP) — низкий профиль 1".

Каждая поверхность любого из дисков разбивается на отдельные дорожки. Дорожки на одной вертикали на всех поверхностях образуют цилиндр. Дорожка разбивается на секторы. Доступ к необходимой информации осуществляется по номеру дорожки номеру цилиндра, номеру сектора.

Плотность записи на внешних секторах меньше, чем на внутренних секторах. В современных винчестерах форм-фактора 3,5", 5,25" диск разбивают на зоны, в пределах которых количество секторов постоянно. Чем зона дальше от центра, тем больше она содержит секторов.

Среди основных параметров, определяющих производительность винчестера, можно выделить следующие: среднее время доступа, которое определяется временем позиционирования магнитных головок на дорожке и временем ожидания сектора, и скорость обмена данными, которая в основном зависит от используемого интерфейса.

Накопители на сменных дисках

Приводы для *floppy*-дисков появились уже на первых персональных компьютерах. Форм-фактор этих дисков 5,25". Емкость одного диска составляла 160 Кбайт. Со временем емкость одного дискового носителя увеличилась до 1,2 Мбайт. Следующий этап — форм-фактор 3,5". Емкость одного носителя 720 Кбайт, 1,44 Кбайт. Операции чтения/записи осуществляются контактным способом, когда магнитная головка устройства соприкасается с поверхностью носителя. У таких устройств невысокая плотность записи, скорость обмена, значительное время доступа.

Современные разработки floppy-устройств позволяют улучшить показатели подобных устройств, используя гибкие диски форм-фактора 3,5", на которых можно хранить до 100—200 Мбайт информации. Некоторые накопители могут работать с обычными floppy-дисками, некоторые — нет. Для увеличения плотности записи используются современные технологии, такие как применение лазерного луча для точного позиционирования магнитной головки устройства, эффекта Бернулли для бесконтактного способа записи/чтения.

MOD (Magneto-Optical Disk) — *магнитооптические* диски имеют различную емкость от 128 Мбайт до 640 Мбайт. Запись производится магнитным способом после нагревания лазером магнитного слоя до определенной температурной точки (точка Кюри). Надежность хранения информации обеспечивается тем, что при обычной температуре информация не подвержена действию внешних магнитных полей.

Устройства CD-ROM используют носители, емкостью до 650 Мбайт. Носитель представляет собой диск со светоотражающим слоем на одной стороне, на которой хранится информация. На диск нанесена дорожка-спираль от центра к краю диска, состоящая из отражающих и не отражающих свет точек. Считывание производится лазерным лучом. Скорость считывания информации определяется в сравнении со стандартом Audio CD — 150 Кбайт/с. На рис. 3.13 изображено устройство CD-ROM с компакт-диском. Низкоскоростные приводы использовали линейную скорость считывания информации. При этом угловая скорость увеличивалась при чтении на внешней стороне диска. В высокоскоростных приводах стали использовать постоянную угловую скорость. В маркировках таких приводов указывается максимально достижимая скорость считывания информации. Скорость чтения неполного диска никогда не достигнет максимального значения.



Рис. 3.13. Устройство CD-ROM с компакт-диском

Накопители CD-R с возможностью записи позволяют однократно записывать информацию на диски диаметром 120 и 80 мм. Луч лазера прожигает пленку на поверхности диска, меняя его отражательную способность. Перезапись невозможна. Такие диски читаются на любом приводе CD-ROM.

Накопители CD-RW позволяют делать многократную запись на диск. Здесь используется свойство рабочего слоя переходить под действием лазерного луча в кристаллическое или аморфное состояние, имеющие разную отражательную способность. Такие диски могут не читаться на некоторых, особенно устаревших приводах CD-ROM.

Накопители DVD (Digital Versatile Disc) — цифровой универсальный диск. Предназначен для хранения видео, аудио высокого качества, компьютерной информации большого объема. На рис. 3.14 изображено устройство DVD. Односторонние однослойные DVD имеют емкость 4,7 Гбайт информации, двухслойные — 8,5 Гбайт; двухсторонние однослойные вмещают 9,4 Гбайт, двухслойные — 17 Гбайт. Плотность записи выше, чем у обычных CD-ROM. Накопители DVD могут читать обычные CD-ROM-диски. Двухскоростные накопители DVD могут читать и CD-R, и CD-RW-диски.

Накопители DVD-RAM позволяют записывать и перезаписывать информацию. На одностороннем однослойном диске можно разместить 2,58 Гбайт данных, на двухстороннем — 5,2 Гбайт. Конкурирующий стандарт *DVD-R* позволяет хранить 3,95 Гбайт информации.

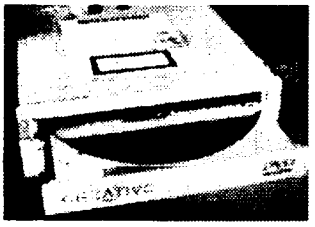


Рис. 3.14. Устройство DVD

Накопители на сменных жестких дисках используют технологию винчестеров. Параметры таких устройств приближаются к параметрам устройств с жесткими несъемными дисками. Как правило, используется форм-фактор 3,5". Емкость одного носителя варьируется от 230 Мбайт до 2 Гбайт.

RAID

RAID (Redundant Array of Inexpensive Disks) массив — устройство, состоящее из нескольких винчестеров и RAID-контроллера. Такое устройство обладает большим объемом дискового пространства, повышенной скоростью обмена данными, значительной надежностью хранения информации. RAID level 0 обеспечивает высокую скорость обмена данными за счет параллельной записи на несколько дисков. Скорость обмена находится в прямой пропорции от количества используемых винчестеров. Надежность такого RAID-массива невелика. RAID level 1 обеспечивает надежность хранения информации за счет дублирования информации на несколько дисков. RAID level 3 и 5 являются промежуточными вариантами, где надежность обеспечивается не простым дублированием информации, а дополнительной избыточной информацией, необходимой для восстановления информации при сбоях системы.

RAID-массивы допускают замену винчестера без отключения питания и останова компьютерной системы, без потерь информации.

3.3.3. Устройства ввода информации

Клавиатура

Клавиатура является основным устройством ввода информации в персональный компьютер. В настоящее время существует большое количество видов клавиатур, отличающихся в основном эргономическими качествами. В клавиатуру встраиваются дополнительные устройства, такие как микрофон, акустическая система, тачпад и др. Клавиатура может оснащаться дополнительными клавишами, например Старт, для использования на компьютерах с операционной системой Windows. Несмотря на эти новшества, основное назначение клавиатуры — ввод символьной информации. Клавиатура содержит 101 и более клавиш (у мобильных компьютеров количество клавиш существенно меньше).

Мышь

Мышь — манипулятор, созданный для удобства ввода информации в компьютер. Мышь не заменяет клавиатуру. Мышь получила распространение на компьюте-

рах, на которых используются графические программные оболочки. Мышь имеет две или три кнопки. Двухкнопочная мышь может иметь специальное колесико между клавишами для быстрого просмотра многостраничной информации. Такое же назначение имеет качающаяся средняя кнопка.

Механические мыши используют шарик, передающий перемещение мыши на специальные датчики. Более точного позиционирования позволяет достичь оптическая мышь, для работы которой используется специальный коврик с мельчайшей сеткой темных и светлых полос.

Мышь может быть подключена к компьютеру через последовательный COM-порт, порт PS/2, порт USB. Последний вариант предпочтительней. Используются и беспроводные мыши, работающие в инфракрасном диапазоне или на радиочастотах.

Другие координатные устройства ввода информации

Джойстик (joystick) — рычажный манипулятор для ввода координатной информации. Следующие устройства нашли свое первое применение в мобильных компьютерах. *Трекбол* (trackball) — перевернутая мышь с увеличенным шариком, который необходимо вращать пальцем. *Трекпойнт* (trackpoint) — маленький джойстик, который размещается обычно в центре клавиатуры. Управляется нажатием пальца. *Тачпад* (touchpad) — площадка, чувствительная к нажатию пальца.

Сканер

Сканер — устройство ввода в компьютер информации с бумажного или другого немашинного носителя. Сканер (рис. 3.15) используется для ввода текста, графических изображений. Отраженный от сканируемого изображения свет попадает на матрицу или линейку светочувствительных элементов на основе приборов с зарядовой связью (ПЗС), которые преобразуют аналоговый сигнал в цифровой. Ручные сканеры необходимо перемещать рукой, стараясь выдерживать определенную скорость и равномерность перемещения. Они имеют небольшую ширину захвата и невысокое разрешение. В планшетных сканерах сканирующая головка перемещается относительно изображения с помощью шагового двигателя. Рулонные сканеры протягивают сканируемые изображения через сканирующее устройство. Барабанные сканеры в качестве светочувствительного элемента используют фотоэлектронный умножитель, что позволяет получать высококачественный результат. В однопроходных сканерах используются три параллельных линейки ПЗС для получения информации о трех основных цветах. В трехпроходных устройствах используется только одна линейка ПЗС. За один проход получается информация об одном цвете.

Оптическое разрешение сканера определяет размер самых мелких деталей, которые сканер может передать без искажения. Разрешение сканера зависит от количества используемых элементов на единицу длины в линейке ПЗС и шага перемещения сканирующего устройства. Измеряется в dpi (dot per inch), количестве точек на дюйм. Разрешение современных сканеров может быть от 200 dpi для ручных сканеров до 1200 dpi для планшетных.

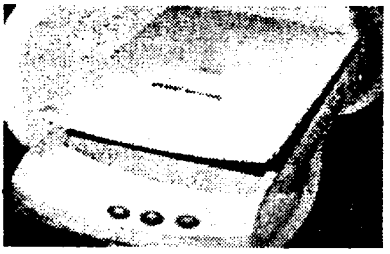


Рис. 3.15. Сканер

Цветовая разрядность сканера определяется количеством битов, используемых для хранения информации о цвете. Современные сканеры используют не менее 24 бит, по 8 бит на каждый цвет. В высококачественных сканерах используется 30 или 36 бит.

Графическая информация имеет большие объемы. Поэтому для коммутирования с компьютером необходимо отдавать предпочтение быстрым интерфейсам. Наиболее распространенный интерфейс для сканеров — интерфейс SCSI или USB.

Важной характеристикой сканера является количество и качество программного обеспечения сканера, которое должно обеспечивать определенный сервис: предварительный просмотр сканируемого изображения с выбором необходимой части, простейшие операции с изображением (поворот, инверсия), автоматическую и ручную коррекцию цветопередачи и контраста, режим копира. Сканер может вводить изображение, в том числе изображение текста, однако оно не может непосредственно использоваться как текст, введенный, например, с клавиатуры. Для распознавания текстов используются специальные программы OCR (Optical Character Recognition) оптического распознавания текста.

3.3.4. Устройства вывода информации

Монитор

CRT

Сегодня самый распространенный тип мониторов — это *CRT* (Cathode Ray Tube) мониторы. В основе этих мониторов лежит электронно-лучевая трубка (ЭЛТ). С фронтальной стороны внутренняя часть стекла трубки покрыта люминофором (Luminofor). Люминофор — это вещество, которое испускает свет при бомбардировке его заряженными частицами. Для создания изображения в CRT-мониторе используется электронная пушка, которая испускает поток электронов сквозь металлическую маску или решетку на внутреннюю поверхность стеклянного экрана монитора, которая покрыта разноцветными люминофорными точками. Поток электронов на пути к фронтальной части трубки проходит через модулятор интенсивности и ускоряющую систему. Светящиеся точки люминофора формируют изображение. Как правило, в цветном CRT-мониторе используются три электронные пушки, в отличие от одной пушки, применяемой в монохромных мониторах.

Глаза человека реагируют на основные цвета: красный (Red), зеленый (Green) и синий (Blue) — и на их комбинации, которые создают бесконечное число цветов. Каждая из трех пушек соответствует одному из основных цветов и посылает пучок электронов на различные частицы люминофора, чье свечение основными цветами с различной интенсивностью комбинируется и в результате формируется изображение с требуемым цветом. Электронный луч, предназначенный для красных люминофорных элементов, не должен влиять на люминофор зеленого или синего цвета. Чтобы добиться такого действия, используется специальная маска, обеспечивающая дискретность (растровость) изображения. ЭЛТ можно разбить на два класса — трехлучевые с дельтаобразным расположением электронных пушек и с планарным расположением электронных пушек. Трубки с планарным расположением электронных пушек еще называют кинескопами с самосведением лучей, так как воздействие магнитного поля Земли на три планарно расположенных луча практически одинаково и при изменении положения трубки относительно поля Земли не требуется производить дополнительные регулировки. В этих трубках применяются щелевые (Slot Mask) и теневые (Shadow Mask) маски.

Теневая маска — это самый распространенный тип масок для CRT-мониторов. Теневая маска состоит из металлической сетки перед частью стеклянной трубки с люминофорным слоем. Отверстия в металлической сетке работают как прицел, который обеспечивает то, что электронный луч попадает только на требуемые люминофорные элементы. Теневая маска создает решетку с однородными точками (триадами), где каждая такая точка состоит из трех люминофорных элементов основных цветов — зеленого, красного и синего, — которые светятся с различной интенсивностью под воздействием лучей из электронных пушек. Изменением тока каждого из трех электронных лучей можно добиться произвольного цвета элемента изображения, образуемого триадой точек.

Минимальное расстояние между люминофорными элементами одинакового цвета называется dot pitch (или *шаг точки*) и является показателями качества изображения. Шаг точки обычно измеряется в миллиметрах. Чем меньше значение шага точки, тем выше качество воспроизводимого на мониторе изображения.

Щелевая маска состоит из вертикальных линий. Вертикальные полосы разделены на эллиптические ячейки, которые содержат группы из трех люминофорных элементов трех основных цветов. Минимальное расстояние между двумя ячейками называется slot pitch (*щелевой шаг*). Чем меньше значение slot pitch, тем выше качество изображения на мониторе.

Апертурная решетка (aperture grill) — это тип маски, которая имеет решетку из вертикальных линий. Вместо точек с люминофорными элементами трех основных цветов апертурная решетка содержит серию нитей, состоящих из люминофорных элементов, выстроенных в виде вертикальных полос трех основных цветов. Такая система обеспечивает высокую контрастность изображения и хорошую насыщенность цветов, что вместе обеспечивает высокое качество мониторов с трубками на основе этой технологии. Маска представляет собой тонкую фольгу,

на которой процарапаны тонкие вертикальные линии. Она держится на горизонтальной проволочке (в 17" мониторах — на двух), тень от которой видна на экране. Эта проволочка применяется для гашения колебаний. Минимальное расстояние между полосами люминофора одинакового цвета называется strip pitch (или *шагом полосы*) и измеряется в миллиметрах. Чем меньше значение strip pitch, тем выше качество изображения на мониторе.

Нельзя напрямую сравнивать размер шага для трубок разных типов: шаг точек (или триад) трубки с теневой маской измеряется по диагонали, в то время как шаг апертурной решетки, иначе называемый горизонтальным шагом точек, — по горизонтали. Поэтому при одинаковом шаге точек трубка с теневой маской имеет большую плотность точек, чем трубка с апертурной решеткой. Для примера, 0,25 мм strip pitch приблизительно эквивалентно 0,27 мм dot pitch.

Трубки с теневой маской дают более точное и детализированное изображение, поскольку свет проходит через отверстия в маске с четкими краями. Поэтому мониторы с такими CRT хорошо использовать при интенсивной и длительной работе с текстами и мелкими элементами графики. Трубки с апертурной решеткой имеют более ажурную маску, она меньше заслоняет экран и позволяет получить более яркое, контрастное изображение насыщенных цветов. Мониторы с такими трубками хорошо подходят для настольных издательских систем и других приложений, ориентированных на работу с цветными изображениями.

Для управления электронно-лучевой трубкой необходима управляющая электроника, качество которой во многом определяет и качество монитора. Разница в качестве управляющей электроники, создаваемой разными производителями, является одним из критериев, определяющих разницу между мониторами с одинаковой электронно-лучевой трубкой. Эта электроника должна оптимизировать усиление сигнала и управлять работой электронных пушек, которые инициируют свечение люминофора, создающего изображение на экране. Выводимое на экране монитора изображение выглядит стабильным, хотя на самом деле таковым не является. Изображение на экране воспроизводится в результате процесса, в ходе которого свечение люминофорных элементов инициируется электронным лучом, проходящим последовательно по строкам в следующем порядке: слева направо и сверху вниз на экране монитора. Этот процесс происходит очень быстро, поэтому нам кажется, что экран светится постоянно. В сетчатке наших глаз изображение хранится около 1/20 с. Если луч последовательно пробегает по всем горизонтальным линиям сверху вниз за время, меньшее 1/25 с, мы увидим равномерно освещенный экран с небольшим мерцанием. Чем быстрее электронный луч проходит по всему экрану, тем меньше будет заметно мерцание картинки. Считается, что такое мерцание становится практически незаметным при частоте повторения кадров (проходов луча по всем элемента изображения) примерно 75 в секунду. Однако эта величина в некоторой степени зависит от размера монитора. Мерцание мониторов с большими углами обзора становится заметным при больших частотах кадров.

LCD

LCD (Liquid Crystal Display, жидкокристаллические мониторы) сделаны из вещества, которое находится в жидком состоянии, но при этом обладает некоторыми свойствами, присущими кристаллическим телам. Экран LCD-монитора представляет собой две прозрачные пластины с тонким слоем жидких кристаллов между ними. При появлении электрического поля свет, проходящий через жидкокристаллическую панель или отражающийся от нее, меняет плоскость поляризации. Для того чтобы человеческий глаз мог различать изменения в поляризации светового потока, добавляются два поляризационных фильтра. Экран разделен на отдельные элементы (ячейки), к которым подведены электроды, создающие электрическое поле. Для вывода цветного изображения необходима подсветка монитора сзади так, чтобы свет порождался в задней части LCD-дисплея. Это необходимо для того, чтобы можно было наблюдать изображение с хорошим качеством, даже если окружающая среда не является светлой. Цвет получается в результате использования трех фильтров, которые выделяют из излучения источника белого света три основных компонента.

Комбинируя три основных цвета для каждой точки или пиксела экрана, можно воспроизвести любой цвет.

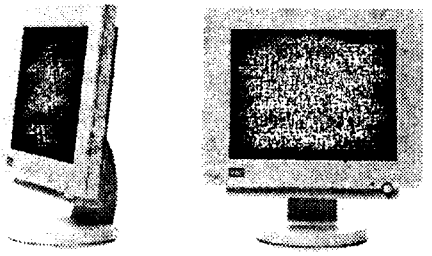


Рис. 3.16. LCD-дисплей

Первые LCD-дисплеи были очень маленькими, около 8", в то время как сегодня они достигли размеров 15" для использования в ноутбуках, а для настольных компьютеров производятся 19" и более LCD-мониторы. На рис. 3.16 изображены LCD-мониторы в настольном исполнении. Вслед за увеличением размеров следует увеличение разрешения, вследствие чего появляются новые проблемы, требующие своего решения с помощью специальных технологий.

Технология STN (Super Twisted Nematic) позволяет увеличить торсионный угол (угол кручения) ориентации кристаллов внутри LCD-дисплея с 90 до 270°, что обеспечивает лучшую контрастность изображения при увеличении размеров монитора. Часто STN-ячейки используются в паре. Это называется *технологией DSTN* (Double Super Twisted Nematic), и этот метод очень популярен для мониторов портативных компьютеров, использующих дисплеи с пассивной матрицей, где DSTN обеспечивает улучшение контрастности при отображении изображе-

ний в цвете. Две STN-ячейки располагаются вместе так, чтобы при вращении они двигались в разных направлениях. Изображение формируется строка за строкой путем последовательного подведения на отдельные ячейки управляющего напряжения, делающего их прозрачными. Из-за довольно большой электрической емкости ячеек напряжение на них не может изменяться достаточно быстро, поэтому обновление картинки происходит медленно, изображение дрожит на экране. Маленькая скорость изменения прозрачности кристаллов не позволяет правильно отображать движущиеся изображения. Кроме того, между соседними электродами возникает некоторое взаимное влияние, которое может проявляться в виде колец на экране. Для решения части названных проблем применяют специальные хитрости, например разделение экрана на две части и применение двойного сканирования в одно и то же время обеих частей, в результате чего экран дважды регенерируется и изображение не дрожит и плавно отображается.

Также лучших результатов с точки зрения стабильности, качества, разрешения, гладкости и яркости изображения можно добиться, используя экраны с *активной матрицей* (active matrix). В активной матрице используются отдельные усилительные элементы для каждой ячейки экрана, компенсирующие влияние емкости ячеек и позволяющие значительно уменьшить время изменения их прозрачности. Активная матрица имеет лучшую яркость и возможность смотреть на экран даже с отклонением до 45° и более (то есть при угле обзора 120°—140°) без ущерба для качества изображения, что невозможно в случае с пассивной матрицей. При помощи активной матрицы можно отображать движущиеся изображения без видимого дрожания, так как время реакции дисплея с активной матрицей около 50 мкс против 300 мкс для пассивной матрицы и качество контрастности лучше, чем у CRT-мониторов. Яркость отдельного элемента экрана остается неизменной на всем интервале времени между обновлениями картинки, а не представляет собой короткий импульс света, излучаемый элементом люминофора CRT-монитора сразу после похождения по этому элементу электронного луча. Именно поэтому для LCD-мониторов достаточной является частота регенерации 60 Гц. Благодаря лучшему качеству изображений эта технология также используется и в мониторах для настольных компьютеров.

Функциональные возможности LCD-мониторов с активной матрицей почти такие же, как у дисплеев с пассивной матрицей. Разница заключается в матрице электродов, которая управляет ячейками жидких кристаллов дисплея. В случае с пассивной матрицей разные электроды получают электрический заряд циклическим методом при построчной регенерации дисплея, а в результате разряда емкостей элементов изображение исчезает, так как кристаллы возвращаются к своей изначальной конфигурации. В случае с активной матрицей к каждому электроду добавлен запоминающий транзистор, который может хранить цифровую информацию (двоичные значения 0 или 1), и в результате изображение сохраняется до тех пор, пока не поступит другой сигнал. Запоминающие транзисторы должны производиться из прозрачных материалов, что позволит световому лучу проходить сквозь них, а значит, транзисторы можно располагать на тыльной части

дисплея, на стеклянной панели, которая содержит жидкие кристаллы. Для этих целей используется пластиковая пленка, называемая Thin Film Transistor (или просто TFT). Ее толщина в пределах от 1/10 до 1/100 мкм. Технология создания TFT очень сложна, при этом имеются трудности с достижением приемлемого процента годных изделий из-за того, что число используемых транзисторов очень велико.

Разрешение LCD-мониторов одно, и его еще называют *native*, оно соответствует максимальному физическому разрешению CRT-мониторов. Это разрешение определяется размером пикселов, который у LCD-монитора фиксирован. LCD-монитор лучше всего воспроизводит изображение именно с таким разрешением. При этом есть возможность выводить на экран изображение с меньшим, чем native, разрешением. Для этого есть два способа. Первый называется центрированием, когда для отображения изображения используется только то количество пикселов, которое необходимо для формирования изображения с более низким разрешением. В результате изображение получается не во весь экран, а только в середине. Все неиспользуемые пикселы остаются черными, то есть вокруг изображения образуется широкая черная рамка. Второй метод называется растяжением. Суть его в том, что при воспроизведении изображения с более низким, чем native, разрешением используются все пикселы, то есть изображение занимает весь экран. Однако из-за того, что изображение растягивается на весь экран, возникают небольшие искажения и ухудшается резкость.

Стоит отметить и такую особенность части LCD-мониторов, как возможность поворота самого экрана на 90° с одновременным автоматическим разворотом изображения. В результате, например, если вы занимаетесь версткой, то теперь лист формата A4 можно полностью уместить на экране без необходимости использовать вертикальную прокрутку, чтобы увидеть весь текст на странице. Эта функция становится почти стандартной.

К преимуществам LCD-мониторов можно отнести то, что они действительно плоски в буквальном смысле этого слова, а создаваемое на их экранах изображение отличается четкостью и насыщенностью цветов. Отсутствуют искажения на экране и масса других проблем, свойственных традиционным CRT-мониторам. Добавим, что потребляемая и рассеиваемая мощность у LCD-мониторов существенно ниже, чем у CRT-мониторов.

При производстве мониторов используются и другие, более экзотические на данный момент технологии.

Плазменные мониторы PDP (Plasma Display Panels). Работа плазменных мониторов очень похожа на работу неоновых ламп, которые сделаны в виде трубки, заполненной инертным газом низкого давления. Внутрь трубки помещена пара электродов, между которыми зажигается электрический разряд и возникает свечение. Плазменные экраны создаются путем заполнения пространства между двумя стеклянными поверхностями инертным газом, например аргоном или неоном. Затем на стеклянную поверхность помещают маленькие прозрачные электроды, на которые подается высокочастотное напряжение. Под действием этого напряжения в прилегающей к электроду газовой области возникает электрический раз-

ряд. Плазма газового разряда излучает свет в ультрафиолетовом диапазоне, который вызывает свечение частиц люминофора в диапазоне, видимом человеком. Фактически, каждый пиксел на экране работает, как обычная флуоресцентная лампа (иначе говоря, лампа дневного света). Высокая яркость и контрастность наряду с отсутствием дрожания являются большими преимуществами таких мониторов.

Технология FED (Field Emission Display). Мониторы FED основаны на процессе, который немного похож на тот, что применяется в CRT-мониторах, так как в обоих методах применяется люминофор, светящийся под воздействием электронного луча. В FED-мониторе используется множество маленьких источников электронов, расположенных за каждым элементом экрана, и все они размещаются в пространстве по глубине меньшей, чем требуется для CRT. Каждый источник электронов управляется отдельным электронным элементом, так же как это происходит в LCD-мониторах, и каждый пиксел затем излучает свет благодаря воздействию электронов на люминофорные элементы, как и в традиционных CRT-мониторах. При этом FED-мониторы очень тонкие.

Кратко остановимся на основных характеристиках монитора.

Под *размером монитора* обычно понимают размер диагонали монитора (в дюймах), при этом размер видимой пользователем области экрана обычно несколько меньше, в среднем на 1", чем размер трубки. Производители могут указывать в сопровождающей документации два размера диагонали, при этом видимый размер обычно обозначается в скобках или с пометкой «Viewable size», но иногда указывается только один размер, размер диагонали трубки.

Разрешение монитора (или разрешающая способность) связана с размером отображаемого изображения и выражается в количестве точек по ширине (по горизонтали) и высоте (по вертикали) отображаемого изображения. Например, если говорят, что монитор имеет разрешение 640×480, это означает, что изображение состоит из 640×480 = 307 200 точек в прямоугольнике, чьи стороны соответствуют 640 точкам по ширине и 480 точкам по высоте. Это объясняет, почему более высокое разрешение соответствует отображению более содержательного (детального) изображения на экране. Понятно, что разрешение должно соответствовать размеру монитора, иначе изображение будет слишком маленьким, чтобы его разглядеть. Возможность использования конкретного разрешения зависит от различных факторов, среди которых возможности самого монитора, возможности видеокарты и объем доступной видеопамяти, которая ограничивает число отображаемых цветов.

На величину максимально поддерживаемого монитором разрешения напрямую влияет *частота горизонтальной развертки* электронного луча, измеряемая в kHz (килогерцах, кГц). Значение горизонтальной развертки монитора показывает, какое предельное число горизонтальных строк может прочертить электронный луч на экране монитора за одну секунду. Соответственно, чем выше это значение (а именно оно, как правило, указывается на коробке для монитора), тем большее разрешение может поддерживать монитор при приемлемой частоте кадров.

Частота кадровой развертки для экрана CRT-мониторов — это параметр, определяющий, как часто изображение на экране заново перерисовывается. Частота

регенерации измеряется в Hz (герцах, Гц), где 1 Гц соответствует одному циклу в секунду. Частота регенерации монитора должна быть достаточной, чтобы не было заметно мерцания изображения. Чем выше частота регенерации, тем более устойчивым выглядит изображение на экране. Мерцание изображения приводит к утомлению глаз, головным болям и даже к ухудшению зрения. Минимально безопасной частотой кадров считается 75 Гц. Исследования показали, что при частоте вертикальной развертки выше 110 Гц глаз человека уже не может заметить никакого мерцания.

Мониторы опасны для здоровья. С целью снижения риска для здоровья различными организациями были разработаны рекомендации по параметрам мониторов, которым должны следовать производители устройств. Все стандарты безопасности для мониторов регламентируют максимально допустимые значения электрических и магнитных полей, создаваемых монитором при работе. Практически в каждой развитой стране есть собственные стандарты, но особую популярность во всем мире завоевали стандарты, разработанные в Швеции и известные под именами TCO и MPRII.

TCO (The Swedish Confederation of Professional Employees) — Шведская Конфедерация профессиональных коллективов рабочих. Стандарты TCO разрабатываются с целью гарантировать пользователям компьютеров безопасную работу. Этим стандартам должен соответствовать каждый монитор, продаваемый в Швеции и в Европе. Рекомендации TCO используются производителями мониторов для создания более качественных продуктов, которые менее опасны для здоровья пользователей. Суть рекомендаций TCO состоит не только в определении допустимых значений различного типа излучений, но и в определении минимально приемлемых параметров мониторов, например поддерживаемых разрешений, интенсивности свечения люминофора, запаса яркости, энергопотребления, шумности и т. д. Более того, кроме требований в документах TCO приводятся подробные методики тестирования мониторов.

В состав разработанных TCO рекомендаций сегодня входят три стандарта: TCO 92, TCO 95 и TCO 99. Цифры означают год их принятия. Большинство измерений во время тестирований на соответствие стандартам TCO проводятся на расстоянии 30 см спереди от экрана и на расстоянии 50 см вокруг монитора.

Стандарт TCO 92 был разработан исключительно для мониторов и определяет величину максимально допустимых электромагнитных излучений при работе монитора, а также устанавливает стандарт на функции энергосбережения мониторов. Кроме того, монитор, сертифицированный по TCO 92, должен соответствовать стандарту на энергопотребление NUTEK и европейским стандартам на пожарную и электрическую безопасность.

Стандарт TCO 95 распространяется на весь персональный компьютер, то есть на монитор, системный блок и клавиатуру, и касается эргономических свойств, излучений (электрических и магнитных полей, шума и тепла), режимов энергосбережения и экологии (с требованием обязательной адаптации продукта и технологического процесса производства на фабрике). Стандарт TCO 95 существует наряду с TCO 92 и не отменяет последний. Требования TCO 95 по отношению к

электромагнитным излучениям мониторов не являются более жесткими, чем по TCO 92.

TCO 99 предъявляет более жесткие требования, чем TCO 95, в следующих областях: эргономика (физическая, визуальная и удобство использования), энергия, излучение (электрических и магнитных полей), окружающая среда и экология, а также пожарная и электрическая безопасность. Стандарт TCO 99 распространяется на традиционные CRT-мониторы, плоскопанельные мониторы (Flat Panel Displays), портативные компьютеры (Laptop и Notebook), системные блоки и клавиатуры.

MPR II — это еще один стандарт, разработанный в Швеции. MPR II определяет максимально допустимые значения излучения магнитного и электрического полей, а также методы их измерения. Стандарты TCO жестче, чем MPR II.

Видеоадаптер

Видеоадаптер (рис. 3.17) представляет собой специальное устройство, сконструированное в виде отдельной платы расширения. Видеоадаптер управляет выводом информации на монитор. Характеристики видеосистемы зависят как от параметров используемого монитора, так и от установленного в компьютере видеоадаптера.

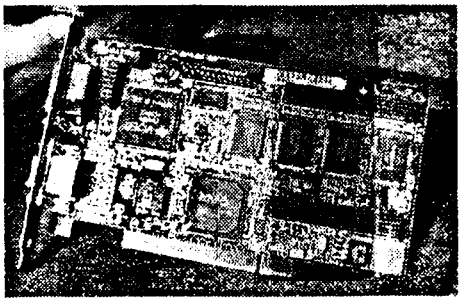


Рис. 3.17. Видеоадаптер

Принтер

Принтеры — устройства для вывода информации на бумажный носитель (в настоящее время используются и прозрачные пленки). По технологии печати можно выделить принтеры: матричные, струйные, лазерные, LED и др.

Матричный принтер

Печатающим элементом является головка с 9, 18 или 24 иголками, удар которых через красящую ленту формирует выводимый символ на бумаге. Используется бумага формата A4 или A3. Возможно использование рулонной бумаги.

Струйный принтер

Струйные принтеры — это безударные устройства, работающие практически бесшумно. Печатающая головка разбрызгивает специальные чернила через сопла, количество которых может достигать 256, на бумагу. Толщина струи меньше

толщины иглы матричного принтера, поэтому качество печати выше. Печатающие головки могут быть совмещенными с чернильницей или нет. В первом случае заменяется весь блок. Во втором — лишь чернильница, хотя сама печатающая головка тоже является расходным материалом.

Принтеры непрерывного действия, когда чернила непрерывно разбрызгиваются на бумагу при печати, а излишки возвращаются обратно в чернильницу, в настоящее время не имеют широкого распространения. Принтеры дискретного действия, когда чернила используются только по необходимости, используют либо специальную пузырьковую технологию, либо пьезоэффект. Принтеры с пузырьковой технологией в печатающей головке имеют небольшой нагревательный элемент, который под действием тока очень быстро нагревает чернила. Необходимая порция чернил выбрасывается из сопла. Одновременно с остыванием всасывается очередная чернильная капля. В принтерах второго типа используется принцип изменения размеров пьезокристалла под действием тока.

Лазерный принтер

В *лазерных принтерах* (рис. 3.18) используется лазерный луч, который формирует изображение на специальном фотопроводящем барабане. Перед печатью поверхность барабана электрически заряжается. Луч лазера изменяет потенциал в зависимости от выводимого изображения. После формирования очередной строки барабан поворачивается на определенный шаг для формирования следующей строки. Этот шаг определяет физическую разрешающую способность принтера. Формирование печатаемой страницы на барабане принтера похоже на формирование изображение на экране монитора.

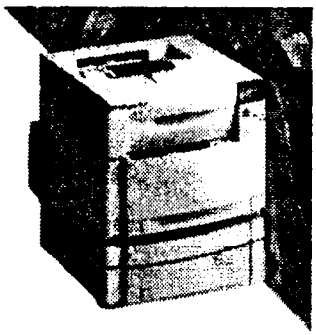


Рис. 3.18. Лазерный принтер

После формирования страницы каждый участок барабана имеет свой потенциал, благодаря которому притягиваются заряженные частицы порошкообразного тонера. Барабан покрывается тонером, после чего тонер переносится на заряженный лист бумаги и закрепляется специальными резиновыми валиками за счет разогрева тонера до температуры расплава.

LED-принтер

В *LED-принтерах* (Light Emmiting Diode) вместо полупроводникового лазера используются мельчайшие светодиоды.

Принтеры с другими принципами работы

Печатающая головка термопринтера состоит из мельчайших нагревательных элементов, которые переносят специальное красящее вещество на бумагу именно в том месте, где обеспечивается необходимая температура. Различаются принтеры с переносом специальной красящей мастики и принтеры, в которых перенос вещества осуществляется в газообразном состоянии. Эти устройства имеют отличные показатели печати почти фотографического качества.

Основные характеристики принтера

Разрешение принтера характеризует, насколько мелкие детали изображения может передавать принтер. Этот показатель измеряется в dpi (dot per inch), в точках на дюйм. Современные принтеры имеют разрешение от 600 до 1200 dpi. Этот показатель приводится для черно-белых изображений. Для цветных и полутоновых изображений этот показатель ниже. Многие принтеры используют специальные алгоритмы, улучшающие разрешение.

На качество получаемого изображения большое влияние оказывает используемая бумага. Качество бумаги особенно актуально для струйных принтеров, использующих чернила, которые могут расплываться, ухудшая качество печати.

Количество цветов имеет важное значение. Обычно используются три основных цвета, смешением которых получается любой цвет. Черный цвет получить трудно, поэтому он добавляется к другим. При использовании трех цветов достаточно сложно получить приемлемый результат при печати изображений светлых тонов. Поэтому некоторые модели принтеров используют 6 цветов: три основных цвета и дополнительно три светлых. В основном это относится к струйным принтерам.

Качество драйверов и используемые алгоритмы цветоделения играют немаловажную роль для получения качественной печати.

Скорость печати измеряется количеством страниц в единицу времени, обычно в минуту. Современные принтеры имеют скорость печати от 1-2 до 10-12 страниц в минуту.

Ресурс принтера определяется временем непрерывной эксплуатации и измеряется количеством страниц в единицу времени, обычно в месяц. Превышение ресурса существенно повышает вероятность поломки принтера.

Ресурс одной заправки вместе с ее стоимостью определяет стоимость эксплуатации принтера. Лазерные принтеры обычно стоят дороже струйных, но эксплуатация последних несколько дороже.

3.3.5. Периферийные устройства

Стандартные порты ввода/вывода

Различные устройства ввода, вывода, хранения информации и другие периферийные устройства могут и подключаются через устройства сопряжения или адаптеры, реализующие различные интерфейсы. Последовательный интерфейс предполагает передачу данных последовательно бит за битом. Параллельный интерфейс позволяет передавать несколько битов данных одновременно. В со-

временных компьютерах многие подобные устройства стали стандартными устройствами, интегрированными на материнской плате.

Стандартные последовательный и параллельный интерфейсы называются *портами ввода/вывода.*

Параллельный порт чаще используется для подключения принтера. Современные стандарты параллельных интерфейсов поддерживают скорость передачи данных на уровне 2-5 Мбит/с. Подключаемые устройства должны располагаться в непосредственной близости от компьютера.

Последовательный порт используется для многих устройств, таких как мышь, внешний модем и др. Скорость обмена данными не превышает 9600 бит/с, но данные могут передаваться на значительное расстояние.

Инфракрасный порт используется для подключения устройств к портативным компьютерам, а также для подключения принтеров. Стандарты инфракрасного порта основываются на стандарте последовательного порта. Это беспроводное соединение, основанное на инфракрасном излучении, обеспечивает скорость передачи данных до 4 Мбит/с.

Порт USB устанавливается на каждый современный компьютер. Он должен заменить последовательный и параллельный порты. Порт USB реализует последовательный интерфейс. Через этот порт можно подключить до 128 устройств. Скорость передачи данных составляет 12 Мбит/с. Интерфейс USB поддерживает автоопределение и автоконфигурацию подключаемого устройства и возможность подключения без перезагрузки компьютера. На рис. 3.19 изображены разъемы USB на клавиатуре.



Рис. 3.19. Разъемы USB на клавиатуре

Интерфейс FireWare используется для подключения высокоскоростных устройств, для которых недостаточна скорость порта USB. Интерфейс поддерживает синхронную и асинхронную передачу данных со скоростью до 400 Мбит/с. Скорость обмена данными может меняться для разных устройств. На один порт могут подключаться до 63 устройств. Стандарт поддерживает автоопределение устройств и «горячее» подключение. Интерфейс может обрабатывать многие операции ввода/вывода, не занимая ресурсов центрального микропроцессора.

Модем

Модем — это устройство, необходимое для организации связи между компьютерами, обычно с использованием телефонных каналов. Оно выполняет функции модуляции и демодуляции (МОдуляция-ДЕМодуляция) информационных сигналов, то есть преобразования входных аналоговых сигналов телефонной линии в

цифровые биты, и наоборот. Модемы различаются методами модуляции: частотная, фазовая модуляция или их сочетание.

Основной характеристикой модема является скорость передачи несжатых данных без коррекции, которая измеряется в битах в секунду.

Для увеличения фактической скорости передачи данных используется сжатие данных, которое осуществляет модем.

Телефонная линия является не самым надежным способом соединения между компьютерами. Во время передачи данных возникают помехи, ошибки в данных, данные теряются. Модем исправляет ошибки, возникшие при передаче данных.

Для организации соединения между двумя компьютерами необходимы два модема, которые «понимают» друг друга. Разработкой стандартов и протоколов работы модемов занимается Международный союз электросвязи.

Модемы могут быть как во внешнем, так и во внутреннем исполнении.

Звуковая карта

Звуковая карта используется для воспроизведения и записи звуковых сигналов. При записи аналоговый сигнал преобразуется в цифровой (оцифровка). Глубина оцифровки определяется количеством используемых битов (8, 16 бит или больше). При воспроизведении звука используется несколько способов. Цифровой сигнал преобразуется в аналоговый, который подается на акустическую систему. Синтез звука — другой способ. Управляющая последовательность поступает на синтезатор, который формирует звук либо на основе использования частотной модуляции (FM, Frequency Modulation), либо с использованием таблицы волн (WT, Wave Table). Использование WT-синтезаторов дает более естественное звучание. Образцы звучания для WT-синтеза хранятся в специальных микросхемах на звуковой плате.

Звуковые карты должны поддерживать стандарты AdLib, Sound Blaster, Sound Blaster Pro, обеспечивать совместимость с Microsoft DirectX.

Сетевая карта

Сетевая карта (рис. 3.20) используется для объединения компьютеров в локальную сеть. Сетевые карты используют высокоскоростные интерфейсы сопряжения с компьютером. Основной характеристикой является скорость передачи данных.

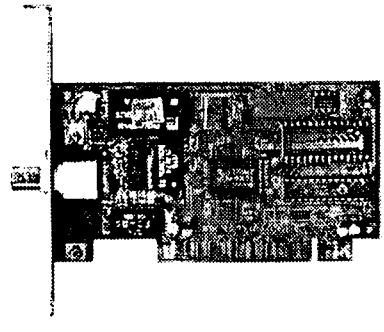


Рис. 3.20. Сетевая карта с интерфейсом PCI

Ключевые понятия

- компьютер;
- процессор;
- системная память;
- кэш-память;
- видеопамять;
- монитор;
- клавиатура;
- мышь;
- винчестер;
- системная и локальные шины;
- модем;
- сетевая карта.

Контрольные вопросы

1. Классификация компьютеров по функциональному назначению.
2. Основные компоненты персонального компьютера.
3. Основные характеристики видеосистемы персонального компьютера.
4. Назначение сканера.
5. Основные типы принтеров.
6. Что такое винчестер?
7. Основные интерфейсы накопителей на дисках.

Литература

1. Гук М. Аппаратные средства IBM PC. СПб.: Питер, 1999.
2. Нортон П., Гудман Дж. Персональный компьютер. Аппаратно-программная организация. Книга 1. Наиболее полное руководство в подлиннике. СПб.: BHV, 1999.

Источники в Интернете

1. iXBT Hardware – http://ixbt.stack.net
2. HardFactory – http://hard.net.ru
3. A complete illustrated Guide to the PC Hardware – http://www.mkdata.dk

Глава 4. Системное и сервисное программное обеспечение

В самом общем плане программное обеспечение для вычислительной техники может быть разделено на системное и прикладное. *Прикладным* называется программное обеспечение, которое непосредственно нацелено на решение профессиональных задач пользователя, а к *системному* относятся все программные средства, которые играют вспомогательную (служебную) роль в процессе организации работы компьютера и управления его ресурсами. Изучению системного программного обеспечения посвящена настоящая глава.

4.1. Операционная система

4.1.1. Понятие, основные функции и составные части операционной системы

Наверное, ни одно из существующих на настоящий момент определений понятия «операционная система» не может претендовать на универсальность и полноту. С учетом данной оговорки остановимся на следующей формулировке:

Операционная система (ОС) — это комплекс специальных программных средств, предназначенных для управления загрузкой, запуском и выполнением других (пользовательских) программ, а также для планирования и управления вычислительными ресурсами ЭВМ.

Важно подчеркнуть то, что операционная система — это именно **комплекс программ**, неоднородный по характеру и многоплановый по уровню. Этот комплекс программ динамичен по своему составу: из него можно удалять и в него добавлять определенные части. Та часть программ, которая взаимодействует с аппаратными средствами непосредственно и потому должна постоянно храниться в компьютере, составляет *ядро операционной системы*. В частности, программное обеспечение, входящее в состав ядра, отвечает за проверку работоспособности компьютера и выполнение элементарных (базовых) операций, связанных с работой дисплея, клавиатуры, магнитных накопителей и т. п.

Операционная система образует автономную среду, не связанную ни с одним из языков программирования. Любая прикладная программа связана с операционной системой и может эксплуатироваться только на тех компьютерах, где имеется аналогичная системная среда (или должна быть обеспечена возможность конвертации — преобразования программ).

Очевидно, что операционная система должна храниться на внешнем запоминающем устройстве, к которому может быть обеспечен относительно быстрый доступ. Например, на жестком диске, или на специальном гибком диске, который называется системным[1]. При включении компьютера операционная система автоматически загружается с диска в оперативную память.

Операционная система MS DOS, являвшаяся до последнего времени наиболее популярным продуктом данного класса для персональных компьютеров, была создана фирмой Microsoft в 1981 г. В настоящее время существуют ее версии 6.22 и 7.0 (в составе Windows 95), а также, условно говоря, системы-дублеры других фирм-разработчиков (DR DOS, PC DOS). Начиная с 1996 г. MS DOS распространяется в составе Windows 95 — 32-разрядной многозадачной и многопоточной операционной системы с графическим интерфейсом и расширенными сетевыми возможностями.

Операционная система как расширенная машина: использование большинства компьютеров на уровне машинного языка затруднительно, особенно это касается ввода/вывода. Например, для организации чтения блока данных с гибкого диска программист может использовать 16 различных команд, каждая из которых требует 13 параметров, таких как номер блока на диске, номер сектора на дорожке и т. п. Когда выполнение операции с диском завершается, контроллер возвращает 23 значения, отражающих наличие и типы ошибок, которые, очевидно, надо анализировать. Даже если не входить в курс реальных проблем программирования ввода/вывода, ясно, что среди программистов нашлось бы не много желающих непосредственно заниматься программированием этих операций. При работе с диском программисту-пользователю достаточно представлять его в виде некоторого набора файлов, каждый из которых имеет имя.

Программа, которая скрывает от программиста все реалии аппаратуры и предоставляет возможность простого, удобного просмотра указанных файлов, чтения или записи, — это, конечно, операционная система. Точно так же, как ОС ограждает программистов от аппаратуры дискового накопителя и предоставляет ему простой файловый интерфейс, операционная система берет на себя все малоприятные дела, связанные с обработкой прерываний, управлением таймерами и оперативной памятью, а также другие низкоуровневые проблемы. В каждом случае та абстрактная, воображаемая машина, с которой, благодаря операционной системе, теперь может иметь дело пользователь, гораздо проще и

[1] В ретроспективном плане необходимо заметить, что появление операционных систем, загружаемых в оперативную память с «быстрых» магнитных носителей, было своего рода знаковым этапом в их развитии. Последнее получило отражение, в том числе, и в их названиях: DOS — Disk Operation System.

удобнее в обращении, чем реальная аппаратура, лежащая в основе этой абстрактной машины.

С этой точки зрения функцией ОС является предоставление пользователю некоторой расширенной или виртуальной машины, которую легче программировать и с которой легче работать, чем непосредственно с аппаратурой, составляющей реальную машину.

Операционная система как система управления ресурсами: в соответствии со вторым подходом функцией ОС является распределение процессоров, памяти, устройств и данных между процессами, конкурирующими за эти ресурсы. ОС должна управлять всеми ресурсами вычислительной машины таким образом, чтобы обеспечить максимальную эффективность ее функционирования. Критерием эффективности может быть, например, пропускная способность или реактивность системы. Управление ресурсами включает решение двух общих, не зависящих от типа ресурса задач:

- планирование ресурса — то есть определение, кому, когда, а для делимых ресурсов — и в каком количестве необходимо выделить данный ресурс;
- отслеживание состояния ресурса — то есть поддержание оперативной информации о том, занят или не занят ресурс, а для делимых ресурсов — какое количество ресурса уже распределено, а какое свободно.

4.1.2. Классификация операционных систем

От эффективности алгоритмов управления локальными ресурсами компьютера во многом зависит эффективность всей сетевой ОС в целом. Поэтому, характеризуя сетевую ОС, часто приводят важнейшие особенности реализации функций ОС по управлению процессорами, памятью, внешними устройствами автономного компьютера. Так, например, в зависимости от особенностей использованного алгоритма управления процессором операционные системы делят на многозадачные и однозадачные, многопользовательские и однопользовательские, на системы, поддерживающие многонитевую обработку и не поддерживающие ее, на многопроцессорные и однопроцессорные системы.

1. **Поддержка многозадачности**. По числу одновременно выполняемых задач операционные системы могут быть разделены на два класса:
   - однозадачные (например, MS DOS, MSX);
   - многозадачные (ОС ЕС, OS/2, UNIX, Windows 95).

Однозадачные ОС в основном выполняют функцию предоставления пользователю виртуальной машины, делая более простым и удобным процесс взаимодействия пользователя с компьютером. Однозадачные ОС включают средства управления периферийными устройствами, средства управления файлами, средства общения с пользователем.

Многозадачные ОС, кроме выполнения вышеперечисленных функций, управляют разделением совместно используемых ресурсов, таких как процессор, оперативная память, файлы и внешние устройства.

2. **Поддержка многопользовательского режима**. По числу одновременно работающих пользователей ОС делятся:
   - на однопользовательские (MS DOS, Windows 3.x, ранние версии OS/2);
   - многопользовательские (UNIX, Windows NT).
3. **Вытесняющая и невытесняющая многозадачность**. Важнейшим разделяемым ресурсом является процессорное время. Способ распределения процессорного времени между несколькими одновременно существующими в системе процессами (или нитями) во многом определяет специфику ОС. Среди множества существующих вариантов реализации многозадачности можно выделить две группы алгоритмов:
   - невытесняющая многозадачность (NetWare, Windows 3.x);
   - вытесняющая многозадачность (Windows NT, OS/2, UNIX).

ПРИМЕЧАНИЕ

При невытесняющей многозадачности активный процесс выполняется до тех пор, пока он сам, по собственной инициативе, не отдаст управление операционной системе для того, чтобы та выбрала из очереди другой готовый к выполнению процесс. При вытесняющей многозадачности решение о переключении процессора с одного процесса на другой принимается операционной системой, а не самим активным процессом.

4. **Поддержка многонитевости**. Важным свойством операционных систем является возможность распараллеливания вычислений в рамках одной задачи. Многонитевая ОС разделяет процессорное время не между задачами, а между их отдельными ветвями (нитями).
5. **Многопроцессорная обработка**. Другим важным свойством ОС является отсутствие или наличие в ней средств поддержки многопроцессорной обработки — *мультипроцессирование*. Мультипроцессирование приводит к усложнению всех алгоритмов управления ресурсами.

В наши дни становится общепринятым введение в ОС функций поддержки многопроцессорной обработки данных. Такие функции имеются в операционных системах Solaris 2.x фирмы Sun, Open Server 3.x фирмы Santa Crus Operations, OS/2 фирмы IBM, Windows NT фирмы Microsoft и NetWare 4.1 фирмы Novell.

4.2. Файловые системы

4.2.1. Основные функции файловой системы

Основной задачей, которую решает *файловая система*, является обеспечение взаимодействия программ и физических устройств ввода/вывода, таких как накопители на жестких дисках, магнитных лентах и т. д.

Основные функции, выполняемые файловой системой, можно условно разделить на две группы:

- функции для работы с файлами, то есть их создание, удаление, изменение атрибутов, определение структуры файлов;
- функции для работы с данными, хранящимися в файлах, то есть чтение и запись, поиск и т. д.

Таким образом, в логическом плане файловую систему можно разделить на следующие составные части:

- файлы, хранящиеся на устройстве ввода/вывода;
- структура файлов;
- функции работы с файлами и их структурой.

4.2.2. Файлы и каталоги

В разных источниках по информатике и вычислительной технике определения термина «файл» так же, как и термина «операционная система», могут варьироваться. Наиболее распространенными являются формулировки, что «*файл* — это наименьший именованный массив информации» или «*файл* — основная единица организации информации на носителе». В различных файловых системах файлы могут описываться различными, вообще говоря, наборами параметров и характеристик.

Основным атрибутом файла является его *имя*. Имя файла — это символьная строка, длина которой зависит от конкретной файловой системы. Так, в файловой системе FAT (File Allocation Table), используемой в MS DOS, длина имени файла не может превышать 11 символов, 3 из которых предназначены для расширения. В ОС UNIX System V под имя файла отводится 14 символов, а в файловой системе NTFS (New Technology File System) для Windows NT — 255 символов. От файловой системы также зависит, какие символы могут использоваться в имени файла.

Поскольку существуют различные файловые системы, перед их разработчиками встает проблема обеспечения совместимости между ними. Например, система, позволяющая присваивать файлам длинные имена, должна уметь корректно преобразовывать их в короткие. Но, к сожалению, пока не удалось решить проблему использования единственного имени файла в различных файловых системах. Это связано, в первую очередь, с различием наборов разрешенных символов в различных системах.

В некоторых системах, например UNIX, одному файлу может быть сопоставлено несколько имен. Это особенно удобно для многопользовательских систем. В таком случае необходимо адресовать файл идентификатором, который будет связывать файл и его имена. В системе UNIX таким идентификатором служит номер индексного дескриптора.

Другие атрибуты файла, которые могут использоваться файловой системой, перечислены ниже:

- текущий размер файла;
- максимальный размер файла;
- длина записи;
- времена создания, последнего доступа и последнего изменения;
- владелец файла;
- создатель файла;
- информация о доступе к файлу;
- пароль для доступа к файлу;
- признак «только для чтения»;
- признак «скрытый файл»;
- признак «системный файл»;
- признак «архивный файл»;
- признак «двоичный/символьный»;
- признак «временный»;
- признак блокировки.

Для логической организации файлов используются каталоги. Каталог содержит файлы, объединенные по какому-либо признаку — их создатель, их тип, тема и т. д. *Каталог* — это файл, который содержит информацию о входящих в него файлах. Каталогов на носителе может быть много, и они могут иметь различные степени вложенности. Все каталоги, находящиеся на носителе, образуют иерархическую структуру.

Структура каталогов в зависимости от файловой системы может быть древовидной, когда один файл может входить только в один каталог, и сетевой, когда один файл может входить в различные каталоги. Пример системы с древовидной структурой каталогов — система FAT. Сетевая структура, более подходящая для многопользовательских систем, реализована в UNIX.

В так называемых DOS-совместимых (впрочем, и в некоторых других) системах к файлу обращаются с помощью *полного имени*, которое состоит:

- из *пути* (англ. Path) — последовательности имен каталогов, в которых содержится файл, разделенных символом \ (в UNIX для этого используется символ /);
- *собственно имени файла* (англ. Name);
- *расширения*[1] (англ. Extension), которое является необязательным элементом и, как правило, информирует о типе данных, хранимых в файле. Расширение отделяется от имени точкой (.)

[1] Понятие «расширение», видимо, следует толковать как «расширение имени файла».

В качестве примера может быть приведено следующее имя файла:

```
C:\DIR1\DIR2\DIR3\NAME.EXT
```

В различных файловых системах существуют ограничения на то, какие символы могут присутствовать в именах файлов.

4.2.3. Физическая организация данных на носителе

Физическая организация данных описывает правила расположения файлов на носителе. Расположение файла описывается расположением принадлежащих ему блоков. *Блоком* называется наименьшая единица данных, которой устройство ввода/вывода может обмениваться с памятью.

Простейший способ расположения файла — непрерывная последовательность блоков. Такой способ наряду со своим основным достоинством — простотой, которая позволяет адресовать файл всего лишь адресом его первого блока, имеет ряд существенных недостатков:

- во-первых, во время создания файла системе может быть не известен его размер, то есть система не знает, сколько места на носителе надо зарезервировать;
- во-вторых, неизбежна сильная фрагментация носителя.

Для устранения этих недостатков могут использоваться связанные блоки. В таком случае блок помимо данных содержит ссылку на следующий блок и т. д. Основным недостатком такого способа является то, что программа не может непосредственно обратиться к произвольному участку файла, и чтобы прочитать, например, последний блок, необходимо последовательно обратиться ко всем блокам файла. Недостатком такого способа организации данных является также то, что информация, хранящаяся в блоке данных, теряет однородность, так как содержит не только данные файла, но и служебную информацию.

Решить эти проблемы может использование связанного списка индексов, что делается, например, в MS DOS. При такой организации данных нет необходимости просматривать все данные для чтения последнего блока файла, достаточно просмотреть таблицу индексов. При этом сохраняется однородность данных, хранящихся в блоке, так как служебная информация хранится в отдельной области, располагающейся, в случае жесткого диска, на внутренних дорожках, что обеспечивает быстрый доступ к ней.

Перечисленные способы физической организации файлов на носителе схематично изображены на рис. 4.1.

4.2.4. Права доступа к файлу

В многопользовательских системах доступ к файлу разных пользователей должен быть разграничен, то есть операции с определенными файлами или каталогами должны быть разрешены для одних пользователей и запрещены для других. Это касается операций как с данными, так и с их структурой.

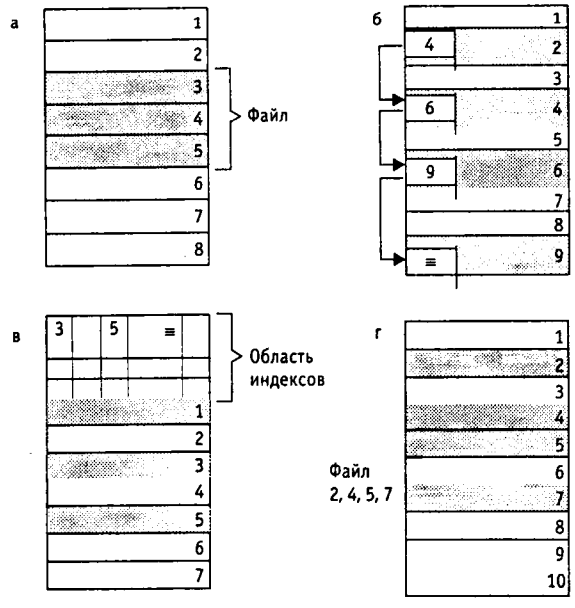


Рис. 4.1. Способы физической организации файлов

В общем случае система разграничения доступа использует матрицу доступа, столбцы в которой соответствуют файлам системы, а строки — пользователям. На пересечении описываются операции, которые данный пользователь может производить с данным файлом или каталогом.

Для облегчения администрирования некоторые системы позволяют описывать правила доступа для групп пользователей, тогда правила, установленные для группы, действуют для всех входящих в нее пользователей. Системой может быть предоставлена возможность включения одного пользователя в разные группы, а также описания правил как для групп, так и для отдельных пользователей, что позволяет гибко разграничивать доступ к файлам.

4.2.5. Другие функции файловых систем

Помимо функций, обеспечивающих основные операции с файлами, файловая система может предоставлять некоторые оптимизирующие функции. Основные из этих функций — *кэширование диска* и отображение файла в память.

Кэширование диска служит для ускорения доступа к наиболее часто используемым данным. При запросе данных система сначала ищет их в буфере, располагающемся на носителе с более высокой скоростью доступа. Если требуемые данные не найдены в буфере, они читаются с диска и заносятся в буфер. По мере наполнения буфера из него удаляются наиболее редко используемые данные.

Отображение файла в память заключается в создании образа файла, организованного как память, то есть представляющего собой непрерывную последовательность байтов. Это упрощает доступ к произвольному участку файла и в целом повышает скорость выполнения основных операций с данными.

4.2.6. Файловые системы FAT16, FAT32

В качестве примера файловых систем мы рассмотрим FAT16 и FAT32.

В сфере персональных компьютеров в 1987 г. возник кризис. Возможности файловой системы FAT, разработанной фирмой Microsoft за десять лет до этого для интерпретатора Standalone Disk Basic и позднее приспособленной для операционной системы DOS, были исчерпаны. FAT предназначалась для жестких дисков емкостью не свыше 32 Мбайт, а новые НЖМД (накопители на жестких магнитных дисках) большей емкости оказывались совершенно бесполезными для пользователей PC. Некоторые независимые поставщики предлагали собственные способы решения этой проблемы, однако лишь с появлением DOS 4.0 этот кризис был (на некоторое время!) преодолен.

Значительные изменения структуры файловой системы в DOS 4.0 позволили операционной системе работать с дисками емкостью до 128 Мбайт; а после внесения незначительных дополнений впоследствии этот предел был поднят до 2 Гбайт. В то время казалось, что такой объем памяти превышает любые мыслимые потребности. Однако если история персональных компьютеров чему-то и научила, то именно тому, что емкость, «превышающая любые мыслимые потребности», очень быстро становится «почти недостаточной для серьезных работ». Действительно, в настоящее время в продаже имеются жесткие диски емкостью, как правило, 2,5 Гбайт и выше, а когда-то очень высокий и избавивший нас от ограничений потолок в 2 Гбайт превратился в еще одно препятствие, которое предстояло преодолеть.

Для систем Windows 95 фирма Microsoft разработала новое расширение системы FAT – FAT32.

Области диска

Файловая система FAT предусматривает ряд специальных областей на диске, выделенных для организации пространства диска в процессе его форматирования, — головную запись загрузки, таблицу разбиения диска, запись загрузки, таблицу размещения файлов (от которой система и получила свое название[1]) и корневой каталог. На физическом уровне пространство диска разбивается на области по 512 байт, называемые секторами. В системе FAT место для файлов выделяется блоками, которые состоят из целого числа секторов и именуются *кластерами*. Число секторов в кластере должно быть кратно степени двойки. В Microsoft эти кластеры называют единицами выделения памяти (allocation unit). Обычно размер кластера можно определить, поделив объем памяти диска на 64 Кбайт (65 536 байт) и округлив результат до ближайшего числа, кратного степени двойки. Так, размер кластеров 1,2-гигабайтного диска составляет 32 Кбайт: если 1,2 Гбайт (1 258 291,2 Кбайт) поделить на 65 536, получим 19,2 Кбайт, а после округления — 32 Кбайт.

[1] Напомним, FAT — это аббревиатура от File Allocation Table.

Цепочка FAT

FAT представляет собой базу данных, связывающую кластеры дискового пространства с файлами. В этой базе для каждого кластера предусматривается только один элемент. Первые два элемента содержат информацию о самой системе FAT. Третий и последующие элементы ставятся в соответствие кластерам дискового пространства, начиная с первого кластера, отведенного для файлов. Элементы FAT могут содержать несколько специальных значений, указывающих, что кластер свободен, то есть не использован ни одним файлом (для FAT16 это значение составляет 0000H); кластер содержит один или несколько секторов с физическими дефектами и не должен использоваться (для FAT16 это значение составляет FFF7H); данный кластер — последний кластер файла (для FAT16 это значение составляет FFF8 FFFFH).

Для любого используемого файлом, но не последнего кластера элемент FAT содержит номер следующего кластера, занятого файлом.

Каждый каталог — корневой или подкаталог — также представляет собой базу данных. В каталоге DOS для каждого файла предусмотрена одна главная запись (в среде Windows 95 для длинных имен файлов введены дополнительные записи). В отличие от FAT, где каждый элемент состоит из единственного поля, записи для файла в каталоге состоят из нескольких полей (табл. 4.1). Некоторые поля — имя, расширение, размер, дата и время — могут быть выведены на экран по команде DIR. Но в системе FAT предусмотрено поле, которое не отображается командой DIR, — поле с номером первого кластера, отведенного под файл.

Таблица 4.1. Структура записи об атрибутах файла в FAT

Смещение	**0 — 7**	**8 — 10**	**11**	**12 — 21**	**22 — 23**	**24 — 25**	**26 — 27**	**28 — 31**
	Имя файла	Расширение	Атрибут	Не используется	Время	Дата	Начальный кластер	Размер файла
Размер	8 байт	3 байта	1 байт	10 байт	2 байта	2 байта	2 байта	4 байта

Запись в каталоге для файла состоит из восьми полей, содержащих основную информацию о файле, например его размер, начальный кластер на диске, время и дату последнего внесения в него изменений.

Когда программа отправляет запрос к операционной системе с требованием предоставить ей содержимое какого-либо файла, ОС просматривает запись каталога для него, чтобы найти первый кластер этого файла. Затем она обращается к элементу FAT для данного кластера, чтобы найти следующий кластер в цепочке. Повторяя этот процесс, пока не обнаружит последний кластер файла, ОС точно определяет, какие кластеры принадлежат данному файлу и в какой последовательности. Таким образом, система может предоставить программе любую часть запрашиваемого ею файла. Такой способ организации файла носит название цепочки FAT.

В системе FAT файлам всегда выделяется целое число кластеров. На 1,2-гигабайтном жестком диске с 32-килобайтными кластерами в каталоге может быть указано, что размер текстового файла, содержащего слова «Hello, world», составляет всего 12 байт, но на самом деле этот файл занимает 32 Кбайт дискового пространства. Неиспользованная часть кластера называется потерянным местом (slack). В небольших файлах почти весь кластер может быть потерянным местом, а в среднем потери составляют половину размера кластера.

Один из способов высвобождения пространства на диске — с помощью программ сжатия диска, например DriveSpace, которая выделяет «потерянные места» для использования другими файлами.

Расчеты размеров секторов и кластеров

Сначала в файловой системе FAT в DOS 1.0 использовались 12-разрядные элементы FAT. В DOS 2.0, оснащенной дополнительными средствами для работы с жесткими дисками, в FAT уже были предусмотрены 16-разрядные элементы. Последнее стало причиной проблемы, возникшей в 1987 г. Сущность ее заключалась в том, что в низкоуровневой команде считывания сектора диска под значение параметра с номером считываемого сектора было отведено 16 разрядов. Таким образом, максимальный объем внешнего устройства, с которым, в принципе, могла работать операционная система DOS 2.0, ограничивался 32 Мбайт (данный предел получается как результат умножения 65 536, максимального числа, укладывающегося в 16 разрядов, на 512 байт в секторе). В DOS 4.0 для обработки запросов на считывание секторов нижнего уровня используются 32-разрядные параметры.

Предел в 2 Гбайт на самом деле обусловлен не проблемами операционной системы, а существующими программами. На диске емкостью 2 Гбайт и более кластеры имеют размер 64 Кбайт, или 65 536 байт. Но самое большое значение, которое может быть представлено в 16 разрядах, составляет 65 535; 64 Кбайт — слишком большая величина для 16 разрядов. В Microsoft обнаружили, что при разработке многих существующих в настоящее время программ их авторы исходили из предположения, что значение числа байт в кластере умещается в 16 разрядах.

Ограничения на размеры секторов и кластеров в FAT16

Даже если такой проблемы не возникло при объеме в 2 Гбайт, то операционная система все равно не сможет работать с дисками емкостью свыше 4 Гбайт. При использовании FAT16 число секторов в кластере сохраняется самой ОС в одном байте блока параметров диска, устанавливаемого ОС. Это число должно быть кратным степени 2 и при этом меньше 256. Таким образом, в кластер может входить не более 128 секторов, или 64 Кбайт. В итоге, даже если ограничение в 2 Гбайт снимается, система FAT16 приемлема лишь для дисков емкостью менее 4 Гбайт.

В файловой системе FAT32 как элементы FAT, так и номера секторов 32-разрядные. Вот что это значит: умножим 4 294 967 296 различных 32-разрядных значений на 512 байт в секторе и получим огромное число 2 Тбайт (2 199 023 255 552 байт), которое представляет собой максимально возможную емкость диска при использовании FAT32.

Таблица 4.2. Ограничения на размеры секторов и кластеров

Размер диска	Число секторов в кластере		Размер кластера	
	Десятичное	Двоичное	Десятичное	Двоичное
Менее 128 Мбайт	4	00000100	2 Кбайт	0000100000000000
Менее 128 Мбайт	8	00001000	4 Кбайт	0001000000000000
Менее 128 Мбайт	16	00010000	8 Кбайт	0010000000000000
Менее 128 Мбайт	32	00100000	16 Кбайт	0100000000000000
Менее 128 Мбайт	64	01000000	23 Кбайт	1000000000000000
Менее 128 Мбайт	128	10000000	264Кбайт	Ошибка: проблема для существующих программ
Менее 128 Мбайт	256	Ошибка: проблема для ОС		

Некоторое устранение потерь

Теперь, когда допустимое число кластеров и число секторов одинаковы, видимо, можно было бы предположить, что для минимизации потерь места на диске в FAT32 будут использоваться кластеры, состоящие только из одного сектора. Но в этом вопросе необходим какой-то компромисс, поэтому в спецификации FAT32 размер кластера для накопителей емкостью до 8 Гбайт составляет 4 Кбайт.

Таблица 4.3. Зависимость размера кластера от емкости диска

Емкость диска, Гбайт	Размер кластера, Кбайт
Менее 8	4
Менее 16	8
Менее 32	16
32 и более	32

Одним из факторов, сыгравшим роль в принятии такого решения, стало то, что теперь нужно учитывать и размер самой FAT. Операционная система сохраняет две копии FAT, поэтому под элемент каждого кластера в FAT требуется 8 байт. На 2-гигабайтном диске FAT займет 32 Мбайт его пространства при размере кластера 512 байт. А если размер кластера составляет 4 Кбайт, для хранения двух таблиц FAT потребуется всего 4 Мбайт, то есть будет сэкономлено 28 Мбайт.

Но причиной для компромисса послужили не только соображения экономии дискового пространства; для ускорения доступа к диску встроенная программа дискового кэша VCACHE загружает всю FAT в оперативную память.

Очевидно, что этого проще добиться, если каждая отдельная таблица FAT занимает 2 Мбайт, а не 16 Мбайт. И не беспокойтесь о том, что увеличение числа кластеров и удвоение размера элементов FAT приведут к снижению эффективности VCACHE: для FAT32 предусмотрен новый метод работы с кэшем, который, согласно заявлениям Microsoft, повышает общую производительность.

Другие изменения в FAT32

Чтобы обеспечить возможность работы с возросшим числом кластеров, в записи каталога для каждого файла должно выделяться 4 байт для начального кластера файла (вместо 2 байт в системе FAT16). Традиционно каждая запись в каталоге состоит из 32 байт (см. рис. 4.1). В середине этой записи не используются 10 байт (байты с 12-го по 21-й), которые Microsoft зарезервировала для своих собственных нужд в будущем. Два из них теперь отводятся как дополнительные байты, необходимые для указания начального кластера в системе FAT32.

Операционная система всегда предусматривала наличие на диске двух экземпляров FAT, но использовался только один из них. С переходом к FAT32 операционная система может работать с любой из этих копий. Еще одно изменение состоит в том, что корневой каталог, раньше имевший фиксированный размер и строго определенное место на диске, теперь можно свободно наращивать по мере необходимости, подобно подкаталогу. Теперь не существует ограничений на число записей в корневом каталоге. Это особенно важно, поскольку под каждое длинное имя файла используется несколько записей каталога.

Следует помнить, что, поскольку поле для размера файла в записи каталога занимает 32 разряда, размер отдельного файла не может превышать 4 Гбайт.

4.3. Операционная система MS DOS

В этом параграфе мы остановимся на базовых принципах организации операционной системы MS DOS. Как уже упоминалось ранее, длительное время она явно «господствовала» на подавляющем большинстве персональных компьютеров. С наступлением эры графических операционных сред DOS отошла в тень. Однако ее роль в обеспечении процессов управления ЭВМ по-прежнему остается достаточно ощутимой. Более того, практика показывает, что знание основ DOS способствует более глубокому пониманию того, как функционирует программное обеспечение последующих поколений.

4.3.1. Основные составные части MS DOS

Перечислим основные составные части DOS:

1. Модуль расширения системы ввода/вывода (IO.SYS).
2. Базовый модуль операционной системы (MSDOS.SYS).
3. Командный процессор (или командная оболочка) (COMMAND.COM).
4. Внешние команды и драйверы, утилиты — файлы с расширением *.COM, *.EXE, *.SYS.

5. В отдельную составную часть также могут быть выделены инструментальные средства DOS (система программирования, текстовый редактор, отладчик DEBUG, реализующий простейшие функции тестирования и отладки программ).

В основу разделения DOS на перечисленные блоки был положен принцип модульности: разработчиками в идеологию системы была изначально заложена возможность относительно безболезненной замены одного блока при сохранении неизменными остальных. В частности, блок IO.SYS объединил процедуры ввода/вывода, которые являются наиболее зависимыми от конкретных аппаратных устройств персонального компьютера и потенциально должны быть более «гибкими», чем основывающиеся на них процедуры из базового модуля MSDOS.SYS.

Необходимо также сказать несколько слов о программном обеспечении, которое находится «ближе» к аппаратным ресурсам компьютера, чем DOS. Это *загрузчик* и *базовая система ввода/вывода*. Они записаны в так называемую постоянную память компьютера[1] и являются его неизменяемыми атрибутами (по крайней мере, в обычных условиях эксплуатации).

Загрузчик, как нетрудно догадаться по названию, является небольшой программой, осуществляющей инициализирующие действия компьютера сразу после включения питания.

BIOS — базовая система ввода/вывода (Basic Input/Output System) называется так потому, что включает в себя обширный набор модулей, благодаря которым операционная система и другие прикладные программы могут взаимодействовать с различными устройствами компьютера (дисплеем, клавиатурой, дисководом, принтером, таймером и т. д). Еще раз подчеркнем, что, с одной стороны, BIOS можно рассматривать как составную часть аппаратных средств, а с другой — она является программным обеспечением.

Большинство современных видеоадаптеров, а также контроллеры накопителей имеют собственную систему BIOS, которая обычно дополняет системную. Во многих случаях программы, входящие в конкретную BIOS, заменяют соответствующие программные модули основной BIOS. Вызов программ BIOS, как правило, организован в форме так называемой *обработки аппаратных и программных прерываний*.

ПРИМЕЧАНИЕ

Аппаратные прерывания (interrupt) — это события, инициируемые техническими устройствами (клавиатурой, мышью и т. п.), требующие программной обработки. *Программные прерывания* принципиально отличаются от аппаратных тем, что наступают по командам программ. По наступлении прерывания вызывается связанная с ним процедура-обработчик.

Выделение BIOS в отдельный модуль позволяет обеспечить независимость программного обеспечения от специфики конкретной модели компьютеров.

[1] ПЗУ (ROM) — постоянное запоминающее устройство, физически реализуется в виде специальных микросхем, в которые с помощью специального устройства — программатора — может записываться та или иная информация. В том числе программные модули.

Кратко перечислим вспомогательные функции BIOS:

- поиск на гибком и на жестком диске программы-загрузчика операционной системы и загрузка с диска в оперативную память;
- тестирование аппаратной части (в том числе оперативной памяти);
- инициализация векторов прерывания нижнего уровня.

BIOS содержит: специальные программы по управлению работой стандартными внешними устройствами (*драйверы* — программы, расширяющие возможности операционной системы); тестовые программы для контроля работоспособности аппаратуры; программу начальной загрузки операционной системы.

В качестве основных функций драйвера могут быть названы:

- прием и обработка запроса к периферийному устройству;
- преобразование запроса в серию команд управления устройством;
- обработка сигнала прерывания.

Драйверы могут быть:

1. *Стандартные* (*внутренние*) — программы, находящиеся внутри BIOS или его модуля расширения EM BIOS. Они подключаются к системе автоматически после перехода компьютера в рабочее состояние.
2. *Загружаемые* (*внешние*) — программы, находящиеся на диске и предназначенные для управления внешними устройствами, которые отличаются от стандартных своими техническими параметрами; подключаются к системе только тогда, когда они указаны в файле конфигурации CONFIG.SYS.

4.3.2. Командный процессор Command.com

Остановимся более подробно на той роли, которую играет в работе операционной системы командный процессор COMMAND.COM[1]. Выделение его в отдельный блок было обусловлено желанием разработчиков MS DOS предусмотреть возможность гибкой замены синтаксиса системы управляющих команд[2].

Взаимодействие пользователя с операционной системой осуществляется с помощью *командной строки*. В начале командной строки всегда имеется *приглашение* (к вводу команды) MS DOS. По умолчанию в качестве приглашения выводятся имена текущих диска (логического устройства) и каталога. Приглашение заканчивается символом >. В принципе, его внешний вид по желанию пользователя может быть изменен: в строке могут быть отображены текущее время и дата, путь, символы-разделители и т. п. Простейший вид командной строки с набранной командой (dir /w) показан на рис. 4.2.

---

[1] Вопросы организации и работы таких блоков, как IO.SYS и MSDOS.SYS, относятся к области программно-технологических решений и выходят за рамки данной книги.

[2] Справедливости ради, необходимо заметить, что данная возможность на практике применялась крайне редко.

```
C:\UTIL>dir /w
```

Рис. 4.2. Командная строка MS DOS

Работа в операционной системе MS DOS организуется с помощью команд, которые вызывают определенную последовательность действий компьютера.

Ввод команды осуществляется в командной строке в соответствии со следующими правилами:

- команда DOS состоит из *имени команды* (вводится латинскими буквами, регистр не имеет значения) и отделенных от него пробелом *параметров* (ключей, опций), уточняющих действия команды;
- формат параметров определяется исключительно внутренней логикой работы команды. В некоторых случаях они могут просто отсутствовать. Часто в качестве признака начала параметра (или для отделения параметров друг от друга) используются символы / или —. В состав параметров могут входить имена файлов.

Для того чтобы получить контекстную справочную информацию по формату команды, ее надо ввести с параметром / ?, как это показано на рис. 4.3:

```
C:\UTIL>copy /?
Копирование одного или нескольких файлов в другое место.

COPY [/A | /B] источник [/A | /B] [+ источник [/A | /B] [+ ...]] [результат
     [/A | /B]] [/V] [/Y | /-Y]

  источник     Имя копируемого файла (файлов).
  /A           Файл является текстовым файлом ASCII.
  /B           Файл является двоичным.
  результат    Папка для размещения и/или имя создаваемого файла.
  /V           Проверка правильности копирования путем сравнения файлов.
  /Y           Отключение режима запроса подтверждения на замену файлов.
  /-Y          Включение режима запроса подтверждения на замену файлов.

Ключ /Y может быть установлен с помощью переменной среды COPYCMD.
Ключ /-Y, заданный непосредственно в командной строке, имеет над ним приоритет.

Чтобы объединить несколько файлов в один, необходимо указать единственный
результирующий файл и несколько исходных (путем использования подстановочных
знаков или формата файл1+файл2+файл3).

C:\UTIL>_
```

Рис. 4.3. Получение справки по команде Copy

Для *ввода команды* следует:

- в соответствии с форматом в командной строке набрать имя и необходимые параметры команды;
- нажать клавишу ввода — Enter. При отсутствии ошибок в формате команды будет выполнено указанное действие, а при их наличии — выведено соответствующее сообщение.

Результат выполнения команды dir, запущенной с параметром /w, задающим вывод содержимого текущего каталога в несколько столбцов, показан на рис. 4.4.

```
C:\UTIL>dir /w

 Том в устройстве C имеет метку A_TRADE
 Серийный номер тома: 3A36-16EC
 Содержимое папки C:\UTIL

[.]             [..]            [CHECKIT]       [SEA]           [NU]
[3D]            [CCT386]        [GSPEED]        [HDD_TEST]      [CDTEST29]
[OLEG_A~1]
               0 файлов              0 байт
              11 папок     369 065 984 байт свободно

C:\UTIL>
```

Рис. 4.4. Результат выполнения команды dir /w

При вводе команды пользователем управление передается командному процессору Command.Com, первой функцией которого является синтаксический анализ.

Можно сформулировать несколько принципов классификации команд операционной системы MS DOS:

1) **по способу реализации**: резидентные (внутренние, встроенные); транзитные (внешние, утилиты, обслуживающие);
2) **по функциональному назначению**: команды для работы с каталогами, с файлами, с дисками, для управления памятью и устройствами, для конфигурирования системы и др.

Резидентные (встроенные) команды входят в состав командного процессора COMMAND.COM и после окончания загрузки операционной системы MS DOS располагаются в оперативной памяти, доступны в любой момент времени. В качестве примера встроенных команд могут быть названы:

- DIR — выводит информацию о содержании каталогов логических устройств;
- COPY — выполняет операции копирования файлов и каталогов;
- DEL — выполняет операции удаления файлов и каталогов;
- REN — выполняет операции по переименованию файлов и каталогов;
- CLS — очищает содержимое экрана;
- DATE — вывод/корректировка текущей даты;
- TIME — вывод/корректировка текущего времени;
- PATH — вывод/задание списка каталогов, в котором (последовательно) будет искаться файл с запускаемой программой, если он не найден в текущем каталоге и явно не задано его полное имя.

Транзитные команды (утилиты) реализуются в виде файлов с расширениями *.COM или *.EXE. С точки зрения процедуры запуска они ничем не отличаются от любых других служебных и прикладных программ, не входящих в операционную систему.

Идеология реализации основной части функций операционной системы через внешние утилиты, заложенная в MS DOS, позволила на начальном этапе развития обеспечить исключительную компактность ее ядра, что было очень важно для первых моделей персональных компьютеров, ограниченных по аппаратным возможностям. В дальнейшем это обеспечило широкие возможности по наращиванию функциональности системы исключительно за счет добавления все новых и новых утилит. Одновременно такая открытость в организации MS DOS очень быстро сформировала развитой рынок дополнительного сервисного программного обеспечения, предлагаемого независимыми по отношению к Microsoft разработчиками.

Среди наиболее «популярных» DOS-утилит могут быть названы:

- FORMAT — программа форматирования внешних запоминающих устройств;
- SCANDISK — служебная программа проверки состояния дисков;
- FDISK — утилита для разбиения жестких дисков на логические разделы;
- XCOPY — утилита, реализующая расширенные операции копирования файлов и каталогов.

Особенностью MS DOS является то, что любая строка, не распознанная синтаксическим анализатором в качестве внутренней команды, считается именем файла, содержащим программу (служебную или прикладную), которую следует выполнить. Если такой файл не удается найти, то выводится сообщение

```
Имя команды или файла указано неправильно
```

или в англоязычных версиях

```
Bad command or file name
```

Правила формирования имен файлов были рассмотрены в 4.2.

Таблица 4.4 содержит стандартные типы расширений имен файлов, часто встречающихся в MS DOS и DOS-совместимых операционных системах.

Таблица 4.4. Стандартные типы файлов и их расширения

Расширение	Тип файла
*.COM	Исполняемый файл (компактный файл, размером до 64 Кбайт)
*.EXE	Исполняемый файл
*.BAT	Командные (пакетные) файлы, см. 4.3.3
*.DAT	Файл с данными
*.HEX	Файл с данными, представляемыми в шестнадцатеричном формате
*.OVR	Файл с оверлейными (догружаемыми в процессе выполнения) частями программ
*.DLL	Файл с динамически компонуемыми библиотеками (Dynamic Linked Libraries). Содержит программные модули, общие для различных задач в формате MS Windows, которые ими вызываются в процессе выполнения
*.BAK	Резервная копия файла, сохраняемая перед началом каких-либо операций с ним (от англ. backup)

Расширение	Тип файла
*.SYS	Системный файл
*.TMP	Временный файл
*.TXT	Текстовый файл
*.DOC	Файл с документом (допустим в формате текстового процессора MS Word)
*.BAS	Файл с текстом программы на языке BASIC
*.PAS	Файл с текстом программы на языке Паскаль (Pascal)
*.C	Файл с текстом программы на языке C
*.OBJ	Объектный модуль — скомпилированная программная процедура, переведенная в машинные коды
*.LIB	Библиотека — объединенная в единый файл специального формата совокупность объектных файлов

4.3.3. Создание командных файлов

Достаточно часто при работе с операционной системой возникает необходимость организации автоматического выполнения целого ряда команд (одной за другой), или, как еще говорят, пакетного выполнения. Средством пакетной обработки является *командный* (*пакетный*) *файл*. Он облегчает работу пользователя за счет автоматизации выполнения часто повторяющейся совокупности команд и освобождает его от необходимости запоминания многих деталей и тонкостей форматов команд.

Командные файлы, работающие в операционной среде MS DOS, имеют расширение *.BAT (от английского *batch* – пачка).

Командный файл с точки зрения внутреннего содержания представляет собой текстовый файл и может быть создан в любом текстовом редакторе. Его текст состоит из последовательности команд управления операционной системой, которые, вообще говоря, могли быть введены пользователем с клавиатуры. Помимо этого в командном файле могут содержаться специальные операторы, накладывающие простейшие логические ограничения на процесс выполнения. Более подробно ознакомиться с синтаксисом языка командных файлов можно в **Руководстве пользователя по операционной системе DOS**.

При создании командного файла необходимо соблюдать некоторые правила:

- командный файл может состоять из любых команд операционной системы MS DOS, имен файлов запуска и сервисных программных средств;
- каждая команда располагается на отдельной строке (при вводе команд могут использоваться строчные и прописные буквы);
- после ввода каждой команды нужно нажать клавишу Enter;
- имя командного файла уникально в пределах каталога, где он находится.

Запуск командного файла: в командной строке задается только имя файла (без указания типа) и нажимается клавиша ввода – Enter.

Для прерывания работы командного файла могут быть использованы комбинации клавиш Ctrl+C или Ctrl+Break.

4.3.4. Командный файл автонастройки AUTOEXEC.BAT

Файл AUTOEXEC.BAT играет особую роль в операционной системе MS DOS: являясь по содержанию обычным командным файлом, он автоматически выполняется при каждой загрузке операционной системы[1]. Его основное назначение состоит в сохранении и оперативной установке значений параметров операционной среды для конкретного пользователя конкретного компьютера.

Среди команд, которые особенно часто используются в составе AUTOEXEC.BAT, могут быть перечислены:

- REM — комментарии; неисполняемая команда, служит для пояснения действий;
- PAUSE — прерывание выполнения командного файла до тех пор, пока не будет нажата любая клавиша;
- CLS — очистка экрана;
- ECHO — вывод сообщений на экран;
- ECHO OFF — подавляет вывод на экран всех стоящих после нее команд;
- ECHO ON — поддерживает вывод на экран всех стоящих после нее команд;
- @ — запрет на индикацию одной команды, перед которой стоит этот символ;
- VERIFY ON или OFF — установка (ON) / отключение (OFF) режима контрольного чтения при операциях записи файлов (если данный режим включен, то он позволяет выявлять возможные ошибки, но, естественно, замедляет работу);
- PATH [Путь] — задание последовательности каталогов, в которых будут последовательно искаться исполняемые файлы, если пользователь при запуске не задает их полное имя;
- PROMPT [параметры] — установка формата приглашения в командной строке DOS (по умолчанию используется текущее логическое устройство и символ >, например, C:\>);
- SET <Имя> = <Значение> — устанавливает имя и значение так называемой переменной системной среды.

ПРИМЕЧАНИЕ

Переменные среды (enviroment variables) в операционных системах серии DOS являлись простейшим способом передачи внешней управляющей информации программам (со стороны ОС или других программ), или, другими словами, задавали среду их работы.

К одной из главных функций файла автозагрузки относится запуск так называемых *резидентных программ*, которые после выполнения должны остаться в памяти машины для реализации тех или иных служебных функций.

[1] Подчеркнем, что здесь имеется в виду файл с именем AUTOEXEC.BAT, расположенный в корневом каталоге того диска, с которого загружается операционная система.

Очень часто из AUTOEXEC.BAT запускаются:

- антивирусные программы (например, VSAFE — резидентная команда слежения за появлением вирусов);
- программы-оболочки (Norton Commander);
- драйверы клавиатуры, принтера (для печати русских букв) и т. п.

Пример возможного содержания файла AUTOEXEC.BAT представлен на рис. 4.5.

```
REM отключить вывод команд на экран
@ECHO OFF
REM задать переменную среды, сохраняющую имя каталога
REM для временных файлов
SET TEMP=C:\TEMP
REM задать путь
PATH C:\DOS;C:\;C:\ARCH;C:\NC;C:\UTIL\NU
REM задание номера кодовой страницы для таблицы символов
MODE CON CODEPAGE PREPARE=((866) C:\DOS\EGA.CPI)
CHCP 866
REM загрузка драйвера руссификации клавиатуры
LH KEYB RU,,C:\DOS\KEYBOARD.SYS
REM загрузка прочих драйверов: кэширования жесткого диска, мыши, CD-ROM
LH C:\DOS\SMARTDRV.EXE /X
LH C:\DOS\MOUSE.COM
LH C:\CDROMDRV\MSCDEX /V /D:CD003 /M:10
REM запустить Norton Commander
NC
```

Рис. 4.5. Пример файла AUTOEXEC.BAT

4.3.5. Файл конфигурации CONFIG.SYS

Файл CONFIG.SYS предназначен для настройки операционной системы на конкретную конфигурацию аппаратуры компьютера. Так же, как и AUTOEXEC.BAT, он присутствует практически на любом компьютере в главном каталоге на диске, с которого производится загрузка операционной системы.

Среди основных отличительных особенностей CONFIG.SYS могут быть названы следующие:

- как правило, задаваемые в нем настройки действуют в течение всего сеанса работы (то есть могут быть изменены только после перезагрузки компьютера);
- настройки CONFIG.SYS устанавливаются в автоматическом режиме перед выполнением AUTOEXEC.BAT.

Основное назначение CONFIG.SYS состоит в организации загрузки в оперативную память необходимых драйверов для управления аппаратной частью компьютера (подключение различных видов памяти, клавиатуры, мыши, принтера).

Начиная с MS DOS версии 6.0, в CONFIG.SYS появилась возможность создания описаний различных вариантов загрузки, из которых пользователь имеет возможность в режиме диалога выбирать нужный на текущий сеанс работы.

CONFIG.SYS создается и редактируется в любом текстовом редакторе, при этом необходимо соблюдать некоторые правила:

- файл CONFIG.SYS состоит из специальных команд операционной системы MS DOS по настройке аппаратуры. Формат этих команд имеет особый вид: Имя команды = значение;
- каждая команда располагается на отдельной строке (при вводе команд могут использоваться строчные и прописные буквы);
- после ввода каждой команды нужно нажать клавишу Enter;
- порядок записи команд DEVICE и DEVICEHIGH в файле CONFIG.SYS **очень важен**: сначала устанавливаются драйверы, позволяющие работать с разными видами памяти, которые дополняют стандартную память объемом 640 Кбайт, а затем устанавливаются драйверы, использующие эту память;
- после создания (редактирования) файла CONFIG.SYS необходимо перезагрузить операционную систему для установки новой конфигурации настройки.

Перечислим «часто используемые» команды и устанавливаемые драйверы в файле CONFIG.SYS:

BREAK = ON или OFF — установка/отмена режима возможности прерывания процесса выполнения программ по нажатию комбинации клавиш Ctrl+Break (или Ctrl+C);

BUFFERS = <число> — выделение объема оперативной памяти в соответствии с заданным в команде числом буферов (ячеек стандартной длины). Обычно рекомендуется использовать не менее 40 буферов;

DEVICE = <имя драйвера> [параметры]> — подключение драйвера для управления каким-либо устройством;

DEVICEHIGH = <имя драйвера> [параметры] — загружает драйвер устройства в верхнюю память (UMB)[1];

DOS = <параметры> — устанавливает режим использования области высокой памяти (HMA) и обеспечения доступа к блокам верхней памяти (UMB);

FILES = <число> — устанавливает максимальное число одновременно открытых файлов. Обычно не менее 20;

NUMLOCK = ON или OFF — устанавливает состояние клавиши NumLock;

REM или ; — комментарии;

LASTDRIVE — установка количества букв логических дисков;

Драйверы, входящие в поставку операционной системы MS DOS:

ANSI.SYS — поддерживает эмуляцию терминала ANSI;

DISPLAY.SYS — поддерживает переключение кодовых страниц для монитора;

HIMEM.SYS — управляет использованием расширенной (Extended) памяти;

EMM386.EXE — организует расширенную (Extended) память и обеспечивает доступ к области верхней памяти на компьютерах с процессором 386 и выше, которые имеют расширенную (Extended) память.

Пример возможного содержания файла CONFIG.SYS представлен на рис. 4.6.

[1] В задачи данной книги не входит раскрытие таких специальных технических понятий, как высокая (HMA), верхняя (UMB) и расширенная (Extended) память, в связи с чем можем отослать заинтересованного читателя к специальным источникам.

```
DEVICE=C:\DOS\HIMEM.SYS /TESTMEM:OFF
DEVICE=C:\DOS\EMM386.EXE RAM
COUNTRY=007,866 C:\DOS\COUNTRY.SYS
DEVICEHIGH=C:\DOS\DISPLAY.SYS CON=(EGA,,1)
INSTALL=C:\DOS\NLSFUNC.EXE C:\DOS\COUNTRY.SYS
DEVICE=C:\CDROMDRV\CYBIDE.SYS /D:CD003
BUFFERS=30
FILES=40
SWITCHES=/F
DOS=HIGH,UMB
BREAK=ON
LASTDRIVE=M
```

Рис. 4.6. Пример файла CONFIG.SYS

4.3.6. Программные оболочки

Управление компьютером при помощи команд DOS требует определенной квалификации, большой аккуратности и внимания. Для того чтобы сделать общение с компьютером более простым, используются специальные *программы-оболочки*. Оболочка — это такая программа, которая позволяет пользователю осуществлять действия по управлению ресурсами компьютера в рамках более развитого (более удобного и интуитивно понятного) интерфейса. К числу наиболее популярных оболочек относится пакет Norton Commander (фирма Symantec).

Применение операционной оболочки Norton Commander значительно упрощает управление компьютером, позволяет в наглядном виде получать информацию о его основных ресурсах (и их загруженности), осуществлять все основные процедуры управления компьютером (выбор диска, каталога; создание каталога; создание, просмотр, редактирование текстовых файлов, копирование, перемещение, удаление файлов и каталогов; поиск файлов и каталогов; работа с архивными файлами; сравнение каталогов и т. п.).

Общий вид интерфейса Norton Commander, который зачастую знаком даже людям, весьма далеким от компьютеров и программ, представлен на рис. 4.7.

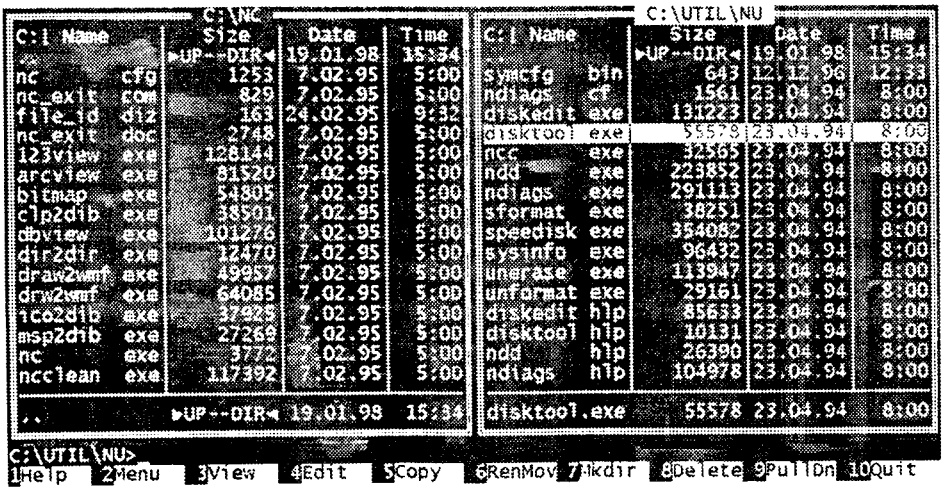


Рис. 4.7. Norton Commander, интерфейс

Вопрос о том, какие именно качества Norton Commander обеспечили данной программной оболочке столь грандиозную популярность среди пользователей, не получил однозначного ответа даже к настоящему времени, когда, в общем-то, ее время уже прошло. По мнению авторов, на первое место среди прочих достоинств Norton'a следует поставить идею двухпанельного интерфейса, предоставляющего пользователю возможности по одновременной манипуляции содержимым нескольких логических устройств. Действительно, однопанельные оболочки, очень напоминавшие одну отдельно взятую панель Norton'a, существовали и до него, однако все они в подавляющем большинстве стали достоянием истории, не оставив о себе сколько-нибудь значительных воспоминаний.

В завершение следует сказать, что широкая популярность данной оболочки у пользователей привела к феномену появления программ, «похожих на Norton». Их авторы, оставаясь в фарватере основной мультипанельной идеи, вносили определенные усовершенствования (как правило, сервисного характера). Иногда данные усовершенствования оказывались непринципиальными экзотическими излишествами, а иногда давали пользователям ощутимый эффект в работе. Среди таких «клонов» Norton Commander могут быть названы Volkov Commander (VC), DOS Navigator (DN), Far (File and archive manager).

Также необходимо отметить, что Norton Commander явился не единственным шагом фирмы Symantec по расширению функциональных возможностей служебного и системного программного обеспечения, работающего в среде DOS. Другой ее известный продукт — Norton Utilities — объединил в себе большое количество утилит, реализующих многие важные и полезные функции, которые затруднительно или даже невозможно осуществить с помощью штатных средств операционной системы.

4.4. Операционные системы Windows

4.4.1. Общая характеристика и история развития

Дальнейшим шагом в развитии оболочек операционных систем стало появление в 1986 г. графической многооконной операционной оболочки Windows фирмы Microsoft. После своего возникновения она пережила ряд модификаций, не все из них были удачными. Однако к завершению 1991 г. вышла версия Windows 3.1 и несколько позже — сетевой вариант Windows 3.11 (Windows 3.11 For WorkGroups), завоевавшие широкое признание миллионов пользователей. Дальнейший гомологический ряд Windows-продуктов продолжили высокопроизводительные многозадачные, полнофункциональные операционные системы с графическим интерфейсом Windows 95, затем Windows 98 и Windows 2000.

Кратко перечислим основные характеристические черты программных продуктов серии Windows:

- ключевой идеей Windows является обеспечение полной независимости программ от аппаратной части компьютера — программная совместимость;
- единый графический пользовательский интерфейс;

- многозадачность;
- возможность работы в сетевой среде;
- наличие универсальной системы средств обмена данными между приложениями.

Графика в Windows также является универсальной. Таким образом снимается проблема обеспечения совместимости с конкретным типом дисплея или принтера.

4.4.2. Windows 3.1

Windows 3.1 — популярная графическая операционная оболочка, которая запускается на выполнение, как обычная программа MS DOS. Windows работает на базе MS DOS, используя на нижнем уровне внутренние функции и процедуры данной операционной системы.

Принципиальным условием для программных приложений, предназначенных для работы в среде Windows, является то, что они должны работать с внешними устройствами (монитором, принтером, плоттером и т. п.) не напрямую, а через универсальную систему команд. Управляющая система транслирует вызовы (обращения к тому или иному физическому устройству) и передает их соответствующему Windows-драйверу данного устройства, который непосредственно отвечает за работу с ним с учетом конкретных особенностей его работы. Почти все драйверы устройств Windows 3.1 фактически выполняют функции BIOS, работают с устройствами напрямую.

Пользовательский интерфейс Windows 3.1 является графическим (использует графический режим работы видеомонитора). Его основу составляет иерархически организованная система окон и других графических объектов. *Окно* — структурный и управляющий элемент пользовательского интерфейса, который представляет собой обрамленную часть экрана, в котором может отображаться приложение, документ или сообщение.

Пользовательский интерфейс Windows (в отличие от интерфейсов командной строки в DOS и оболочки Norton Commander) реализует оперативное управление на основе выбора того или иного графически визуализированного элемента (кнопки, пиктограммы, списка и т. п.) с помощью манипулятора мышь (команды клавиатуры, как правило, имеют вспомогательное или резервное значение). Набор используемых элементов интерфейса стандартен, что позволяет легко осваивать интерфейс прикладных программ, работающих под Windows.

Windows — объектно-ориентированная система. Общую суть объектно-ориентированного программирования можно сформулировать следующим образом: не программы управляют данными, а данные управляют программами. Документ, если его рассматривать с позиций объектно-ориентированного программирования, вряд ли можно считать объектом, однако в Windows посредством самостоятельного механизма связи можно определить, каким приложением документ будет обрабатываться (аналогичный механизм есть и в Norton Commander). Реализация данного подхода в Windows соответствует теории объектно-ориентированного управления и освобождает пользователя от необходимости помнить, каким

именно приложением должен обрабатываться документ; достаточно дважды щелкнуть на значке документа или файла, и соответствующее приложение запускается, загружая этот документ в свое рабочее окно.

Другой важной характеристикой Windows как многозадачной среды явилась реализация в ней технологий обмена данными между различными приложениями (причем сразу на нескольких уровнях). К ним относятся:

- передача данных через *буфер обмена* (Clipboard);
- DDE (Dynamic Data Exchange);
- OLE (Object Linking and Embedding).

Буфер обмена — это специальным образом организованное динамическое пространство оперативной памяти для временного размещения данных, при этом в нем запоминаются как сами данные, так и то, к какому программному приложению они относятся. Каждое последующее занесение в буфер информации уничтожает предыдущее. Обмен данными как внутри программ Windows, так и между ними построен на базе универсальных системных процедур:

- для копирования/переноса данных в буфер обмена используются команды Копировать/Вырезать (они доступны из любых приложений по нажатии комбинаций клавиш Ctrl+Ins и Shift+Del);
- для вставки (копирования в приложение-приемник) данных из буфера обмена используется команда Вставить (вызывается комбинацией клавиш Shift+Ins).

Для того чтобы непосредственно работать с данными, хранящимися в буфере обмена, нужно использовать окно Главная ▸ Папка обмена. Это приложение позволяет просмотреть содержимое буфера обмена, сохранить его в файле и, наконец, очистить буфер. Очередной буфер получает название страницы в Папке обмена. Пользователь может самостоятельно присвоить ему имя и сохранить в файле с расширением *.CLP.

Технология DDE (Dynamic Data Exchange) представляет собой набор системных процедур, позволяющих обращаться из одного приложения (DDE-клиента) в процессе его выполнения к другому, активному на тот момент программному приложению (DDE-серверу). По запросу клиента сервер обрабатывает так называемый DDE-запрос и возвращает результаты в той или иной форме.

Наконец, еще одним методом организации взаимодействия программ в среде Windows является технология OLE (Object Linking and Embedding) — связывание и погружение объектов. Она подразумевает внедрение данных одного типа (обрабатываемых одной программой) в данные, относящиеся к другой программе. При этом при обращении к внедренным данным (допустим, по щелчку мыши) происходит автоматический вызов того приложения, к которому они относятся. Классическим примером применения технологии OLE является внедрение рисунка или электронной таблицы в текстовый документ.

Еще одним решением, имевшим принципиальный характер, явилось создание в Windows единой системы хранения конфигурационных настроек самой системы и установленных в ее рамках прикладных программ. В Windows 3.* для этого ис-

пользовались организованные специальным образом текстовые файлы с расширением *.INI.

Работа Windows в сетевой среде (версия Windows 3.11 для рабочих групп) обеспечивает пользователям сети ряд преимуществ, типичных при объединении компьютеров в сеть:

- совместное использование ресурсов файлового сервера, принтеров, факс-модемов;
- использование электронной почты и других средств коммуникации.

4.4.3. Windows 95

Windows 95 — высокопроизводительная, многозадачная и многопотоковая 32-разрядная операционная система с графическим интерфейсом и расширенными сетевыми возможностями, работающая в защищенном режиме, поддерживающая 16-разрядные приложения без всякой их модификации. Это интегрированная среда, обеспечивающая эффективный обмен текстовой, графической, звуковой и видеоинформацией между отдельными программами.

Функциональные возможности Windows 95 на качественном уровне превосходят все то, что было заложено в MS DOS и Windows 3.*.

Windows 95 — полномасштабная операционная система (семейства Windows), не требующая MS DOS[1]. Она полностью совместима с используемыми в настоящее время программными и аппаратными средствами.

Windows 95 — первый представитель нового поколения 32-битовых многопотоковых операционных систем.

Перечислим основные преимущества Windows 95:

- интегрированная операционная система (операционная система, ядро которой, загружаемое в момент включения компьютера, активизирует графический интерфейс пользователя и обеспечивает полную совместимость с операционной системой MS DOS);
- вытесняющая многозадачность (свойство операционной системы самостоятельно в зависимости от внутренней ситуации передавать или забирать управление у того или иного приложения, не позволяющее одному приложению занять все аппаратные ресурсы);
- многопоточность (свойство операционной системы выполнять операции одновременно над потоками нескольких 32-битовых приложений, называемых процессами).

Windows 95 использует технологию Plug and Play[2], упрощая работу с компьютером за счет следующих сервисных функций:

---

[1] Точнее говоря, скрывающая от пользователя обращения к функциям DOS.

[2] Дословный перевод с английского «подключи и играй».

- помощи при распознавании устройств для их установки и настройки;
- динамического изменения состояния системы и автоматического уведомления об этом программных приложений;
- интеграции драйверов устройств, системных компонентов и пользовательского интерфейса.

Windows 95 обеспечивает динамическое изменение конфигурации системы, построенной на базе технологии Plug and Play.

Технология Plug and Play, заложенная в Windows 95, позволяет работать с устройствами, не подчиняющимися спецификации Plug and Play, упрощая их настройку и управление оборудованием.

Для корректного обращения с системными ресурсами компьютера Windows 95 отслеживает все устройства и выделяемые им ресурсы. Диспетчер устройств позволяет получить информацию о всех найденных системой устройствах и изменить при необходимости их конфигурацию.

Windows 95 – высокоэффективная платформа для мультимедиа, включает в себя лазерный проигрыватель или CD-плейер; обеспечивает поддержку видеодисков и видеомагнитофонов и т. п.

Кроме этого Windows 95 имеет особые возможности, предназначенные для пользователей с ограниченными возможностями (недостатки слуха, зрения и др.):

- возможности масштабирования элементов интерфейса;
- «залипающие» клавиши;
- режим MouseKeys (все действия с мышью выполняются через клавиатуру);
- визуальное дублирование звуков системы.

Наконец, в Windows 95 вместо разрозненных INI-файлов для хранения конфигурации как самой операционной системы, так и других использующихся в ее рамках программ стала использоваться единая информационная база, получившая название *системного реестра* (Registry). Данное решение позволило централизовать и жестко упорядочить процесс управления со стороны операционной системы настройками установленного программного обеспечения.

4.4.4. Windows 98

По сравнению с Windows 95, Windows 98 включает средства, позволяющие компьютеру работать быстрее без добавления нового оборудования. В состав Windows 98 входит ряд программ, совместное применение которых повышает производительность компьютера

В Windows 98 надежность компьютера повышается за счет применения новых мастеров (мастер обслуживания позволяет быстрее выполнять программы, проверять жесткий диск на наличие ошибок и освобождать место на дис-

ке), служебных программ и ресурсов, обеспечивающих бесперебойную работу системы:

1. Проверка системных файлов позволяет отслеживать состояние наиболее важных файлов, обеспечивающих работу компьютера. Если эти файлы повреждены или перемещены, программа проверки их восстанавливает.
2. Проверка диска запускается автоматически после неверного выключения операционной системы. Программа проверки диска обнаруживает наиболее вероятные повреждения файлов и папок и выполняет исправление ошибок. Кроме того, пользователь имеет возможность выполнить проверку диска в любое время.
3. Новый Web-узел ресурсов Microsoft Windows Update автоматизирует процесс обновления драйверов и системных файлов и обеспечивает новейшие возможности технической поддержки. На узле Windows Update можно выполнить обзор имеющихся на компьютере драйверов устройств и системных программ, сравнить его результаты с основной базой данных в Web и после этого получить рекомендации и установить обновления, подходящие именно для данного компьютера. Средство удаления установки позволяет вернуться к предыдущим драйверам или системным программам.
4. Проверка реестра является системной программой, позволяющей обнаруживать и устранять ошибки в реестре. При каждом запуске компьютера программа проверки реестра автоматически проверяет реестр на наличие несогласованности структуры данных. Кроме того, программа проверки реестра каждый день выполняет резервирование реестра. Если обнаруживаются серьезные ошибки в реестре, реестр можно восстановить по резервной копии. Программа проверки реестра поддерживает до пяти сжатых архивных копий реестра, при которых компьютер успешно запускался. Если архив не удается обнаружить, программа проверки реестра исправляет ошибки реестра.
5. Программа установки автоматически запускает проверку реестра при каждом обновлении операционной системы компьютера. При установке Windows 98 программа проверки реестра исправляет большинство ошибок в реестре, даже те, о которых было неизвестно пользователю.
6. Программа архивации предоставляет расширенные возможности архивации и восстановления данных, в том числе поддержку большего числа накопителей на магнитной ленте и самого современного оборудования. Пользователю становится легче сохранять важные данные. Файлы с жесткого диска можно резервировать на гибких дисках, на магнитной ленте или на другом компьютере в сети. Если исходные файлы повреждены или потеряны, их можно восстановить из архива.

Использование мастера специальных возможностей. Специальные возможности (такие как залипание клавиш, субтитры и управление указателем с клавиатуры) помогают полностью использовать компьютер пользователям с физическими недостатками. Некоторые из этих средств, такие как управление указателем с клавиатуры, могут заинтересовать и более широкий круг пользователей.

Обозреватель Интернета[1]. Internet Explorer делает ряд функций доступными с рабочего стола Windows: возможность поиска в Web из любого места на компьютере, каналы Web-узлов на рабочем столе, подписка на избранные узлы, настраиваемая панель ссылок, панели обозревателя, а также контроллер содержимого и зоны безопасности при поиске в Web.

Проводник Windows 98 и Internet Explorer позволяют объединить ресурсы Web в едином представлении.

Windows 98 делает наиболее продуктивным использование Web за счет применения всех возможностей компьютера к интерактивному содержимому Интернета:

- автоматическое дополнение ранее вызывавшихся адресов Web по мере их ввода;
- улучшенные списки часто посещаемых Web-узлов;
- улучшенный журнал и возможности отслеживания посещаемых Web-узлов;
- поддержка всех основных стандартов Интернета, в том числе ActiveX[2], Java и др.;
- высокая производительность динамического языка HTML, что позволяет сделать Web-страницы более богатыми и интересными;
- новый мастер подключения к Интернету помогает зарегистрироваться для доступа к Интернету и автоматически выполняет шаги по настройке программного обеспечения, необходимые для доступа к Интернету.

Приложение Internet Explorer объединяет рабочий стол с Web, благодаря чему рабочий стол и его папки будут выглядеть и действовать так же, как при работе с Web. Точно так же, как вы просматриваете содержимое Web, можно будет просматривать содержимое компьютера, и при этом просмотр Web станет возможным из любого места в компьютере (везде можно найти меню Переход и Избранное). Кроме того, на рабочий стол, в папки и на панель задач можно будет добавлять «активное содержимое», такое как бегущая строка новостей.

4.4.5. Особенности операционной системы Windows 2000

Перечислим главные из новых технологических решений, реализованных в Windows 2000:

1. **Защита данных средствами** Kerberos. Kerberos — это разработанный в Массачусетском технологическом институте (MIT) протокол аутентификации, обеспечивающий передачу данных по незащищенным сетям. Именно он наряду со службой каталогов Active Directory (AD) составляет важнейшую особенность Windows 2000, отличающую эту систему от Windows NT 4.0. Kerberos имеет ряд преимуществ по сравнению с NT LAN Manager (NTLM), протоколом

[1] Изучению технологий и проблем применения Интернета и глобальных компьютерных сетей посвящена глава 11.

[2] Элементы ActiveX явились еще одним этапом реализации принципов объектно-ориентированного программирования и технологии OLE. Однако подробное раскрытие вопросов, связанных с их применением, выходит за рамки данной книги.

аутентификации Windows NT 4.0. Одно из них состоит в том, что Kerberos обеспечивает создание транзитивных доверительных отношений, тогда как NT 4.0 позволяет формировать лишь нетранзитивные.

2. **Инфраструктура открытых ключей**[1]. Операционная система Windows 2000 оснащается средствами защиты информации на базе инфраструктуры открытых ключей (public key infrastructure, PKI). Эта инфраструктура представляет собой систему цифровых сертификатов и организаций, удостоверяющих сертификаты (Certificate Authorities, CAs). Так же, как и протокол Kerberos, она дает возможность обоим участникам транзакции проверять, действительно ли партнер является тем, за кого себя выдает, а также шифровать транзакции. Система защиты данных на базе PKI лучше всего подходит для использования в среде Интернет и потому представляет собой полезное средство для организации электронной коммерции между предприятиями. Почтовое ведомство США (U.S. Postal Service) использует PKI для предоставления клиентам круглосуточного доступа по каналам Интернета к информации о почтовых расходах. Инфраструктура открытых ключей еще не вступила в пору зрелости, но ее включение в систему Windows 2000 дает основания полагать, что уровни безопасности и шифрование данных станут неотъемлемой частью повседневной коммерческой практики, а это, в свою очередь, откроет перед электронной торговлей новые горизонты.
3. **Windows 2000 оснащается рядом средств, предназначенных для пользователей малых офисов**. В состав системы входит вполне качественный маршрутизатор, обладающий интерфейсом подключения по запросу (demand-dial interface), что значительно облегчает процесс установки соединения с Интернет-провайдером. С помощью таких средств, как реализованный в Windows 2000 Server и в Windows 2000 Professional модуль Internet Connection Sharing, доступ к предоставляемому единому системой выходу в Интернет получают сразу несколько пользователей сети. В данном модуле реализована технология трансляции сетевых адресов Network Address Translation (NAT), которая дает возможность компьютерам, объединенным в локальную Windows-сеть, выходить через маршрутизатор в Интернет, используя единственный IP-адрес, предоставленный Интернет-провайдером[2]. В итоге множество пользователей локальной сети являются, с точки зрения провайдера, одним (но весьма активным) клиентом.
4. **Windows 2000 оснащена усовершенствованными средствами симметричной многопроцессорной обработки**.
5. **Встроенные средства удаленного доступа**.
6. **Новые функции управления хранением данных** (управление квотами дискового пространства, иерархическая система хранения данных, динамическое управление томами, а также путем внесения изменений в файловую систему NTFS).

---

[1] Более подробно вопросы организации защиты информации будут рассмотрены в главе 11, при изучении технологий работы с глобальными вычислительными сетями.

[2] Более подробно о понятии «IP-адрес» будет рассказано в главах 10 и 11.

7. **Дополнительные возможности для мобильных пользователей.** Реализованная в Windows 2000 функция работы с файлами в автономном режиме (offline files feature) позволяет отбирать сетевые файлы и папки для последующей работы с ними без подключения к сети. Затем ОС создает локальный кэш и сохраняет в нем эти файлы и папки, причем они синхронизируются с оригиналами в фоновом режиме. Кроме того, в Windows 2000 предусмотрена функция перехода в режим «спячки», перед отключением питания записывающая в специальный файл все содержимое физической памяти, что обеспечит тем самым сохранность всех данных о состоянии системы. В режиме ожидания система практически не потребляет энергии; переход в это состояние занимает порядка 15 с, а возвращение к рабочему режиму — 45 с (для системы с процессором Pentium II 366 МГц).
8. **Распределенное административное управление операционной среды.** Windows 2000 оснащена целым рядом средств управления клиентами и серверами, позволяющих снизить общую стоимость эксплуатации операционной системы.

4.4.6. Основные элементы графической оболочки Windows

Задача интерфейса — сделать компьютер доступнее, понятнее для пользователей. Из этих соображений для наименования всего пространства экрана в среде Windows используется понятие *рабочий стол* (DeskTop). Видимо, по замыслу разработчиков, общий вид монитора должен интуитивно ассоциироваться у пользователей с видом обычного рабочего стола, на котором располагаются папки с документами. Типичный вид Windows-интерфейса показан на рис. 4.8.

Чтобы выбрать для рабочего стола цвет и фоновый рисунок, нужно поместить указатель в любое пустое место на нем, нажать правую кнопку мыши и выбрать в контекстном меню (нажатие правой кнопки мыши отображает контекстное меню выделенного объекта, это меню содержит список пунктов для работы с объектом) пункт Свойства.

При работе на экране монитора, как на поверхности рабочего стола, можно размещать различные элементы, перемещать и удалять их, закрывать один элемент другим. Каждый элемент рабочего стола может иметь три представления: значок (свернутое окно), нормальное окно с обрамлением, полноэкранное окно.

Расположенные на рабочем столе **ярлыки** какого-либо объекта (файл, программы, сетевая папки, диск и т. п.) представляют собой высокоэффективное средство ускоренного доступа к объектам: двойной щелчок по ярлыку открывает объект, с которым он связан.

В нижней части рабочего стола находится *панель задач*. Она содержит кнопку Пуск, которая предназначена для быстрого запуска программ и поиска файлов, а также обеспечивает доступ к справке.

При открытии программы, документа или окна на панели задач появляется соответствующая открытому окну кнопка. Нажатие этой кнопки позволяет быстро перейти в выбранное окно.

На панели задач также расположены часы (возможна корректировка текущего времени и даты) и индикатор текущей раскладки клавиатуры (позволяет переключиться с одного языка на другой).

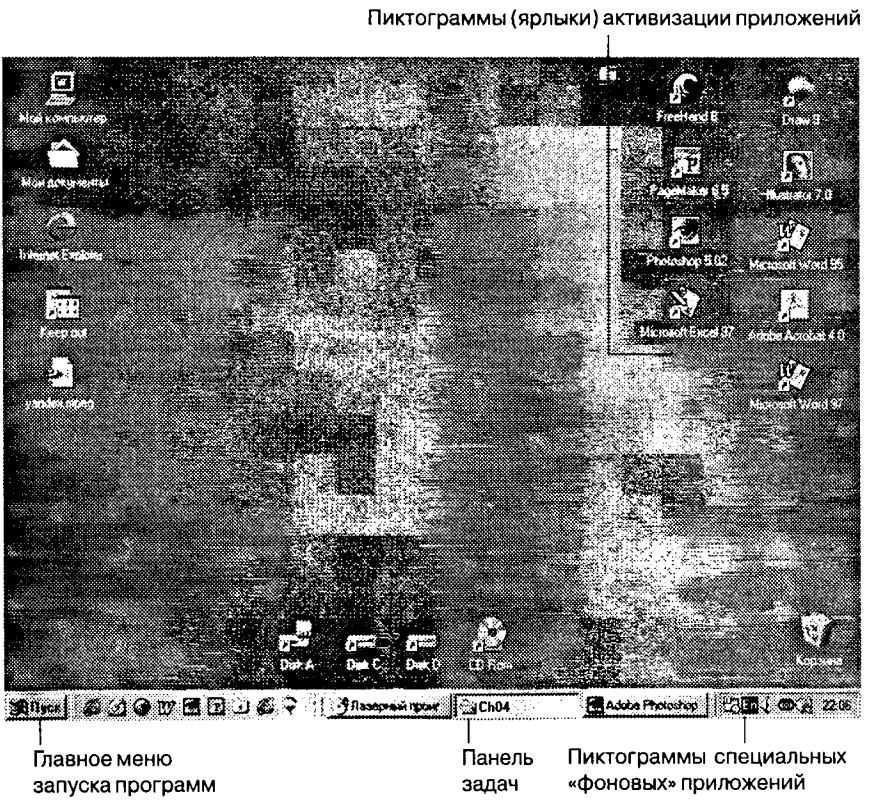


Рис. 4.8. Общий вид рабочего стола Windows 98

Главное системное меню (рис. 4.9) позволяет выполнять различные операции:

- запускать программы;
- открывать документы;
- вызывать панели управления (для настройки компьютера);
- получать справку и т. д.

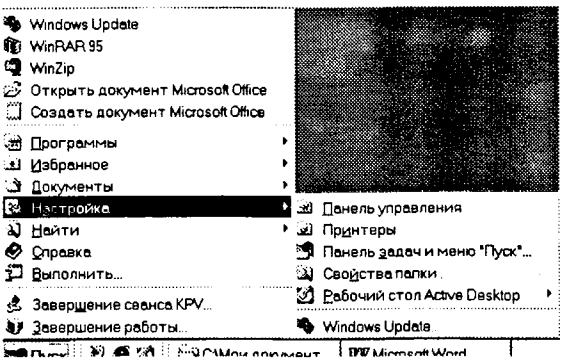


Рис. 4.9. Главное системное меню

Панель управления (в окно Панели управления можно попасть из окна Мой компьютер или из Главного системного меню) используется для изменения режима работы операционной системы и пользовательского интерфейса, для установки программного и аппаратного обеспечения, для настройки параметров клавиатуры, мыши, экрана и т. д., см. рис. 4.10.

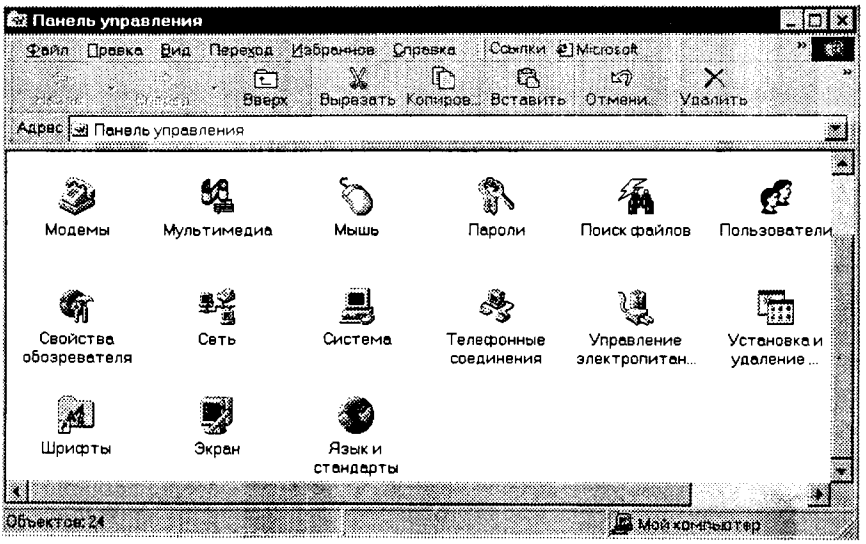


Рис. 4.10. Панель управления

Значок Мой компьютер соответствует на рабочем столе папке, отражающей содержание всего компьютера целиком (рис. 4.11). Чтобы просмотреть находящиеся на компьютере папки, файлы, диски, получить доступ к различным устройствам и к их настройке, нужно дважды щелкнуть по значку.

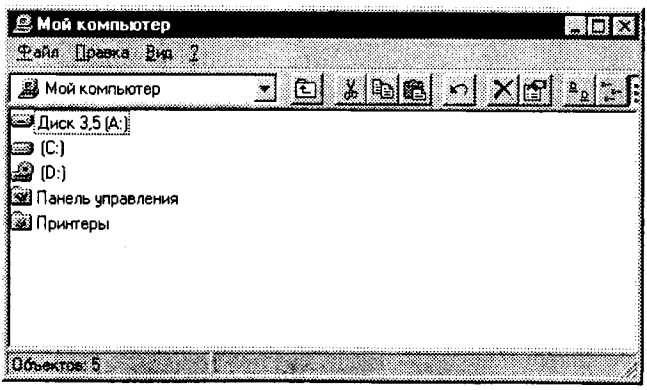


Рис. 4.11. Мой компьютер

Термин *корзина* (рис. 4.12) впервые появился в Windows 95. По сути она представляет собой специальную папку, в которую изначально помещаются удаленные

файлы из других каталогов. Данная организация процесса удаления файлов позволяет «безболезненно» восстанавливать ошибочно удаленные файлы. По мере работы следует регулярно очищать корзину (особенно когда возникают проблемы со свободным пространством на диске). Сделать это можно, вызвав из меню окна Корзина команду Файл ▸ Очистить корзину, см. также 4.5.1, *очистка дисков*.

Если пользователь однозначно уверен, что удаляемый файл больше ему не понадобится, он может удалить его, минуя корзину (например, нажав комбинацию клавиш Shift+Del, находясь в программе Проводник).

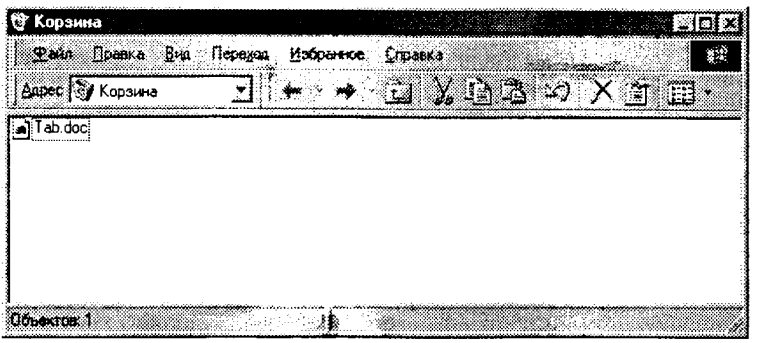


Рис. 4.12. Корзина

Проводник (Explorer) Windows является прекрасным средством для обеспечения высокопроизводительной работы с файловой системой. Он может одновременно отображать как структуру вложенности находящихся на компьютере папок (их иерархию), так и содержимое выделенной папки. Это особенно удобно при копировании и перемещении данных. Достаточно открыть папку, содержащую нужный файл, и перетащить этот файл в другую папку.

Чтобы запустить Проводник Windows, нужно открыть Главное меню и выбрать в нем пункт Программы, см. рис. 4.13.

Проводник (рис. 4.14) обобщает в себе возможности, заложенные в режимах Мой компьютер и Сетевое окружение, и т. п. Фактически, они являются его отдельными реализациями для частных случаев. С помощью меню Вид (или соответствующей пиктограммы на панели инструментов) можно задавать режимы представления содержимого папок (крупные и мелкие значки, список, таблица).

На рабочий стол также могут быть помещены *ярлыки* (shortcuts) какого-либо объекта (файл, программы, сетевая папки, диск и т. п.). Ярлыки представляют собой высокоэффективное средство ускоренного доступа к объектам: двойной щелчок по ярлыку открывает объект, с которым он связан.

Справочная система Windows представлена в виде справочника, имеющего три вкладки: Содержание (Contents), Предметный указатель (Index), Поиск (Search), см. рис. 4.15. Для того чтобы получить справку, нужно выбрать пункт Справка из Главного меню. Заметим также, что пользователь в Windows, нажав клавишу F1, в любой момент может получить справку по работе с любой программой.

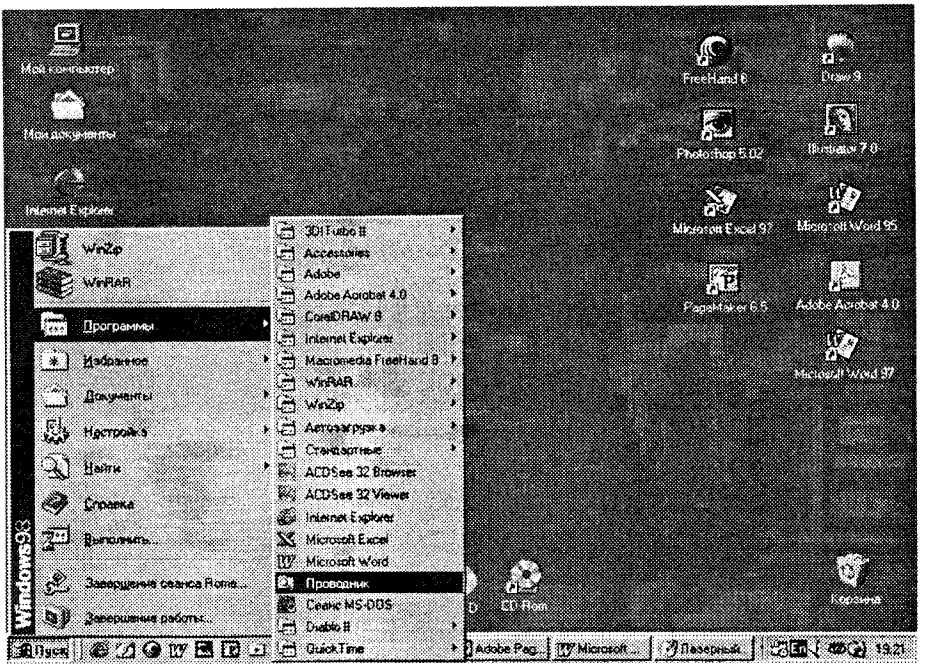


Рис. 4.13. Запуск программы Проводник из Главного системного меню (кнопка Пуск)

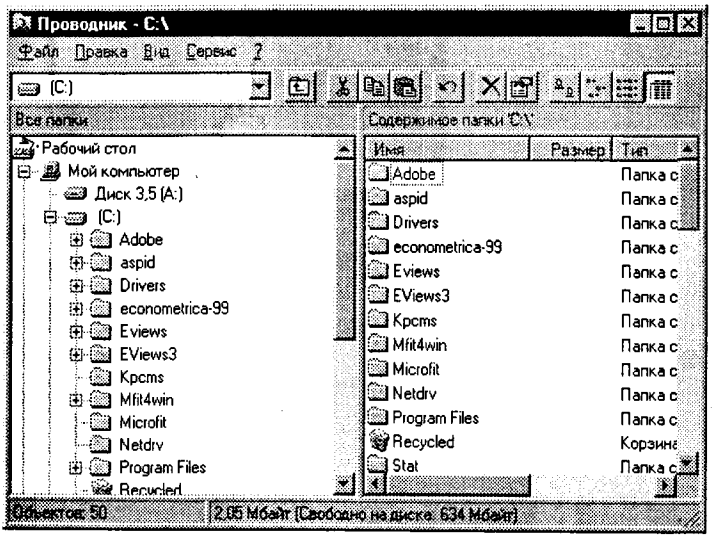


Рис. 4.14. Окно Проводника

В окнах диалога Windows предусмотрен новый способ получения справки. Достаточно нажать расположенную в правом верхнем углу окна кнопку с вопросительным знаком, после чего курсор мыши, форма которого в данном режиме меняется, наводится на интересующий элемент. В результате на экране появляется

всплывающее окно с поясняющем текстом (рис. 4.15). То же самое можно получить, щелкнув правой кнопкой мыши и выбрав в контекстном меню пункт Что это такое?

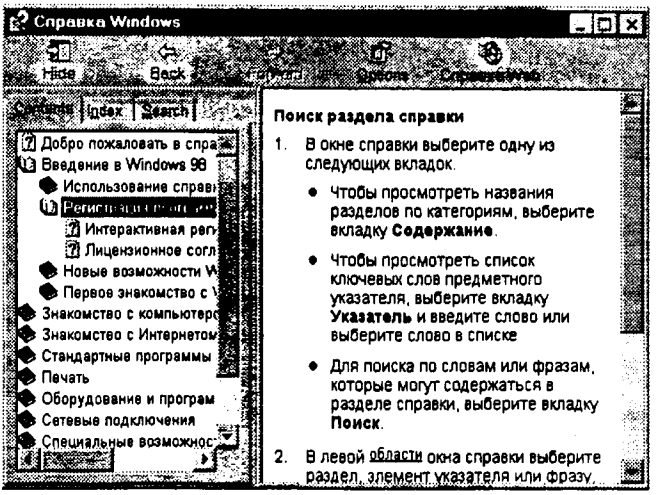


Рис. 4.15. Справочная система Windows 98

Существует несколько способов *запуска программ* из среды Windows. Перечислим основные из них:

- С помощью кнопки Пуск главного системного меню: Пуск ▸ Программы и далее из каскадного меню выбрать нужную программу.

Очевидно, что для этого запускаемая программа должна быть установлена соответствующим образом. Некоторый набор системных программных средств оказывается в группе Программы главного меню в процессе установки Windows на компьютер, другие[1] попадают туда в процессе собственной установки. Наконец, пользователь может самостоятельно добавить нужное ему приложение в группу программ с помощью функции меню Пуск ▸ Настройка ▸ Панель задач и меню Пуск....

- Другой, не менее важный способ организации запуска программ, — запуск с помощью ярлыка, расположенного на рабочем столе.

Для того чтобы создать ярлык запуска программы (и для активации иного объекта), необходимо щелкнуть правой кнопкой мыши по свободному пространству рабочего стола и в появившемся контекстном меню последовательно выбрать Создать ▸ ярлык. Дальше процесс проходит в режиме диалога, в ходе которого пользователь должен указать имя файла задачи (или документа), а также другие вспомогательные параметры: имя ярлыка, пиктограмму и т. п. В дальнейшем установленные параметры запуска задачи можно изменить. Для этого надо выбрать соответствующий ярлык, щелкнуть на нем правой кнопкой мыши и в появившемся контекстном меню выбрать пункт Свойства. После этого появляется окно, вкладки которого содержат поля с различными параметрами ярлыка запуска.

[1] В настоящее время это справедливо для большинства Windows-приложений.

ВНИМАНИЕ

Ни в коем случае не следует отождествлять ярлык запуска программного приложения (shortcut) с самим программным приложением. Ярлык является самостоятельным объектом и хранится в отдельном файле с расширением *.LNK.

- Наконец, третий способ запуска программ — запуск с помощью команды Пуск ▸ Выполнить (рис. 4.16). После ее вызова в появившемся диалоговом окне Запуск программы в поле Открыть следует указать имя файла нужной программы. Очевидно, что для этого его нужно знать. Определенную помощь в поиске данного файла может оказать режим поиска, доступный по нажатию кнопки Обзор. Необходимо заметить, что основное отличие данного режима от предыдущих состоит в том, что его имеет смысл применять при запуске программ, которые редко требуются пользователю и он не хочет ярлыками, ссылающимися на них, загромождать свой рабочий стол (группу запуска).

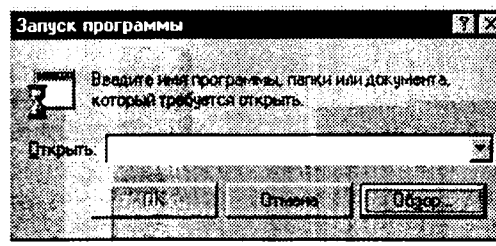


Рис. 4.16. Запуск программы из меню Пуск ▸ Выполнить

В завершение разговора о способах запуска программ необходимо обратить внимание на возможности Windows по организации пакетного старта группы задач при загрузке операционной системы. Для того чтобы программа оказалась в таком списке, пользователь должен внести ее в группу Автозагрузка (Пуск ▸ Программы ▸ Автозагрузка). Последовательность его действий в данном случае полностью аналогична той, которая осуществляется при внесении задачи в группу Программы.

Создание сессии DOS

Сразу же с появлением первых версий Windows возникла проблема запуска из их среды программ, написанных для операционной системы MS DOS. Ее актуальность объяснялась, во-первых, невозможностью сразу отказаться от всего того, что было наработано на предыдущих этапах, а во-вторых, тем, что определенные классы задач требуют непосредственного доступа к аппаратным устройствам компьютера на низком уровне, что удобнее делать в рамках DOS-приложений[1].

Необходимо отметить, что при переходах от одной версии Windows к другой механизмы работы с DOS-программами последовательно совершенствовались, предо-

[1] В частности, это касается многих интерактивных компьютерных видеоигр, в которых требуемый уровень быстродействия может быть достигнут только за счет непосредственного управления периферийными устройствами компьютера в обход стандартных функций Windows.

ставляя пользователю (в зависимости от его квалификации) все более гибкие средства настройки среды их работы.

В стандартной установке Windows предусмотрена возможность запуска DOS-сессии из меню Пуск ▸ Программы ▸ Сеанс MS DOS — в окне появится программа DOS. При запуске DOS-сессии в режиме окна в его верхней части располагается панель инструментов, позволяющая осуществлять с данным окном ряд базовых функций Windows — копирование, вставку, изменение шрифта и размеров окна и т. п. (рис. 4.17).

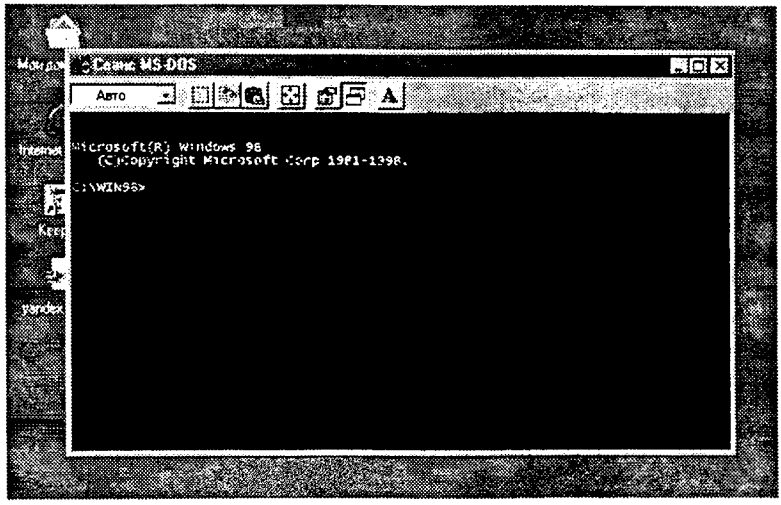


Рис. 4.17. Запуск сеанса MS DOS из-под Windows

При необходимости пользователь может переходить от оконного режима работы с DOS-сессией к полноэкранному и возвращаться назад, попеременно нажимая комбинацию клавиш Alt+Enter. В полноэкранном режиме программа приобретает такой внешний вид, какой бы она имела при полном отсутствии Windows-среды.

Для того чтобы настроить параметры DOS-сессии, следует выбрать пункт меню Пуск ▸ Программы ▸ Сеанс MS DOS, щелкнуть правой кнопкой мыши и в появившемся контекстном меню выбрать пункт Свойства. Появляющееся в результате этих действий диалоговое окно настройки параметров DOS-сессии показано на рис. 4.18. По количеству его вкладок можно судить о том разнообразии конфигурационных настроек, которые могут быть с той или иной целью изменены пользователем.

Аналогичным образом могут быть изменены параметры DOS-приложения, запуск которого производится с помощью ярлыка, располагающегося на рабочем столе.

4.5. Сервисные программные средства

В настоящем параграфе мы рассмотрим еще один тип системных программных средств — *сервисные программные средства*. Их характеризует то, что, с одной стороны, они не входят в состав операционных систем, а с другой — предназначе-

ны для реализации служебных функций по управлению компьютером. Иными словами, сервисные программы — это вспомогательные инструменты, расширяющие и дополняющие функциональность операционных систем.

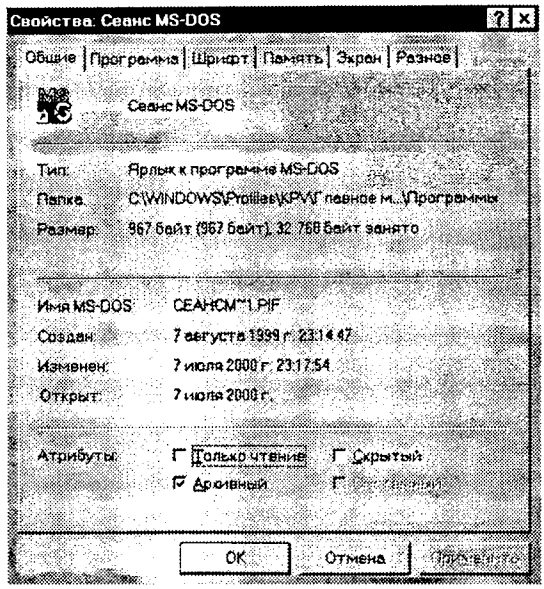


Рис. 4.18. Окно изменения свойств сеанса MS DOS

4.5.1. Служебные программы

Форматирование дисков

Накопители на магнитной основе (гибкие, жесткие диски и т. п.) перед тем, как их можно будет использовать в качестве носителей информации, должны пройти специальную операцию — форматирование. Для *форматирования дисков* обычно используется стандартная программа Windows, которая доступна при вызове контекстного меню соответствующего устройства (рис. 4.19).

Можно форматировать диски доступных floppy-устройств, жесткие диски и диски других устройств, допускающих эту процедуру. Исключение составляет системное устройство, с которого была произведена загрузка системы. Обычно это устройство с именем C:. Для форматирования носителя этого устройства необходимо загрузить систему, например, с floppy-диска A:.

Форматировать можно как новые диски, так и уже бывшие в употреблении, принимая в расчет то, что при форматировании информация, записанная на диске, теряется.

В окне Форматирование (рис. 4.20) необходимо указать параметры форматирования. Обычно, если диск новый, его необходимо полностью форматировать. Если диск уже был форматирован (часто диски форматируются уже при изготовлении), то можно воспользоваться быстрым форматированием. В этом случае область физического размещения файлов на дискете не изменяется, очищается лишь заголовоч-

ная часть диска. Тогда при случайном форматировании возможно восстановление файлов отформатированного диска при условии, что на диск не копировалось ни одного файла. Обычно такой способ форматирования используется, если необходимо удалить все файлы с дискеты, а их очень много и процесс их удаления обычным способом занимает значительное время.

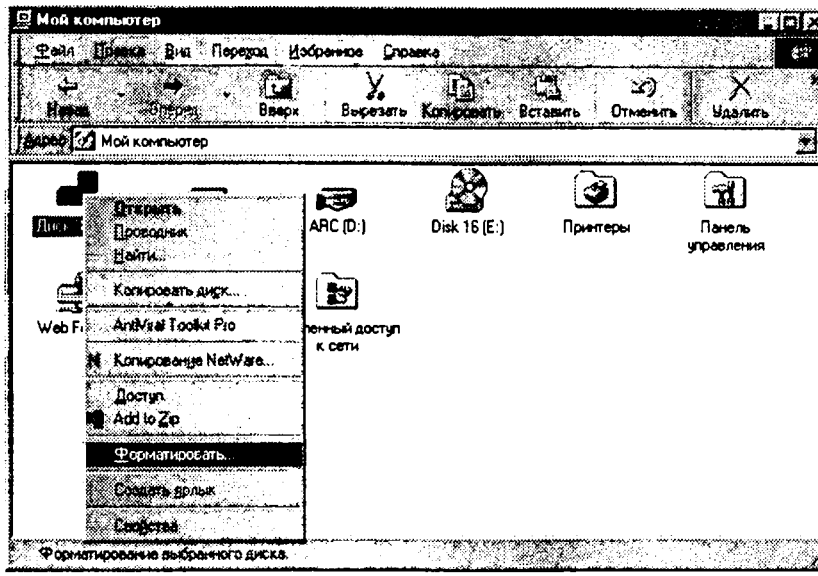


Рис. 4.19. Вызов команды форматирования диска floppy-устройства

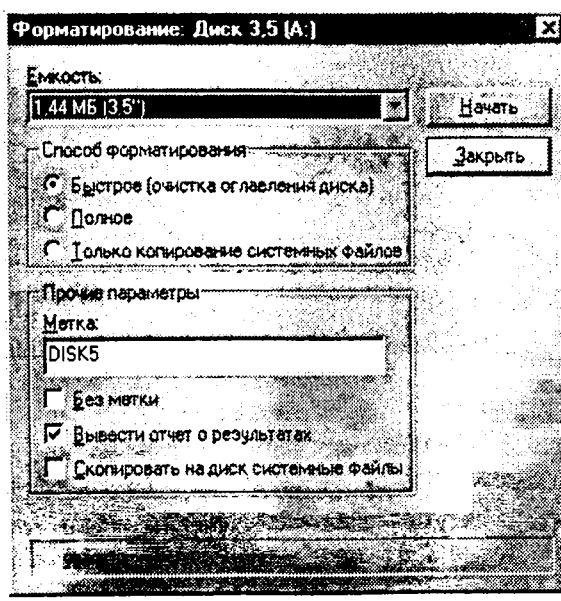


Рис. 4.20. Окно команды форматирования

Полное форматирование целесообразно применять для дисков, на которых появились «сбойные» участки. Эта процедура в некоторых случаях позволяет восстановить работоспособность диска.

При форматировании диска можно сделать его системным. В последующем с него можно загрузить компьютер. При загрузке системы с системного диска, созданного средствами Windows, графическая оболочка не загружается и рассмотренная выше команда недоступна. Если на компьютере не установлена операционная система Windows (например, на новом компьютере), то для форматирования диска можно воспользоваться DOS-утилитой Format. Справку по формату (рис. 4.21) ее командной строки можно получить, введя на приглашение DOS-команду

```
Format /h
```

Заметим также, что утилита Format может быть запущена из DOS-сессии.

```
FORMAT диск: [/V[:метка]] [/Q] [/F:размер] [/B | /S] [/C]
FORMAT диск: [/V[:метка]] [/Q] [/T:дорожки /N:секторы] [/B | /S] [/C]
FORMAT диск: [/V[:метка]] [/Q] [/1] [/4] [/B | /S] [/C]
FORMAT диск: [/Q] [/1] [/4] [/8] [/B | /S] [/C]

  /V[:метка]  Метка создаваемого тома.
  /Q          Выполнение быстрого форматирования.
  /F:размер   Размер форматируемого диска (например
              160, 180, 320, 360, 720, 1.2, 1.44, 2.88).
  /B          Отвод на диске места под системные файлы.
  /S          Копирование на отформатированный диск системных файлов.
  /T:дорожки  Число дорожек на каждой стороне диска.
  /N:секторы  Число секторов, приходящееся на одну дорожку.
  /1          Форматирование только первой стороны диска.
  /4          Форматирование диска 5,25", 360 КБ
              в дисководе для дисков высокой плотности.
  /8          Форматирование с созданием 8 секторов на дорожке.
  /C          Проверка кластеров, помеченных как поврежденные.
```

Рис. 4.21. Синтаксис DOS-команды Format

Дефрагментация дисков

Процедура *дефрагментации диска* связана с особенностями используемой файловой системы FAT, которые были подробно рассмотрены в 4.2.6. Системы FAT16 или FAT32 размещают файлы в кластерах, необязательно смежных. Доступ к файлу, расположенному в одном месте диска (файл размещается последовательно в смежных кластерах), занимает меньше времени, чем доступ к файлу, фрагменты которого разбросаны по всему диску. Чем больше фрагментов в файлах, тем медленнее доступ к ним. Для увеличения быстродействия системы диск необходимо периодически дефрагментировать. Эта стандартная служебная программа запускается в результате выполнения последовательности команд Пуск ▸ Программы ▸ Стандартные ▸ Служебные ▸ Дефрагментация диска.

Отметим, что дефрагментация — это длительная процедура по переносу информации из одних кластеров в другие, в результате которой файлы будут размещаться преимущественно в одном месте диска. Кроме этого файлы передвигаются ближе к началу диска, что также влияет на быстродействие всей системы.

Также при выполнении дефрагментации предусмотрены дополнительные возможности исправления ошибок записи на диск, о чем более подробно будет сказано в следующем пункте.

Проверка диска на наличие ошибок

В процессе эксплуатации диска возможно появление ошибок, связанных со сбоями в процессе записи на него. Некоторые из них можно исправить, не прибегая к процедуре форматирования. Для этого используется стандартная служебная программа *проверки диска* (рис. 4.22).

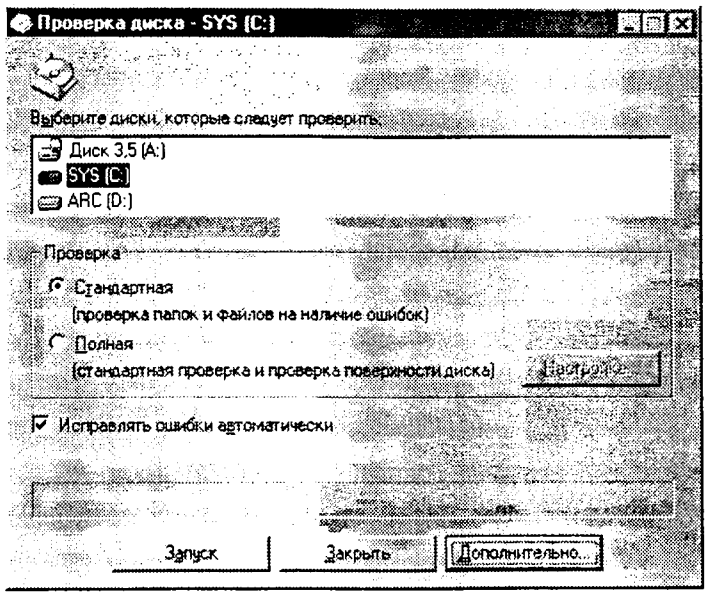


Рис. 4.22. Проверка диска

Ее запуск осуществляется из главного системного меню с помощью последовательности команд Пуск ▸ Программы ▸ Стандартные ▸ Служебные ▸ Проверка диска. Она позволяет проверить целостность файловой системы и поверхность диска (рис. 4.22).

Целостность файловой системы определяется:

- правильностью имен файлов (файлы с неправильными именами, например, нельзя открыть);
- правильностью даты и времени создания файла (неправильные дата и время могут влиять на работу многих программ, обрабатывающих такие файлы);
- уникальностью имен файлов;
- отсутствием файлов с общими кластерами;
- отсутствием кластеров, не принадлежащих ни одному файлу.

Окно настройки параметров процесса проверки диска, вызываемое по нажатии кнопки Дополнительно..., показано на рис. 4.23.

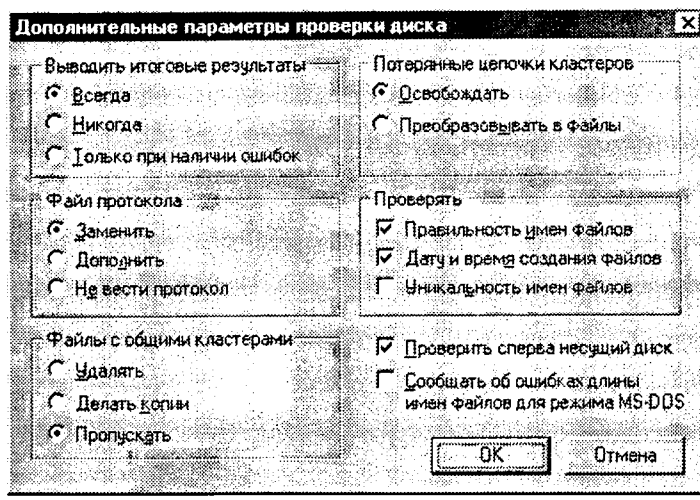


Рис. 4.23. Дополнительные параметры проверки диска

Проверка поверхности диска на наличие повреждений и аппаратных ошибок занимает больше времени, чем проверка целостности файловой системы. Для сокращения времени работы программы можно выбрать проверку не всего диска, а только системной области или области данных (рис. 4.24). При нахождении поврежденных секторов их данные обычно переносятся в другие сектора. Некоторые системные файлы нельзя переносить, поэтому рекомендуется исправление ошибок в секторах, в которых располагаются такие файлы, отменить. Для большей надежности проверки поверхности диска необходимо использовать режим записи на диск (при этом данные на диске не теряются). Для сокращения времени тестирования от этого режима можно отказаться.

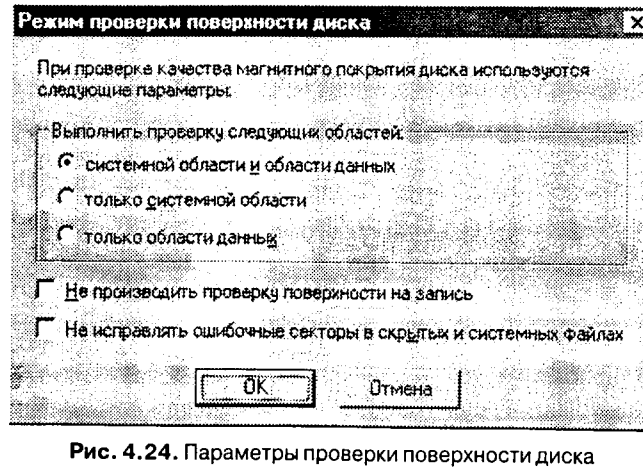


Рис. 4.24. Параметры проверки поверхности диска

Обнаруженные ошибки могут исправляться автоматически в соответствии с заданными параметрами. Для исправления других ошибок файловой системы

необходимо использовать другие программы, например известные утилиты Norton Utilities.

Очистка дисков

В процессе эксплуатации дисковых носителей свободное пространство заполняется файлами. Для создания новых файлов необходимо освободить место, занимаемое уже ненужными файлами. Эти файлы удаляются. Удаленные файлы помещаются в корзину на рабочем столе, то есть файлы меняют «прописку», но не удаляются с диска. Для освобождения места, занятого удаленными файлами, необходимо очистить корзину на рабочем столе.

«Ненужность» тех или иных файлов в конечном итоге определяется пользователем. В помощь пользователю предоставляется специальная программа *очистки диска*, которая запускается последовательностью Пуск ▸ Программы ▸ Стандартные ▸ Служебные ▸ Очистка диска.

Программа очистки диска может найти и освободить место, занимаемое файлами следующих категорий (рис. 4.25):

- Temporary Internet Files – файлы, которые копируются из Интернета для быстрого просмотра;

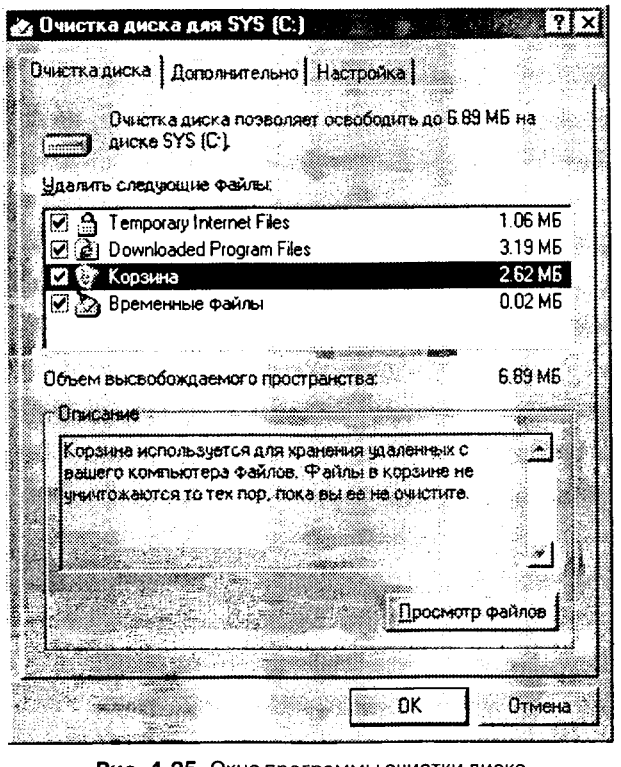


Рис. 4.25. Окно программы очистки диска

- Downloaded Program Files — файлы, которые временно сохраняют элементы ActiveX и приложения Java, автоматически загружаемые из Интернета при просмотре некоторых страниц;
- корзина — удаленные файлы (см. 4.4.6, *корзина*);
- временные файлы — файлы, создаваемые программами при своей работе. Обычно эти файлы удаляются после окончания работы программ, но по тем или иным причинам эти файлы могут остаться на диске.

  При недостатке свободного места на диске можно удалить неиспользуемые компоненты Windows либо неиспользуемые программы. Данная возможность предоставляется программой очистки диска. Следует помнить, что при этом освобождается место на том диске, на котором установлено выбранное программное обеспечение.

Дополнительное свободное место можно получить, установив FAT32 вместо FAT16. Эта процедура запускается последовательностью Пуск ▸ Программы ▸ Стандартные ▸ Служебные ▸ Преобразование диска в FAT32.

4.5.2. Архивация данных

Наиболее простым и эффективным способом восстановления потерянных или испорченных данных на компьютере является создание копий. Для уменьшения места, необходимого для содержания копий, и удобства эксплуатации создано достаточно много программ, позволяющих работать с архивными файлами.

Архивный файл может содержать один и более файлов в сжатом виде. Можно как извлекать из этого файла исходные файлы в их первоначальном виде, так и добавлять в него новые.

Среди популярных программ *архивации*, работающих в среде Windows, можно назвать такие, как WinZip, WinRar, WinArj, NetZip и др.

Архиватор WinZip

После установки этой программы создается не только меню WinZip, но и ярлыки на рабочем столе и в кнопке Пуск, в контекстное меню для папок и файлов добавляются пункты, связанные с архивацией.

Программа может быть запущена последовательностью Пуск ▸ Программы ▸ WinZip. На рис. 4.26 показан стандартный вид окна WinZip версии 8.0.

Создание и изменение архива

Работа с архивом начинается с создания файла архива при помощи последовательности команд File ▸ New Archive (как в любых программах Windows, любая команда или операция может быть осуществлена разными способами, в частности, клавишами Ctrl+N или кнопкой New на панели инструментов). Файл архива будет создан после задания его имени. При установке флажка Add dialog будет открыто дополнительное диалоговое окно Add, в котором можно указать имена и парамет-

ры файлов, добавляемых в архив. В противном случае (если флажок Add dialog не установлен) будет создан пустой архивный файл.

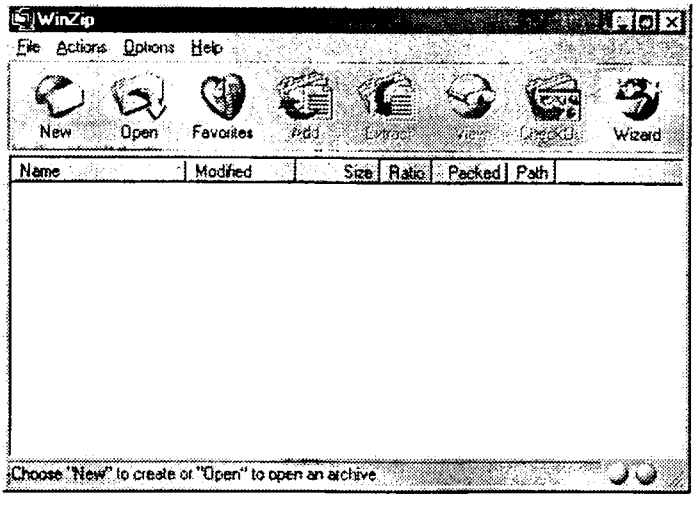


Рис. 4.26. Окно программы архиватора WinZip

При добавлении файлов в уже существующий архив необходимо открыть выбранный файл архива. Для этого используется команда открытия File ▸ Open Archive.

Для включения файла в архив используется команда добавления Actions ▸ Add.

В окне Add (рис. 4.27) кроме указания файла (файлов) для добавления в архив задаются параметры включения в архив:

- Action
  - Add (and replace) files — добавляет файлы в архив. Если файл в архиве уже есть, то он перезаписывается;
  - Freshen existing files — обновляются только те файлы, которые уже есть в архиве. Новые файлы не добавляются. Обновление осуществляется при условии, что обновляемые файлы имеют более старую дату создания;
  - Move files — добавляемые файлы удаляются с диска, то есть файлы перемещаются с диска в архив;
  - Update (and add) files — обновляются файлы, которые уже есть в архиве, и добавляются новые.
- Compresion — параметр, значение которого определяет степень сжатия файла. Время архивации и степень сжатия имеют обратную зависимость.
- Include subfolders — добавляются все файлы, находящиеся в папках открытой папки.
- Save full path info — запоминается полный путь файла. При извлечении файлов из архива файлы будут помещаться в соответствующие папки. Если папок нет, они будут созданы. В архив можно поместить целое дерево папок с файлами.

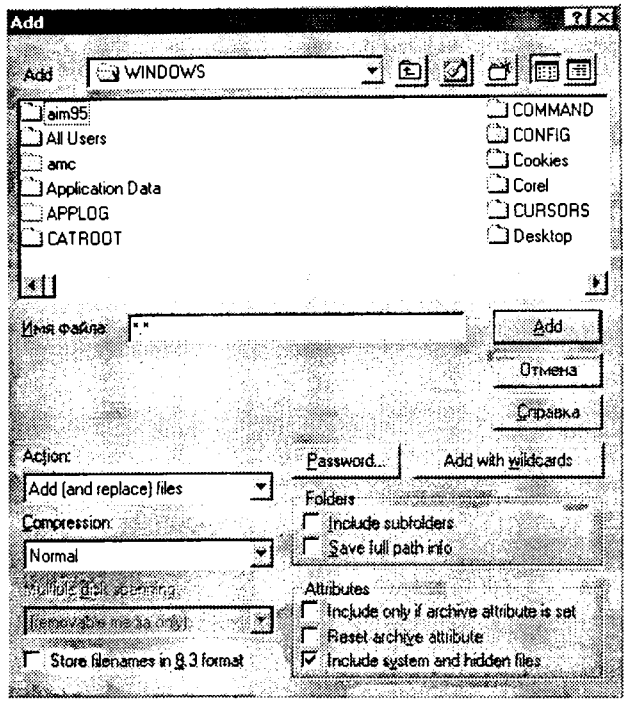


Рис. 4.27. Окно WinZip для добавления файла(ов) в архив

- Include only if archive attribute is set — в архив добавляются только те файлы, которые еще не помещались в архив (соответствующий атрибут файла указывает на необходимость архивации файла).
- Reset archive attribute — добавленный в архив файл помечается как уже подвергнутый архивации.
- Include system and hidden files — в архив можно помещать файлы с указанными атрибутами.

Предпоследние два параметра используются для архивации только тех файлов, которые необходимо архивировать. Файлы, которые хотя бы раз архивировались, в архив не добавляются.

При добавлении файлов можно указать пароль для недопущения несанкционированного доступа к файлам архива.

Команда добавления запускается нажатием кнопки Add.

Извлечение файлов из архива

При извлечении файла из архива извлекаемый файл из архива не удаляется. Для извлечения файлов из архива необходимо открыть файл архива и запустить команду извлечения.

Команда извлечения файла из архива может быть вызвана последовательностью команд Actions ▸ Extract.

В появившемся окне (рис. 4.28) необходимо в поле Extract to: ввести имя папки, в которую будут помещаться извлекаемые из архива файлы, а в поле Files: — имена извлекаемых файлов[1].

Выбор существующей папки можно осуществить в поле Folders/drivers, для создания новой папки следует воспользоваться кнопкой New folder....

Чтобы извлечь все файлы из архива, необходимо установить параметр All files. Параметр `Selected files` используется для извлечения файлов, выбранных в поле `Folders/drivers` перед вызовом команды извлечения.

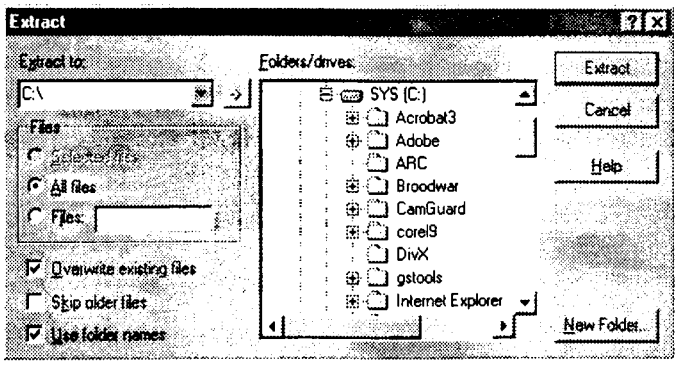


Рис. 4.28. Окно WinZip для извлечения файла(ов) из архива

Дополнительные параметры позволяют управлять процессом извлечения файлов из архива:

- Overwrite existing files — при существовании на диске файла с именем, совпадающим с именем извлекаемого файла, первый из них удаляется.
- Skip older files — при существовании на диске более нового файла с именем, совпадающим с именем извлекаемого файла, последний не извлекается из архива.
- Use folder name — извлекаемый из архива файл помещается в папку, имя которой было записано в архив вместе с файлом.

Команда извлечения запускается нажатием кнопки Extract.

Дополнительная информация о файлах в архиве

В рабочем окне программы WinZip располагается информация о хранимых файлах:

- имя файла;
- дата изменения;
- исходный размер файла;
- размер файла в архиве (в процентах к исходному размеру);
- экономия на размере файла (в байтах);
- полный путь файла.

[1] Если вводится список из нескольких имен, то они разделяются запятыми.

Некоторые дополнительные возможности

Программа WinZip позволяет работать с файлами без извлечения их из архива. При этом файл автоматически извлекается из архива в специальную рабочую папку программы WinZip и обрабатывается стандартным образом. Например, документ редактора Word можно открыть прямо в архиве, щелкнув мышью на его имени. Более того, изменения, которые вносятся в документ, фиксируются и в файле архива.

ВНИМАНИЕ

Если документ состоит из нескольких файлов, то для нормальной работы с ним необходимо предварительно извлечь все необходимые документы, так как WinZip не способен отследить внутреннюю логику связей между этими файлами.

4.5.3. Антивирусные программные средства

Что такое вирус?

Вирус (*вирусная программа*) — это, как правило, небольшая по объему последовательность программных кодов, обладающая следующими свойствами:

1. Возможность создавать свои копии и внедрять их в другие программные объекты.
2. Обеспечение скрытости (латентность) до определенного момента ее существования и распространения.
3. Несанкционированность (со стороны пользователя) производимых ею действий.
4. Наличие отрицательных последствий от ее функционирования.

Приведенные признаки, строго говоря, не являются характеристическими, так как не все программы, обычно называемые вирусами, обладают всеми из перечисленных свойств. С другой стороны, другие программы, которые обладают одним или несколькими свойствами из этого списка, также могут не являться вирусами.

Любая обстоятельная дискуссия о компьютерных вирусах так или иначе затрагивает проблему причин их возникновения. Сразу оговоримся, что их обсуждение выходит за рамки настоящей книги. Поэтому ограничимся констатацией того факта, что реально существует целый комплекс причин существования и развития «индустрии» вирусов. Среди основных из них следует выделить:

- причины технического характера (пробелы в защите ранних версий операционных систем, предназначенных для персональных компьютеров с ограниченными возможностями);
- причины экономического характера (допустим, борьба с конкурентами);
- причины социального и психологического характера (наличие «специалистов», обладающих профессиональной квалификацией для написания вирусов и по той или иной причине не находящих путей для конструктивной реализации своих способностей).

Классификация вирусов

В настоящее время описано большое количество вирусов, исчисляемых десятками тысяч. Но среди них можно насчитать не более 40–50 активных, которые имеют распространение. Вирусы можно классифицировать следующим образом:

- загрузочные вирусы;
- файловые вирусы;
- макровирусы;
- сетевые вирусы.

Загрузочные вирусы

Загрузочные вирусы внедряются в так называемые загрузочные области носителей[1]. Если зараженным окажется системный диск, с которого происходит загрузка системы, то управление получает не обычная программа загрузки, а замещающий его код вируса. При заражении вирус считывает необходимую информацию из первоначального загрузчика и сохраняет ее в своем коде.

Как правило, загрузочные вирусы являются резидентными, то есть при загрузке системы вирус устанавливается в память компьютера и находится там постоянно, перехватывает необходимые прерывания и производит все действия, на которые он запрограммирован.

Файловые вирусы

Файловые вирусы используют особенности файловой организации системы. Такие вирусы внедряются в выполняемые файлы, в библиотечные и объектные модули, загружаемые драйверы, файлы исходных текстов программ.

Файловые вирусы заражают файлы, внедряясь в начало, в конец или в середину файла. Такие файлы остаются работоспособными, так как код вируса дописывается к исходному коду.

Некоторые вирусы вместо исходного содержимого файла записывают свой код. Такие файлы уже не могут быть использованными по своему прямому назначению.

Файловые вирусы могут и не изменять содержимое заражаемых файлов. Иногда для своей активизации они используют свойства операционной системы, определяющие порядок запуска программ. Так, если в одном каталоге есть файлы с одинаковыми именами, то сначала запускается файл с расширением имени *.BAT, потом *.COM и *.EXE. Соответственно, если заражаемый файл имеет расширение *.EXE, то вирусный файл с тем же именем, но расширением *.COM будет запускаться первым. В других случаях файл с вирусом для перехвата управления располагает себя в таком каталоге, который, согласно переменной среды PATH, находится «выше» каталога одноименного программного файла. Некоторые вирусы «присваивают» имя заражаемого файла, а тому дают новое (измененное) имя. Это, опять-таки, приводит к тому, что при запуске «зараженного» файла сначала отработает вирус.

---

[1] Загрузочная область – начальные секторы (Boot – секторы) накопительных устройств, предназначенные для хранения специальной программы – загрузчика операционной системы.

Как правило, затем управление передается на оригинальный файл. Последнее делается для того, чтобы скрыть от пользователя факт работы вируса.

Среди файловых вирусов встречаются вирусы, которые не связаны ни с одним из файлов. Запуск таких вирусов осуществляется «по ошибке». Файлы таких вирусов имеют «интересные» с точки зрения пользователя имена: Install.exe, Start.com и т. д. Такие вирусы называют *червями* (англ. вариант — worm).

Макровирусы

Макровирусы — это файловые вирусы, использующие особенности файлов документов популярных редакторов и электронных таблиц. В этих файлах размещаются программы на макроязыках, возможности которых позволяют создавать программы-вирусы[1].

Сетевые вирусы

Сетевые вирусы используют для своего распространения возможности компьютерных сетей. Наибольшее распространение сетевые вирусы получили в конце 80-х годов. Эти вирусы использовали ошибки в сетевых протоколах и программном обеспечении глобальных сетей. Сейчас эти ошибки исправлены и эти вирусы не имеют распространения.

В настоящее время сетевые вирусы распространяются, используя возможности электронной почты. В основном это макровирусы.

Особенности алгоритма работы вирусов

Вирусы, которые могут оставлять свои копии в оперативной памяти, называются *резидентными*. Такие вирусы остаются активными и после завершения работы программы (и даже после удаления файла вируса) вплоть до перезагрузки системы.

С целью скрыть до определенного момента свое существование некоторые вирусы используют разные методы. *Стелс-вирусы* перехватывают команду чтения зараженного участка и подставляют незараженный код. *Полиморфик-вирусы* шифруют свой код, чтобы затруднить свое обнаружение.

Аппаратные устройства — источники вирусов

Заражение компьютера происходит через устройства, позволяющие вводить информацию в компьютер:

- модем.

Интернет — основной источник распространения вирусов. Заражение происходит через копирование зараженных файлов с серверов глобальной сети, при обмене через электронную почту письмами, в которых содержатся документы формата Word, которые могут быть заражены макровирусами.

---

[1] Более подробно проблема макровирусов будет рассмотрена в главе 5, посвященной текстовым процессорам.

- сетевая карта.

При регистрации в локальной сети зараженной рабочей станции происходит заражение системных файлов сервера. Инфицирование незараженной рабочей станции происходит при регистрации на зараженном сервере.

- устройства хранения информации со сменными носителями.

При использовании дискет на зараженном компьютере происходит заражение дискет, с которых можно заразить любой компьютер. Встречаются зараженные диски CD-ROM, которые, как правило, содержат пиратское программное обеспечение.

Исключение составляют устройства «ручного» ввода: клавиатура, мышь и т. д.

Способы защиты от вирусов

Одним из основных последствий деятельности вирусов является потеря или порча информации. Поэтому для обеспечения устойчивой и надежной работы необходимо (или, по крайней мере, желательно) всегда иметь чистые, незараженные (эталонные) копии используемой информации и программного обеспечения.

При заражении вирусом нарушается целостность информации. Использование специальных утилит, которые позволяют отслеживать информацию о системных областях и файлах (контрольные суммы, размеры, даты создания и др.), позволяет относительно быстро обнаружить проникновение вирусов.

Использование специальных *антивирусных средств* в большинстве случаев позволяет не только выявить вирусное вторжение, но и оперативно обезвредить обнаруженные вирусы и восстановить испорченную информацию.

Среди популярных и эффективных антивирусных программных систем можно выделить:

- антивирус Касперского – AntiViral Toolkit Pro (AVP);
- McAfee VirusScan;
- Panda Antivirus;
- Trend Micro PC-Cillin;
- F-Secure Anti-Virus;
- Sophos Anti-Virus;
- Norton AntiVirus.

К сожалению, не существует антивирусов, которые бы обеспечивали абсолютную защиту от вирусов. Это связано, прежде всего, с постоянным развитием отрасли написания вирусов, в ответ на которое создатели антивирусов, в свою очередь, стремятся как можно быстрее создать нужную вакцину. Поэтому необходимо пользоваться антивирусами, версии которых постоянно обновляются.

При выборе антивирусного обеспечения следует принимать во внимание перечень и качество предоставляемых услуг:

- возможность сканирования «на лету», когда любой объект, к которому осуществляется доступ, проверяется на наличие вируса;

- возможность сканирования по запросу, когда время и область действия антивируса определяются пользователем;
- возможность настройки антивируса в соответствии с потребностями пользователя.

Качество работы антивируса определяется его надежностью (отсутствие ошибок, зависаний и т. п.) и эффективностью (обезвреживание всех вирусов, особенно полиморфик-вирусов, отсутствие ложных срабатываний).

Антивирус Касперского — AntiViral Toolkit Pro

Лаборатория Касперского предлагает достаточно широкий набор антивирусных средств AntiViral Toolkit Pro (AVP) для большинства известных операционных систем. Далее кратко рассмотрены основные особенности работы вирус-сканера для операционной среды Windows (рис. 4.29).

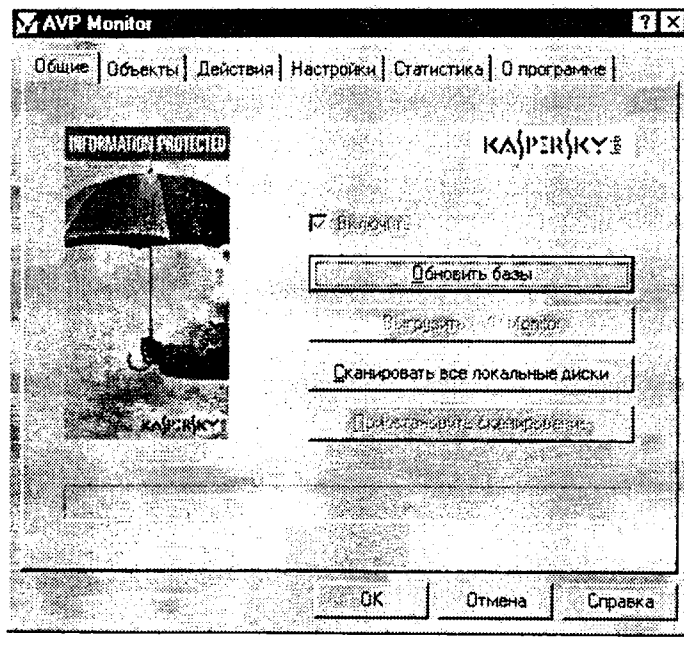


Рис. 4.29. Окно Монитора AVP, вкладка Общие

После установки антивирусной программы монитор запускается при загрузке компьютера (см. папку Автозагрузка). Работа антивируса несколько замедляет работу компьютера. Если нет непосредственной опасности заражения, то антивирус можно временно отключить, убрав соответствующий флажок, либо выгрузить монитор из памяти. Здесь некоторые функции антивируса отключены, что зависит от используемого варианта антивируса.

Монитор антивируса работает постоянно, проверяя файлы, к которым осуществляется текущий доступ. Файлы, которые должны быть проверены, определяются

вкладкой Объекты при помощи указания соответствующего типа файлов и файлов, имеющих вложения, (рис. 4.30). Во вкладке Действия определяется процедура, которая задействуется при обнаружении вируса (рис. 4.31).

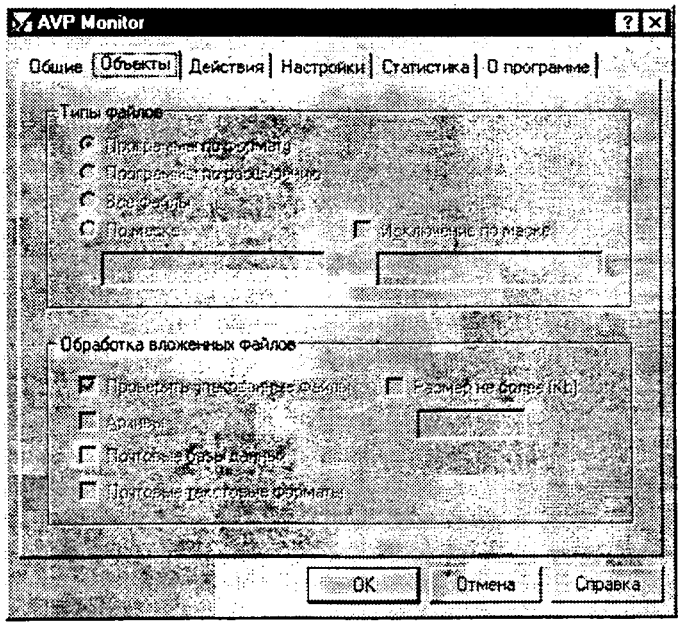


Рис. 4.30. Окно Монитора AVP, вкладка Объекты

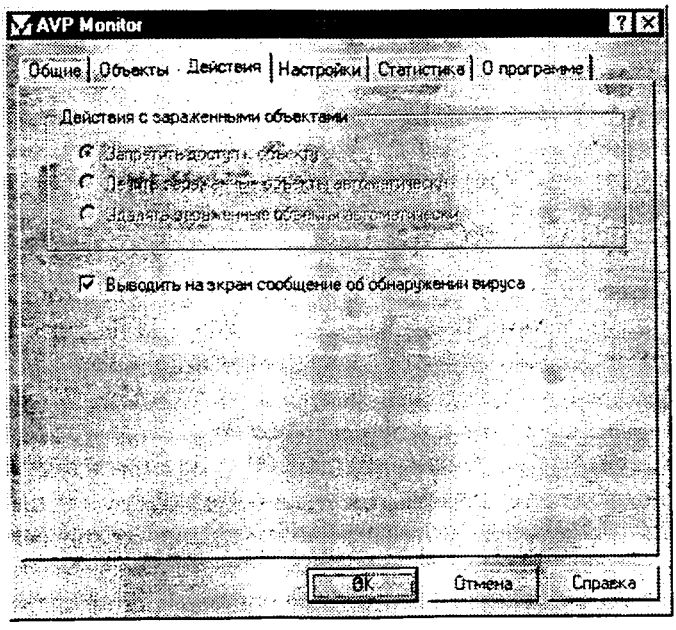


Рис. 4.31. Окно Монитора AVP, вкладка Действия

В антивирусе AVP предусмотрена возможность сканирования всех файлов, находящихся на локальных дисках. Соответствующая кнопка есть в мониторе, но рекомендуется использовать специальную программу, входящую в комплект поставки. Здесь можно выбрать устройство для сканирования, например floppy-дисковод, отдельную папку, файлы которой необходимо сканировать. Антивирус позволяет сканировать любые доступные сетевые устройства (рис. 4.32).

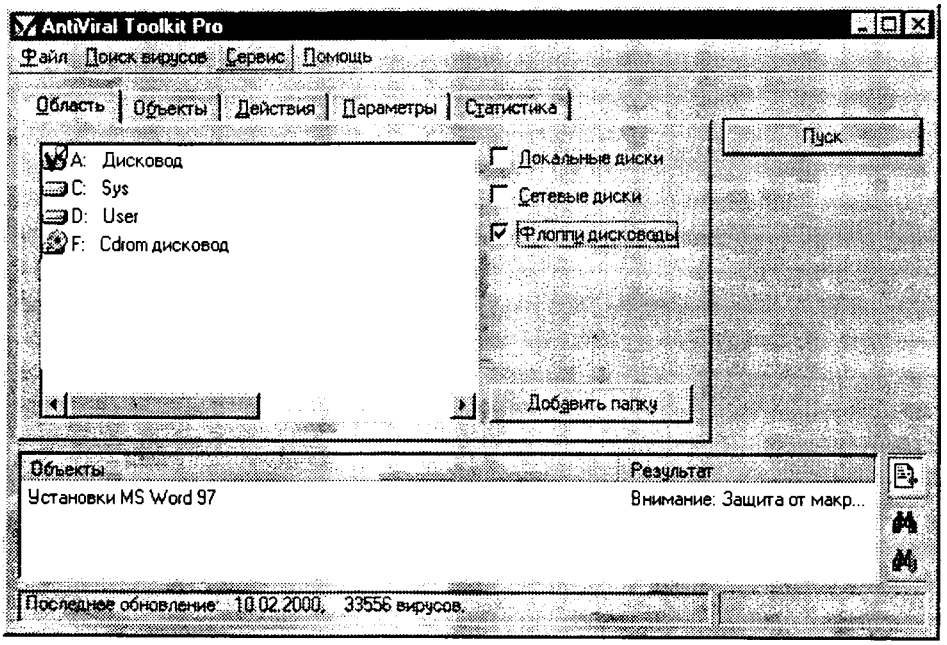


Рис. 4.32. Окно AntiViral Toolkit Pro, вкладка Область

4.6. Краткий обзор современных операционных систем

Одной из альтернатив семейству операционных систем Windows является семейство операционных систем UNIX. Основное отличие и преимущество этой системы заключается в том, что она реализована для очень широкого круга аппаратных платформ. Так, например, серверная операционная система Windows NT существует только для аппаратных платформ Intel и Alpha. В то же время, различные версии UNIX созданы для практически любых компьютеров самых разных производителей, начиная с персональных и заканчивая самыми мощными суперкомпьютерами. Благодаря своей необычайной гибкости за время, прошедшее с момента своего появления в 1969 году, операционные системы семейства UNIX получили широкое распространение на машинах различной мощности и архитектуры, обеспечивая на них общие условия выполнения программ.

UNIX является не только многозадачной операционной системой, но и многопользовательской системой, которая позволяет нескольким пользователям разделять вычислительные ресурсы одного компьютера. Это свойство позволяет использовать UNIX в качестве операционной системы для мини-компьютеров и суперкомпьютеров (мэйнфреймов), к которым пользователи подключаются через терминалы.

В первых версиях UNIX взаимодействие с пользователем осуществлялось с помощью командной строки. Затем появились различные варианты графического интерфейса для UNIX. Графический интерфейс позволил существенно облегчить работу пользователей и сделать систему легко доступной для тех, кто начинал работу в среде Windows.

Среди программного обеспечения, кроме мощных систем управления базами данных, систем управления предприятием, банковских систем, систем автоматизированного проектирования, для UNIX написано огромное количество прикладных программ, в том числе текстовые процессоры, процессоры электронных таблиц, графические редакторы, что делает UNIX еще более универсальной системой.

Средства сетевого взаимодействия являются неотъемлемой частью UNIX, что делает эту систему удобной для создания различных гетерогенных систем, создания сетевых приложений, а также для использования в качестве основной серверной платформы в Интернете.

История семейства операционных систем UNIX началась в 1965 г., когда фирма Bell Telephone Laboratories, объединив свои усилия с компанией General Electric и проектом MAC Массачусетского технологического института, приступили к разработке новой операционной системы, получившей название Multics. Перед системой Multics были поставлены следующие задачи: обеспечить одновременный доступ к ресурсам ЭВМ большого количества пользователей, обеспечить достаточную скорость вычислений и хранение данных и дать возможность пользователям в случае необходимости совместно использовать данные. Хотя первая версия системы Multics и была запущена в 1969 г. на ЭВМ GE 645, она не обеспечивала выполнение главных вычислительных задач, для решения которых предназначалась, и не было ясно, когда цели разработки будут достигнуты. Поэтому фирма Bell Laboratories прекратила свое участие в проекте.

Сотрудники Bell Laboratories, работавшие над проектом Multics, вынуждены были перенести систему на другую аппаратную платформу. Пытаясь усовершенствовать среду программирования, Кен Томпсон, Дэннис Ричи и другие набросали на бумаге проект файловой системы, получивший позднее дальнейшее развитие в ранней версии файловой системы UNIX. Томпсоном были написаны программы, имитирующие поведение предложенной файловой системы в режиме подкачки данных по запросу, им было даже создано простейшее ядро операционной системы. Томпсон воспользовался компьютером PDP-7. Для того чтобы улучшить условия разработки, Томпсон и Ричи выполнили на PDP-7 свой проект системы, включивший первую версию файловой системы UNIX, подсистему управления процессами и небольшой набор утилит. Новая система получила название UNIX, его придумал еще один из сотрудников Исследовательского центра по информатике Bell Laboratories Брайан Керниган.

В 1971 г. система UNIX была перенесена на ЭВМ PDP-11. В тоже время Томпсон и Ричи создали язык программирования С и написали для новой системы транслятор с С. В 1973 г. система была заново написана на С, что позволило переносить систему с одной аппаратной платформы на другую. Количество машин фирмы Bell Laboratories, на которых была инсталлирована система, возросло до 25, в результате чего внутри фирмы была создана группа по системному сопровождению UNIX.

В то время корпорация AT&T не могла заниматься продажей компьютерных продуктов, в соответствии с соглашением, подписанным ею с федеральным правительством в 1956 году, и распространяла систему UNIX среди университетов, которым она была нужна в учебных целях. Следуя букве соглашения, корпорация AT&T не рекламировала, не продавала и не сопровождала систему. Несмотря на это популярность системы устойчиво росла. К 1977 году количество компьютеров, на которых функционировала система UNIX, увеличилось до 500, причем 125 из них работали в университетах. Система UNIX завоевала популярность среди телефонных компаний, поскольку обеспечивала хорошие условия для разработки программ, обслуживала работу в сети в режиме диалога и работу в реальном масштабе времени. Помимо университетов, лицензии на систему UNIX были переданы коммерческим организациям.

С ростом популярности мини- и микрокомпьютеров другие компании стали переносить систему UNIX на новые аппаратные платформы, однако ее простота и ясность побудили многих разработчиков к самостоятельному развитию системы, в результате чего было создано несколько вариантов базисной системы. За период между 1977 и 1982 годами фирма Bell Laboratories объединила несколько вариантов, разработанных в корпорации AT&T, в один, получивший коммерческое название UNIX версия III. В дальнейшем фирма Bell Laboratories добавила в версию III несколько новых особенностей, назвав новый продукт UNIX версия V, и эта версия стала официально распространяться корпорацией AT&T с января 1983 года. В то же время сотрудники Калифорнийского университета в Беркли разработали вариант системы UNIX, получивший название BSD 4.3 для машин серии VAX и отличающийся некоторыми новыми интересными особенностями. К началу 1984 года система UNIX уже была инсталлирована приблизительно на 100 000 компьютеров во всем мире.

В настоящее время существует множество версий операционной системы UNIX от различных производителей. Среди них можно выделить несколько наиболее известных коммерческих версий: SunOS и Solaris для компьютеров компании Sun, AIX для мини-компьютеров компании IBM, IRIX для компьютеров компании Silicon Graphics, SCO UNIX компании Santa Cruz Operation (SCO) для компьютеров на платформе Intel, — а также свободно распространяемых: FreeBSD и Linux для компьютеров на платформе Intel.

Таким образом, можно суммировать основные причины популярности системы UNIX:

- система написана на языке высокого уровня, благодаря чему ее легко понимать, изменять и переносить на другие аппаратные платформы;
- наличие простого пользовательского интерфейса, в котором имеется возможность предоставлять все необходимые пользователю услуги;

- наличие иерархической файловой системы, легкой в сопровождении и эффективной в работе;
- обеспечение согласования форматов в файлах, работа с последовательным потоком байтов. Наличие простого последовательного интерфейса с периферийными устройствами;
- наличие встроенных средств поддержки компьютерных сетей, что делает систему UNIX одной из самых популярных серверных платформ в Интернете;
- система является многопользовательской, многозадачной; каждый пользователь может выполнять одновременно несколько процессов;
- архитектура машины скрыта от пользователя, благодаря этому облегчен процесс написания программ, работающих на различных конфигурациях аппаратных средств.

Отличительными особенностями операционной системы UNIX являются наличие ядра и организация файловой системы. Ядро непосредственно взаимодействует с аппаратной частью компьютера, изолируя прикладные программы от особенностей архитектуры компьютера. Кроме того, ядро предоставляет прикладным программам определенный набор услуг: операции ввода/вывода, создания и управления процессами, синхронизации и межпроцессорного взаимодействия.

В основе взаимодействия ядра и прикладных программ лежит концепция процесса — единицы управления и единицы потребления ресурсов. Процесс представляет собой программу в состоянии выполнения, причем в UNIX в рамках одного процесса не могут выполняться никакие параллельные действия.

Каждый процесс работает в своем виртуальном адресном пространстве. Совокупность участков физической памяти, отображаемых на виртуальные адреса процесса, называется образом процесса.

При управлении процессами операционная система использует два основных типа информационных структур: дескриптор процесса и контекст процесса.

Дескриптор процесса содержит такую информацию о процессе, которая необходима ядру в течение всего жизненного цикла процесса, независимо от того, находится ли он в активном или пассивном состоянии, находится ли образ процесса в оперативной памяти или выгружен на диск. Дескрипторы отдельных процессов объединены в список, образующий таблицу процессов. Память для таблицы процессов отводится динамически в области ядра. На основании информации, содержащейся в таблице процессов, операционная система осуществляет планирование и синхронизацию процессов. В дескрипторе прямо или косвенно (через указатели на связанные с ним структуры) содержится информация о состоянии процесса, расположении образа процесса в оперативной памяти и на жестком диске, о значении отдельных составляющих приоритета и его итоговом значении — глобальном приоритете. Кроме того, дескриптор содержит идентификатор пользователя, создавшего процесс, информацию о родственных процессах, о событиях, осуществления которых ожидает данный процесс, и другую информацию.

Контекст процесса содержит менее оперативную, но более объемную часть информации о процессе, необходимую для возобновления выполнения процесса с прерванного места: содержимое регистров процессора, коды ошибок выполняемых процессором системных вызовов, информацию о всех открытых данным процессом файлов, о незавершенных операциях ввода/вывода и другие данные, характеризующие состояние вычислительной среды в момент прерывания.

Контекст так же, как и дескриптор процесса, доступен только программам ядра, то есть находится в виртуальном адресном пространстве операционной системы, однако он хранится не в области ядра, а непосредственно примыкает к образу процесса и перемещается вместе с ним, если это необходимо, из оперативной памяти на диск. В UNIX для процессов предусмотрены два режима выполнения: привилегированный и обычный. В привилегированном режиме выполняются функции ядра системы, а в обычном режиме — программы пользователя.

Файловая подсистема UNIX обеспечивает единый интерфейс доступа к данным, расположенным на накопителях, и к периферийным устройствам. Одни и те же функции могут использоваться при записи данных на жесткий диск и при выводе информации на принтер.

Программы, выполняемые под управлением системы UNIX, не содержат никакой информации относительно внутреннего формата, в котором ядро хранит файлы данных, данные в программах представляются как бесформатный поток байтов. Программы могут интерпретировать поток байтов по своему желанию, при этом любая интерпретация никак не будет связана с фактическим способом хранения данных в операционной системе.

В последнее время все более популярными становятся реализации операционной системы UNIX для персональных компьютеров. Одной из таких реализаций является Linux. Linux — это оригинальная реализация UNIX для платформы Intel, выполненная молодым сотрудником университета Хельсинки Торвальдом Линусом. Linux распространяется свободно, является очень экономичной операционной системой.

Linux поддерживает большинство свойств, присущих другим реализациям UNIX, и является полной многозадачной и многопользовательской операционной системой. Большинство свободно распространяемых по сети Интернет программ для UNIX можно откомпилировать для Linux практически без особых изменений. Кроме того, все исходные тексты для Linux, включая ядро, драйверы устройств, библиотеки, пользовательские программы и инструментальные средства, распространяются свободно.

Другой специфической внутренней чертой Linux является поддержка национальных и стандартных клавиатур динамически загружаемыми драйверами, что делает эту систему более универсальной.

Linux поддерживает различные типы файловых систем для хранения данных. Некоторые файловые системы были созданы специально для Linux. В Linux реализована также файловая система MS DOS, позволяющая прямо обращаться к файлам MS DOS на жестком диске, а также файловая система ISO 9660 CD-ROM для работы с дисками CD-ROM. Linux обеспечивает полный набор протоколов TCP/IP для работы в сети.

Ключевые понятия

- операционная система;
- файл;
- каталог;
- файловая система;
- кластер;
- сектор;
- FAT16 и FAT32;
- BIOS;
- драйвер;
- команды операционной системы и командные файлы;
- CONFIG.SYS и AUTOEXEC.BAT;
- форматирование;
- дефрагментация диска;
- кэширование диска;
- архивация;
- вирусы и антивирусные программы;
- UNIX;
- Linux.

Контрольные вопросы

1. Что такое операционная система?
2. Какие принципы классификации операционных системы вы можете назвать?
3. Что такое вытесняющая многозадачность? В чем ее отличие от невытесняющей?
4. Какие задачи решает файловая система?
5. Сформулируйте основные принципы организации файловой системы FAT.
6. Что означает термин «потерянное место» (slack)? Почему оно возникает?
7. Что такое BIOS и каковы основные его функции?
8. Основные составные части MS DOS.
9. Дайте определение и перечислите основные функции драйвера.
10. Как создать командный файл?
11. В чем состоит назначение файла конфигурации CONFIG.SYS?
12. Что такое операционная оболочка?
13. В чем особенности операционной оболочки Windows 3.1?
14. В чем состоят основные преимущества Windows 95?

15. Перечислите так называемые «особые возможности» Windows 95.
16. Сравните возможности Windows 95 и Windows 98.
17. В чем особенности операционной системы Windows 2000?
18. Основные служебные программы по обслуживанию дисков.
19. Что представляет из себя и для чего используется процедура дефрагментации диска?
20. Что такое вирус. Какие классы вирусов вы можете перечислить?
21. Перечислите известные вам антивирусные программные средства.
22. Перечислите принципиальные преимущества операционной системы UNIX.
23. Приведите основные характеристики операционной системы LINUX.

Литература

1. Информатика / Под ред. Н. В. Макаровой. М.: Финансы и статистика, 1997.
2. Олифер В. Г., Олифер Н. А. Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2000.
3. Основы экономической информатики/ Под ред. А. Н. Морозевича. ООО «Мисанта». Минск, 1998.
4. Роутледж Д. Р., Валнум К. Ваш персональный компьютер. М.: Бином, 1995.
5. Саймон Б. Файловая система FAT32 для Windows 95// PC Magazine. 1997, April 8. P. 279.

Источники в Интернете

1. http://www.symantec.com
2. http://www.kasperskylab.ru
3. http://www.winzip.com
4. http://www.netzip.com

Глава 5. Обработка текстовой информации

В этой главе мы рассмотрим основные понятия, задачи и методы обработки текстовой информации с помощью персональных компьютеров. Тема эта очень обширна и разнопланова, поскольку большинство воспринимаемой человеком информации так или иначе передается в виде текста. Постараемся максимально сжато и лаконично обсудить приемы работы с текстовыми документами.

5.1. Задачи обработки текстовой информации

Делить на отдельные части что-либо целостное, пусть и многообразное, всегда сложно, поэтому первый раздел посвятим классификации и описанию самых общих задач обработки текстовых документов. Выделим несколько этапов обработки документов:

- ввод текста;
- редактирование;
- сохранение документа;
- публикация;
- поиск и открытие созданного документа;
- перевод.

5.1.1. Ввод текста

На данном этапе развития научной и технической мысли ввод текста может осуществляться несколькими разными способами:

- набором текста при помощи клавиатуры;
- переводом бумажных документов в электронную форму;
- голосовым вводом;
- рукописным вводом.

Набор текста при помощи клавиатуры. Этот способ в настоящее время является самым распространенным. Он представляет собой, как правило, довольно простой технологически процесс, если, конечно, не стараться сразу придать документу окончательно оформленный вид, а заняться этим при редактировании. Но в любом случае, в зависимости от пожеланий заказчика и вида последующей публикации наборщик должен правильно выбрать шрифт, который он будет использовать при наборе. Поэтому, говоря о наборе текста, прежде всего поговорим о шрифтах.

Шрифты в современном понимании явились результатом многовековых разработок. Современное книгопечатание начало развиваться приблизительно с XV века. В XV–XVI веках многие художники занимались разработкой шрифтов. Уже в то время шрифты приобрели вполне законченный современный вид. Некоторые шрифты того времени применяются до сих пор. Примером может служить шрифт Клода Гарамона, на основе которого создана оцифрованная и русифицированная версия шрифта ITC Garamond.

В последнее время для изготовления шрифтов и представления документов стали использоваться компьютеры. Первыми по-настоящему компьютерными шрифтами можно считать *векторные* и растровые (или матричные) шрифты. Они различаются по способу описания символов. Символы векторных шрифтов задаются в виде набора векторов, определяющих перемещение пера графопостроителя по бумаге или электронного луча в трубке векторного дисплея. Эти шрифты имели ограниченный круг применения, связанный с выводными устройствами векторного типа.

Растровые шрифты предназначены, в основном, для вывода на экран и на довольно-таки устаревшие растровые устройства, к которым относятся, например, матричные принтеры. Для печати с помощью лазерных принтеров либо для электронных публикаций их не применяют по той простой причине, что, во-первых, на разных экранах они будут выглядеть по-разному, во-вторых, качество печати и электронный вид документа, как правило, оставляют желать много лучшего. Символы растровых шрифтов задаются как комбинации точек в матрице заданного размера (в растре), что обусловливает ряд недостатков, и главный из них — неприспособленность к трансформациям, таким как масштабирование, поворот, наклон. Размеры растровых шрифтов могут иметь лишь ряд фиксированных значений, поскольку они определяются размерами матрицы (растра), на базе которой построены символы. Изменение размера или начертания шрифта происходит путем замены одного растра на другой. С другой стороны, преимуществом растровых шрифтов может быть названо то, что для них программная процедура вывода на печать является более простой и быстрой.

Контурные шрифты являют собой попытку сделать максимально удобные для пользователей шрифты на основе и идеологии векторных. В принципе, в литературе их часто не различают из-за принципиально одинаковой организации описания символов. Отличие лишь в том, что здесь символы задаются в виде набора отрезков и гладких кривых, имеющих точное математическое описание. У контура гораздо больше возможностей для трансформаций, он легко масштабируется.

Для вывода символов на растровое устройство необходима специальная программа — растеризатор. Сейчас контурные шрифты применяются наиболее широко, хотя и поддерживаются не всеми платформами. Удобство их в том, что применение их при подготовке документов позволяет более или менее реализовать принцип *WYSIWYG* (*What You See Is What You Get — «То, что ты видишь, есть то, что ты получишь»*). Это принцип единообразного вида документа при наборе в данный момент и при любой дальнейшей публикации.

Широкое распространение контурные шрифты получили после 1985 г., когда фирма Adobe — признанный мировой лидер в области производства программного обеспечения для издательской деятельности — выпустила шрифты стандарта *PostScript Type 1* вместе с языком описания страниц PostScript.

Другой распространенный сейчас стандарт контурных шрифтов — это *TrueType*, совместная разработка фирм Apple и Microsoft. Шрифты *TrueType* могут поддерживать до 65535 ($2^{16}-1$) символов. Основное отличие их от шрифтов *Type 1* в том, что Adobe положила в основу построения контура кривые третьего порядка, а не второго. А если быть более точным, то так называемые кривые Безье, являющиеся, в определенном смысле, упрощением и усовершенствованием кривых третьего порядка.

С одной стороны, это обеспечивает шрифтам *Type 1* ряд преимуществ:

- символы шрифтов *Type 1* являются более гладкими из за отсутствия изломов в точках сопряжения фрагментов;
- для задания контура с тем же или более высоким качеством требуется меньшее количество точек, что уменьшает количество хранимой и передаваемой информации.

С другой стороны, поскольку нас как пользователей интересует, в основном, операционная система Windows и соответствующие ей платформы, то в этой среде основными являются шрифты стандарта *TrueType*, так как они не нуждаются в специальных преобразователях и могут применяться в большинстве приложений. На основе стандартов *TrueType* и *Type 1* различные фирмы создают конкретные виды шрифтов.

Осуществлять настройку шрифтов, идущих в комплектации с OS Windows, можно следующим образом: в главном меню Windows через Start (Пуск) выберите команду Settings (Настройки) ▸ Control Panel (Панель управления) и в появившемся окне щелкните двойным щелчком левой кнопки мыши на значке Fonts (Шрифты). В окне (рис. 5.1) появится список шрифтов, из которого вы можете выбирать наиболее подходящие для работы. По мере того, как вы выбираете какой-либо из шрифтов, вам автоматически предлагается посмотреть, как он выглядит на экране, в зависимости от размера. Вы можете ознакомиться с тем, как выглядят печатные и прописные буквы, цифры, знаки препинания, набираемые с использованием этого шрифта, а также принять к сведению характеристики шрифта и то, к какому стандарту он относится. Эта информация также может пригодиться для предсказания возможного «поведения» данного шрифта в рамках подготавливаемой публикации. Ее можно использовать также при разработке собственного, авторского

шрифта. Однако на практике к этому прибегают крайне редко, тем более что набор шрифтов, входящих в стандартную комплектацию, достаточно широк и разнообразен.

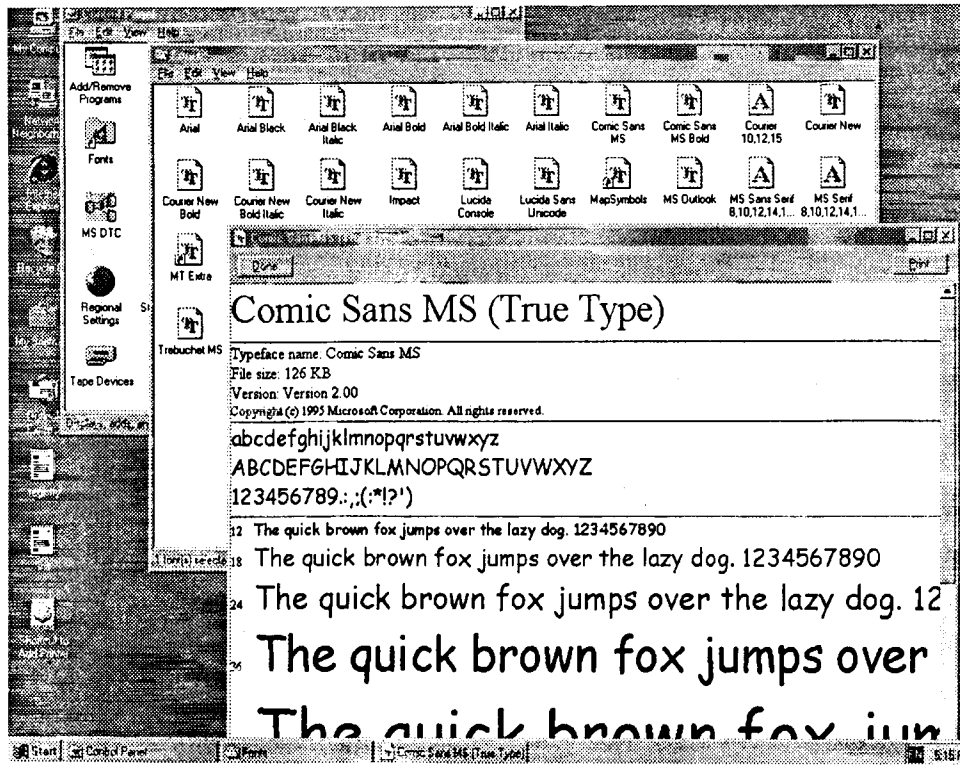


Рис. 5.1. Диалоговое окно настройки шрифтов

Вообще говоря, производство красивых и функциональных шрифтов — занятие довольно трудоемкое, творческое и не дешевое. Контур символов — это еще не все. Из одного набора контуров можно построить различные по качеству шрифты. Имеется масса параметров, таких как кернинг, трекинг, хинты и т. п., о существовании которых обыкновенный использующий шрифт человек даже и не догадывается, но именно они позволяют набранному тексту выглядеть ровно и красиво. В мире существует лишь несколько фирм, занимающихся этим видом деятельности, в России — одна. В последнее время стало модным создание эксклюзивных шрифтов, под конкретного заказчика. Некоторые фирмы — FontShop (Германия), «ПараГраф» (Россия), Signature Software (США) и др. — предлагают своим клиентам услуги по созданию персональных рукописных шрифтов. Обычно клиенту предлагается бланк, на котором он пишет как отдельные буквы, так и целые фразы. Затем эти изображения сканируются, вводятся в компьютер и с использованием профессиональных шрифтовых программ преобразуются в компьютерный шрифт. Процесс дорогой и трудоемкий, так что собственный шрифт не каждому по карману.

Перевод бумажных документов в электронную форму. Необходимость такого преобразования обусловлена тем, что многие текстовые документы имеют вид бумажных изданий: бланки платежных поручений, деловые письма, финансовая документация и многое другое. Огромные массивы информации передаются в виде трудноразличимых факсов и ксерокопий. Для того чтобы иметь возможность быстро и правильно редактировать такие документы, а также публиковать их в дальнейшем в разном виде, и возникает потребность в программном обеспечении, обеспечивающем создание электронного образа бумажного документа, а также в соответствующих устройствах.

На сегодняшний день такими устройствами являются сканеры. Не вдаваясь в техническую сторону дела, отметим лишь, что условно их можно разделить на две категории: универсальные и узкоспециализированные, к которым предъявляются специальные профессиональные требования. Нас как пользователей интересуют, в основном, сканеры универсальные, которые, в свою очередь, будем делить на ручные и автоматизированные. Ручные неудобны в использовании и чаще всего не могут обеспечить удовлетворительное качество сканирования. Из автоматизированных предпочтительнее так называемые планшетные, позволяющие сканировать книги в неразобранном виде. Кроме того, они дают возможность сканировать включенные в документ графические объекты различной природы.

Непосредственным результатом работы сканеров является так называемый графический образ информации. Таким образом, в случае работы с текстами мы на выходе процесса сканирования получаем не структурированные данные, логически разбивающиеся на фразы, слова и символы, несущие самостоятельную смысловую нагрузку, а всего лишь переведенную в электронную форму информацию о внешнем виде отсканированного текстового фрагмента. В результате возникает чрезвычайно сложная задача восстановления смыслового содержания текста по его изображению. Для ее решения предназначен специальный класс программного обеспечения, получивший название программ распознавания образов — OCR (от англ. *optical character recognizer*). В принципе, на рынке ПО присутствует немало пакетов данного назначения, но среди них реально удовлетворяют требованиям пользователя лишь несколько. В России явным лидером выступает пакет *Fine Reader.*

Системы распознавания используют одну из трех базовых технологий: шаблонную, признаковую или структурную. Шаблонные системы требуют предварительного обучения (при этом они, в принципе, не могут распознать объекты, слегка отличающиеся от шаблона), признаковые системы менее чувствительны к формам. Наконец, в структурных системах процесс распознавания строится на принципах выделения и обработки отдельных элементов изображения, поэтому они являются практически «шрифтонезависимыми».

Разработчиком пакета *FineReader* явилась российская компании ABBYY. Первая коммерческая версия была выпущена на рынок в июле 1993 г. Она была более или менее шрифтонезависимой, работала под Windows и распознавала двуязычные англо-русские тексты. На тот момент все эти свойства явились новинкой для нашей страны. Однако через полгода на российском рынке появился первый заметный конкурент — система *CuneiForm* компании Cognitive Technologies. Она

и сейчас пользуется заметным спросом. Поэтому разработчики стали кардинально пересматривать систему, и в 1996 г. вышла версия *FineReader 3.0*. В ней была улучшена точность распознавания, реализованы пакетный режим, возможность работы с немецким, французским и украинским языками. По утверждению разработчиков, теперь удалось создать технологию, которая в большей степени обладает преимуществами и в меньшей степени — недостатками всех трех базовых технологий. По-видимому, это действительно так, поскольку пакет *FineReader* сейчас занимает львиную долю рынка.

Голосовой и рукописный ввод. Голосовой и рукописный способы пока не получили широкого применения, хотя изобретены и прошли первые испытания уже около 30 лет назад и имеют ряд специфических преимуществ. Основной причиной этого явилась дороговизна их разработки и реализации.

Голосовой ввод используется довольно давно в специализированных, в основном технических областях: управляющие системы, робототехника. В последние несколько лет в связи с сильным удешевлением его стали применять, например, в телефонии. Так, многие сотовые телефоны сейчас управляются голосом.

Появляются разноплановые системы распознавания речи. Одним из примеров может являться система автоматического распознавания русской речи «Горыныч».

Данная программа позволяет осуществлять голосовой ввод текстов в компьютер в любом редакторе, работающем под операционными системами Windows 3.xx/95/NT, а также осуществлять голосовое управление компьютером. В качестве ядра системы используется самая, пожалуй, известная и широко используемая американская программа Dragon Dictate.

Скорость голосового набора текстов зависит от производительности вашего компьютера и может достигать 500—700 печатных знаков в минуту, что значительно превышает скорость «слепого» метода печати. При этом система «Горыныч» осуществляет автоматический контроль правописания: в текстах, введенных с ее помощью, исключены орфографические ошибки. Во время работы вы по-прежнему можете пользоваться клавиатурой и мышью.

Рукописный ввод довольно специфичен. Он коммерчески внедряется всего лишь последние 2-3 года и реализуется, как правило, в небольших портативных компьютерах, по размеру схожих с записной книжкой, ввод данных организован с помощью устройства, называемого *электронным пером*. Его внешний вид максимально приближен к виду обычного карандаша. Такие компьютеры называются Handheld — «удерживаемый рукой». Основные платформы — Palm, PSion, Windows CE, Newton. Удобство их в том, что они легко транспортируемы, не требуют наличия клавиатуры, практически бесшумны при работе. Сейчас эта технология все быстрее набирает обороты. Так, например, PalmPilot довольно широко известен и применяется уже и в нашей стране не только профессионалами-программистами, но и специалистами иных профилей как удобная прикладная платформа, а Windows CE используется в реализации портативной системы автоматизированного перевода компании «ПРОМТ». В некоторых из компьютеров семейства Handheld реализован

и голосовой ввод. По-видимому, в ближайшем будущем рынок будет активно перераспределяться в пользу ввода без помощи клавиатуры или с минимальным ее использованием.

5.1.2. Редактирование

Мы будем понимать под редактированием изменение набранного текста и придание ему надлежащего вида, будь то простое удаление ошибочных символов, вставка текстовых массивов либо так называемое форматирование, связанное в основном с изменением параметров шрифта и абзацев. Говорить о работе с текстом безотносительно конкретного текстового процессора весьма нерационально, поэтому рассмотрим конкретные приемы и средства редактирования применительно к Microsoft Word 2000 как приложению интересующего нас Microsoft Office 2000.

5.1.3. Сохранение документа

Сохранение в одном из многообразных существующих форматов является завершающей стадией основной работы по подготовке текстового документа.

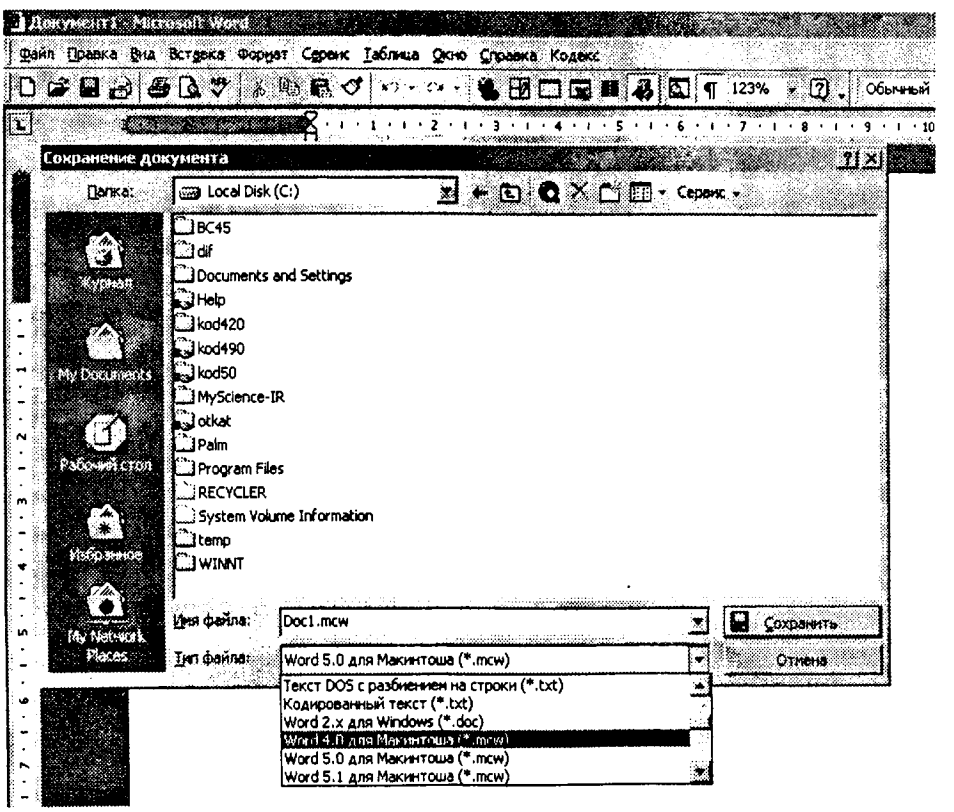


Рис. 5.2. Выбор формата сохранения документа

Это обязательный и весьма ответственный шаг несмотря на всю его кажущуюся тривиальность. Во-первых, потому что, видимо, бессмысленно выполнять какую-либо работу, не позаботившись о сохранности результата. Во-вторых, потому что выбор формата сохраняемого документа зависит от того, где и как мы собираемся дальше с ним работать. На самом деле, наибольшие проблемы возникают при открытии документа, особенно на другой платформе: вдруг оказывается, что пропало все форматирование, исчезли рисунки, текст нечитаем и т. д. Для того чтобы максимально обезопасить себя либо делового партнера, от возможных неурядиц, и стараются сохранять документ в наиболее подходящем формате. Здесь все не так просто и очевидно, поскольку, во-первых, может возникнуть необходимость работы с ним в других приложениях, не в тех, которые предполагались изначально. Во-вторых, разными текстовыми процессорами поддерживаются разные форматы, и не всегда они совместимы. В таких случаях необходимы конвертеры из одного формата в другой. Так, например, в MS Word предусмотрены встроенные конвертеры в свой внутренний формат DOC, которые мы можем подключить при инсталляции. Тогда при завершении работы с документом Word предлагает разные расширения для его сохранения (рис. 5.2).

И все же общее представление о существующих форматах текстовых документов иметь надо, поэтому, не углубляясь во все их многообразие и внутреннюю структуру, опишем основные особенности наиболее часто употребляемых форматов.

ASCII

Самый заслуженный и давно существующий формат — ASCII (American Standard Code for Information Interchange — американский стандартный код для обмена информацией). Он имеет самую простую организацию: например, в английском алфавите (впрочем, как и в русском) одной букве соответствует один байт.

Чтобы понять, почему появление в 1963 году кода ASCII сыграло столь значительную роль, нужно иметь в виду, что до этого различные компьютеры просто-напросто не могли взаимодействовать друг с другом. Каждый производитель пытался по-своему представить символы алфавита, цифры и управляющие коды.

Код ASCII стал общим знаменателем для компьютеров, которые ранее не имели друг с другом ничего общего. Всем буквам, цифрам, знакам препинания и другим символам (управляющим кодам) были поставлены в соответствие стандартные числовые значения. К примеру, заглавная буква «А» обозначалась числом 65.

Однако 60-е еще не отличались высоким уровнем стандартизации. В одних только аппаратных средствах корпорации IBM использовалось девять различных наборов кодировки символов.

Между тем взаимодействие между компьютерами стало настоятельной необходимостью. В 1961 году будущий изобретатель ASCII принял предложение представителей Американского национального института стандартов (ANSI). Возглавляемый вице-президентом компании Teletype Джоном Аувертером комитет ANSI X3.4, в котором была представлена большая часть производителей компью-

теров, приступил к работе. Комитету понадобилось свыше двух лет, чтобы проанализировать позиции всех сторон, найти компромисс и завершить разработку универсального кода.

Сегодня на основе кода ASCII выпускается оборудование стоимостью в миллиарды долларов, большинство операционных систем также до сих пор совместимо с ASCII. Причем в разных операционных системах (ОС) осуществлена несколько разная реализация этого формата. Интересный пример — кодировка Перевода строки (ПС): в ОС семейства UNIX это просто (ПС), в ОС MS-DOS и Windows — (ПС)+(ВК). В результате можно наблюдать интересную картину: созданный под UNIX текст, сохраненный в формате ASCII, открываемый, например, в *Windows Notepad* (через Start (Пуск), выберите команду Programs ▸ Accessories ▸ Notepad (Программы ▸ Стандартные ▸ Блокнот)) или в ином простом текстовом редакторе, весь окажется склеенным в одну строку.

Тем не менее, код ASCII остался одной из немногих технологий, которой удалось успешно пройти сквозь десятилетия и дожить до наших дней.

RTF

RTF (Rich Text Format) — формат обмена документов между текстовыми процессорами. Он был разработан Microsoft в 1986 году и с тех пор существует, в определенном смысле, параллельно остальным. Появление его явилось очередной вехой в разработке средств сохранения и передачи текста. Главное его достоинство в том, что его внутренняя организация предусматривает передачу всех элементов форматирования: размера и параметров шрифта, параметров абзацев и т. д. Он очень хорошо специфицирован, описан документально, довольно просто и строго организован и хорошо распознаваем практически всеми офисными приложениями.

DOC

Пожалуй, наиболее интересующий нас формат — это формат DOC, который является, по сути, внутренним форматом MS Word. Необходимо учитывать, что MS Word 2000 и MS Word 97 полностью совместимы по формату, но более ранние версии — нет, поэтому при сохранении документа, даже если мы собираемся в дальнейшем работать с ним исключительно в MS Word, нужно обратить особое внимание на его версию. При сохранении документа в текстовом процессоре MS Word он предлагает по умолчанию именно формат DOC. Как правило, имеет смысл принять это предложение, если, конечно, нет каких-либо особенных идей относительно дальнейшей публикации либо переноса файла в другие приложения. Более подробно о сохранении документа в MS Word мы поговорим в соответствующем разделе.

HTML

Последний формат сохранения текстовых документов, о котором будет упомянуто здесь, — формат HTML (Hypertext Markup Language). HTML — универсальный язык разметки гипертекста, применяемый в Интернете для разработки Web-

страниц. Основные характеристики его будут описаны в главе 11, посвященной глобальным вычислительным сетям. Отметим лишь, что для Word 2000 формат HTML является внутренним, равноправным с исконно «родным» форматом DOC.

5.1.4. Публикация

Создав и сохранив документ, мы оказываемся перед проблемой его публикации. Публикация документа является, в определенном смысле, его визитной карточкой. Это представление документа в его окончательном, готовом виде. В зависимости от типа представления документа условно выделим три вида публикации.

- печать документа;
- электронная публикация;
- Web-документы.

Здесь договоримся пока не выделять публикацию Web-документов отдельным пунктом, а включим ее в обсуждение проблемы электронной публикации в контексте электронных книг. Более подробно об электронной почте и средствах Интернета рассказывается в главе 11.

Печать документа — это создание его твердой копии на бумаге или прозрачных пленках. Необходимость создания печатного вида документа возникает в силу ряда причин: для оформления рекламных буклетов фирмы, подготовки экономической документации, различных научных трудов, учебников, художественной литературы и пр. Печать осуществляется посредством принтеров, которые являются внешними устройствами для компьютера и подключаются к нему через один из существующих на данный момент интерфейсов либо подсоединяются непосредственно в сеть, являясь самостоятельной единицей в ней. Непосредственное подключение может производиться в том случае, если принтер очень мощный и высокопроизводительный и предназначен для работы в корпоративной сети. В противном случае необходимо подсоединение его к компьютеру. Чаще всего сейчас подключение производится через параллельный порт CENTRONIX, интерфейсы USB, IrDA, BlueTooth. Оставим техническую сторону дела и отметим лишь, что существенным отличием первых двух интерфейсов является то, что их использование предполагает обязательное наличие физических элементов соединения компьютера с принтером, тогда как остальные позволяют распечатывать текст с компьютера, просто поднеся его к принтеру на достаточно близкое расстояние.

Электронная публикация. Будем понимать под электронной публикацией окончательное представление документа в электронном виде с возможностью переноса его в том же виде другим пользователям и чтения с экрана вне зависимости от способа переноса. Мы можем отправить созданный документ другому пользователю или заказчику разными способами:

- по электронной почте;
- выложить его в Интернет как Web-страницу;
- на сменных носителях, например на дискете.

Электронные книги. Проблема электронной публикации сейчас широко обсуждается в связи с быстро развивающимися и изменяющимися средствами создания цифрового издательского инструментария. Мы читаем с экрана то, что вынуждены, — электронную почту, Web-страницы, создаваемые в текстовом процессоре документы, — но лишь потому, что другой альтернативы нет. Символы с разрешением 72 точки на дюйм трудно читаются, любой свет, попадающий на экран, порождает слепящие блики. Есть и эстетические соображения: тексты на экранах чаще всего отформатированы в соответствии с параметрами по умолчанию, с не выдерживающими никакой критики интервалами и выравниванием, а для их отображения используются стандартные, набившие оскомину шрифты семейств Times или Arial.

В новом поколении электронных книг делаются попытки преодолеть эти сложности на основе специализированных аппаратных устройств, тексты в которые загружаются электронным способом. Сейчас даже самые простые электронные книги представляют собой весьма дорогие игрушки. Хотя их создатели с этим не согласны — они считают, что, учитывая стоимость всех печатных публикаций, которые больше не нужно будет покупать, устройства для отображения электронных изданий окупаются в довольно короткие сроки. Грамотный экономист сумеет рассчитать выгодность того или иного предприятия, но все электронные книги имеют ряд положительных особенностей.

Мгновенная доставка. Электронные книги, значительно облегчающие проблему, открывают новые возможности перед современными книжными магазинами. Вы приобретаете на Web-узле текст и сразу загружаете его. Книга может быть доставлена практически мгновенно.

Легкость аннотирования. Вы можете добавлять собственные электронные замечания. Конструкция современных электронных книг позволяет выделять текст и вносить аннотации, по которым можно осуществлять поиск.

Экономическая выгода. Учитывая отсутствие затрат на печать, минимальную стоимость хранения и доставки, цена текста должна существенно снизиться. В большинстве случаев электронные версии должны обходиться намного дешевле, чем печатные издания.

Экологические преимущества. Электронные книги позволят спасти от вырубки множество деревьев. Мы сможем использовать бумагу избирательно и печатать только то, что необходимо. Газеты и другие «маложивущие» издания могут загружаться на электронные книги и впоследствии стираться, а не перерабатываться во вторичное сырье.

Уже существует готовый рынок профессиональных пользователей, таких как врачи, юристы, экономисты, каждый из которых сейчас тратит немалые деньги на справочники и журналы, причем им приходится покупать и периодически выпускаемые обновленные версии справочников. Пользуясь электронными книгами, они смогут сэкономить и время, и деньги.

Пока трудно предсказать, на что будут похожи будущие электронные книги и тексты для них. Иллюстрации, например, могут быть анимированы — так будет легче объяснить сложные процессы или просто удивить читателя. Если вы являетесь

приверженцем традиционных форм книг, то можете вообразить электронную книгу из бумаги в твердом переплете и совершенно привычного формата, в которой по вашему желанию появляется текст любого произведения. Когда текст создается в электронном виде, становится возможным многое.

Транспортировка на сменных носителях. Теперь рассмотрим особенности переноса документа, условно говоря, на дискете. Дело в том, что при таком способе могут возникнуть различные неприятности, связанные с тем, что у получателя документа не тот текстовый процессор, в котором создавался документ, не та версия операционной системы, не такая установка шрифтов и т. д. Иногда приходится на месте исправлять различные ошибки или вносить необходимые изменения, используя непривычные инструменты и технологии. Описанный выше формат RTF, казалось бы, должен являться панацеей от таких бед. Реально же он обеспечивает лишь, как уже говорилось, передачу элементов форматирования, но не средств и команд, которыми они создавались. Усугубляется проблема тем, что редкий документ содержит только текст «черным по белому». Графические объекты, таблицы, особенно в цвете, распознаются еще хуже.

В разделе, посвященном шрифтам, уже упоминался язык описания страниц PostScript фирмы Adobe. Этот язык стал фактическим стандартом передачи графического представления информации между системами разработки страниц (программами обработки изображений, верстки, текстовыми процессорами и т. д.) и выводными устройствами, прежде всего принтерами высокого разрешения. Основой языка можно считать возможность свободно комбинировать текст и графические изображения, применяя к ним одинаковый набор преобразований. Но при всей его высокой эффективности и универсальности, он никогда полностью не отвечал требованиям процесса производства документов, поскольку разрабатывался для устройств вывода, а не для их операторов. Хотя он называется языком описания страниц, PostScript скорее является языком описания документов. В файле PostScript невозможно предсказать, как выглядит страница n без обработки всех страниц с 1 по $n-1$, поскольку некоторые составляющие страницы n — шрифты, цвета и другие — могут описываться на предыдущих страницах. Это усложняет обычные для производственного процесса операции редактирования, такие как внесение последних изменений в законченные полосы или сборка персонифицированных документов из предварительно подготовленных компонентов.

Попыткой найти выход из положения явилась разработка той же фирмой Adobe формата *PDF* (Portable Document Format — переносимый формат документов), чтобы дать юридическим и экономическим фирмам, фармацевтическим корпорациям и другим организациям с большим объемом документов возможность сохранять точное форматирование страниц при обмене электронными файлами. Изначально PDF задумывался как альтернатива печатным документам: инструмент для «безбумажного офиса», который позволял бы просматривать на экране полностью отформатированные документы без помощи создавшего их приложения или установки дополнительных шрифтов. Для оперирования с PDF-файлами Adobe предложила пакет Acrobat. До выхода версии 4.0 он состоял из двух частей: Acrobat Distiller, позволявший конвертировать документы в PDF-файлы, и

Acrobat Reader, который потом эти файлы открывал для прочтения. Acrobat Reader не давал возможности редактирования файлов PDF, что резко ограничивало достоинства и возможности пакета.

В Adobe Acrobat 4.0 (рис. 5.3), заменившем прежний Acrobat Reader, появилось много новых мощных возможностей, включая способность помечать и аннотировать документы PDF, создавать формы PDF Web, выполнять редактирование текста и изображений в файлах PDF, а также интегрировать PDF-файлы в серверы Web и электронную почту. Пользователи версии Acrobat 4.0 получили дополнительный инструмент: способность применять опции безопасности и цифровые подписи. Вообще говоря, вопреки факту, что Acrobat был создан компанией, сделавшей себе «имя» на языке PostScript — а сам PDF основан на PostScript, — первоначально Adobe не позиционировала эту технологию в качестве инструмента для допечатной подготовки. Однако и многие пользователи, работающие в области издательских технологий, положительно восприняли известие о выпуске компанией Adobe версии Acrobat 4.0, что свидетельствует о широких возможностях пакета, и не только в области электронного представления документов.

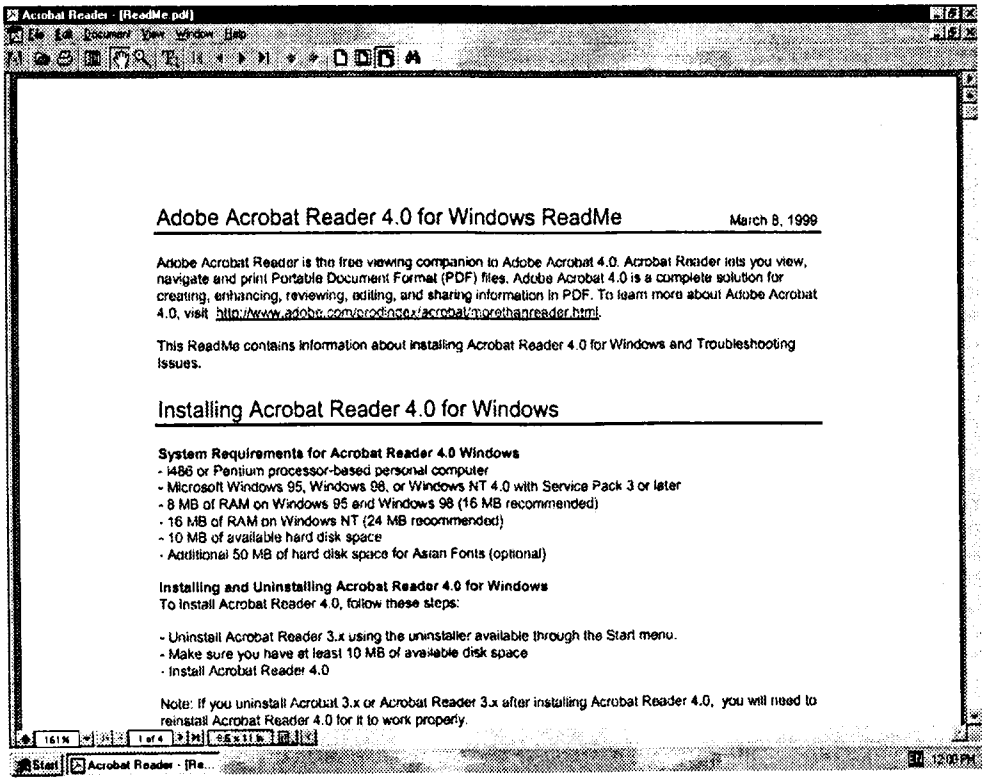


Рис. 5.3. Экранный интерфейс программы Acrobat Reader 4.0

В чем же главные преимущества применения «переносимого формата документов»? PDF является странично-ориентированным, поэтому код, описывающий каждую страницу, полон и самодостаточен. Это значительно упрощает редакти-

рование и последующее растрирование единственной страницы из длинного документа. И, что более важно, вывод файлов PDF более предсказуем и надежен, чем PostScript. Чем бы мы ни открывали документ, сохраненный в формате PDF, он везде будет выглядеть одинаково, поэтому PDF получил свое имя вполне заслуженно. Файлы PDF представляют исходный текст PostScript в виде «объектов», чьи описания составляются и фиксируются в процессе преобразования. Иными словами, файл PDF — это интерпретированный PostScript. Все, что должно быть сделано перед выводом, — это растрирование, то есть конвертирование файла в физические точки на пленке, бумаге или экране.

Значительные удобства несет в себе сжатие первоначально объемного документа в небольшой PDF-файл. Оптимизируя документ в формате PDF, мы можем задавать, что еще тот будет содержать помимо исходного текста и графики: информацию о шрифтах, выборе разрешающей способности, иных средствах создания и оформления документа — или же только текст самого документа. Чем больше информации о создании документа мы закладываем, тем больше окажется итоговый PDF-файл, и наоборот, если мы поместим в него только внутреннее содержимое документа, размер его резко сократится. Выбор конкретной комплектации зависит от предполагаемого вида дальнейшей публикации и от пожеланий заказчика. Как правило, в случае электронной публикации вкладывают меньше дополнительной информации, чем при выводе на печать, поскольку при создании твердой копии документа происходит большее искажение качества исходного изображения. Еще один плюс формата PDF: кроме того, что в процессе оптимизации размер итогового файла PDF значительно сокращается, одновременно выполняется значительная часть работы, которая ранее была возложена на растеризатор (RIP — Raster Image Processor), поэтому вывод файла PDF происходит значительно быстрее.

Оптимизированные файлы PDF полезны также для получения точных цветопроб, поскольку уменьшается вероятность появления различий между тем, как интерпретируют данные RIP цветоопределяющего устройства и фотонаборного аппарата. В действительности, тот факт, что файл PostScript или приложения успешно конвертируются в PDF, уже является своеобразной мини-проверкой «правильного вида» публикации, поскольку многие обычные проблемы с выводом (такие, как отсутствующие шрифты и изображения) становятся очевидными во время предварительного просмотра файла PDF на экране.

В последнее время стало появляться все больше программ от третьих фирм, позволяющих оперировать с файлами PDF.

Существует множество средств — от дополнительных модулей до отдельных наборов утилит. Мы рассмотрим некоторые наиболее полезные инструменты, доступные сегодня.

Программы просмотра.

Исходные тексты Adobe достаточно открыты, что позволяет третьим фирмам создавать программы с равными или даже более широкими возможностями, чем оригинальные продукты. Существует довольно много программ просмотра, некоторые из них могут работать в других операционных системах. Например, *!PDF* предназначен для RiscOS, *BePDF* — для BeOS, *PDFViewer* ориентирован на NeXT/

OpenStep, а *OmniPDF* читает PDF на платформе Rhapsody. Единственным кросс-платформенным вариантом является *jPDF* компании ReUse, который представляет собой написанный на Java броузер Web с поддержкой чтения PDF — устанавливать Acrobat не нужно.

Преобразование.

Существует несколько программ, которые конвертируют сложные документы в PDF-файлы. Например, пакет *DocuLex Pdf-It* может конвертировать в формат PDF огромные документы или большие группы изображений, причем в итоговых файлах будет возможен поиск текста. Дело не только в том, что программа распознает 13 различных языков, но она также определяет ориентацию полосы и выполняет проверку орфографии во всем документе.

Пакет *Robert Schifreen Gymnast* конвертирует в формат PDF любой текстовый файл. Эта утилита генерирует закладки из заголовков полосы, а также сохраняет аннотации. Тем, кто планирует конвертировать в PDF отдельные публикации, имеет смысл попробовать пакет *Gemini Studio* компании Iceni Technology, ориентированный на издание журнальных и газетных публикаций в Web.

Выборка информации.

Иногда бывает необходимо сделать выборку информации из файла PDF без изменения данных. Существует несколько инструментов, которые облегчают эту задачу как для графики, так и для текста. Zorglub.com предлагает *PDF2Doc*, удобную утилиту, которая не только извлекает текст из документа PDF, но и конвертирует его в формат текста PalmPilot, превращая файл в настоящий переносимый документ.

WinPDFdata компании SANFACE Software извлекает информацию из файла PDF, а затем создает из него документ HTML.

PDFCount компании 4Translation извлекает не только данные, но и информацию о данных. Эта программа, предназначенная для подсчета количества слов, символов и страниц, а также другой статистики по файлу PDF, весьма удобна для получения полного представления об информационном объеме файла PDF.

К сожалению, широкие возможности, предоставляемые как самой фирмой Adobe, так и различными третьими фирмами, еще очень мало используются у нас в стране, но для обеспечения высококачественной публикации скоро это станет совершенно необходимо. Экономическая документация бывает настолько сложно и разнообразно оформлена, что для переноса или публикации такого документа понадобится весь потенциал формата PDF и соответствующих пакетов, работающих с ним.

5.1.5. Поиск и открытие созданного документа

Для того чтобы возобновить работу с созданным и сохраненным документом, прежде всего необходимо его найти и открыть. В зависимости от того, был документ опубликован, как Web-страница, или сохранен в файловой системе, его можно искать разными способами: либо средствами Интернета, либо средствами,

соответственно, файловой системы. Поисковые системы Интернета будут освещены в соответствующей главе. Поиск файла средствами операционной системы Microsoft Windows осуществляется через Пуск ▸ Поиск (рис. 5.4).

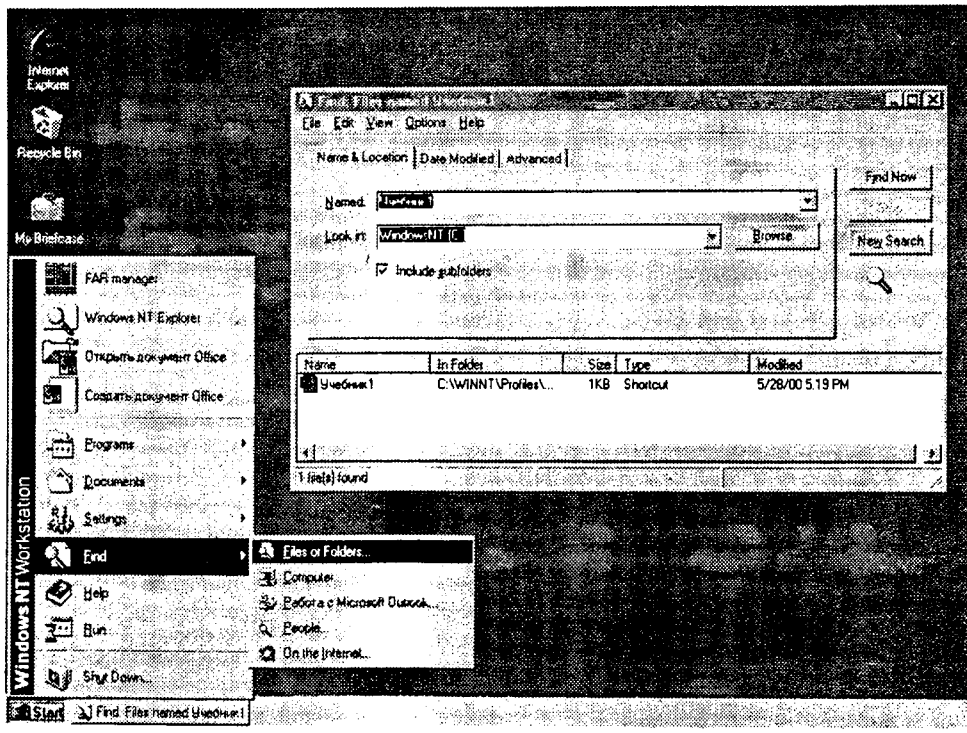


Рис. 5.4. Поиск файлов в MS Windows

5.1.6. Перевод

В настоящее время в связи с быстрым развитием Интернета как средства обмена большими объемами текстовой информации, а также в связи с необходимостью все оперативнее обрабатывать эту информацию особенно остро стоит вопрос о языке. Конечно, значительная часть информации подается на английском — общепринятом языке межнационального общения. Но и на других языках мира содержится очень большая и важная часть информации. Поэтому чрезвычайную значимость и ценность приобретает возможность межъязыковой коммуникации.

Сегодня на рынке имеется много изделий, относящихся к различным разделам языковой инженерии, занимающейся созданием инструментария, который поможет пользователям, во-первых, понимать получаемую информацию, а во-вторых, поставлять свои документы на максимально большом числе языков. Это, например, программы проверки орфографии и грамматики, программы автоматического перевода, системы диктовки, пакеты информационного поиска.

Автоматический перевод часто вызывает нарекания по поводу «глупостей», которые пишет программа, как то: перевод имен собственных, неправильная структура предложения, отсутствие связи существительного с прилагательным и т. д. Нарекания вполне закономерные, но при этом не учитывается, что автоматизированный перевод несет в себе ряд явных преимуществ, надо только умело ими воспользоваться. К числу неоспоримых преимуществ относятся быстрота и сравнительная, относительно ручного перевода, дешевизна обработки текста. Не такую очевидную, но на практике все же пользу приносит пресловутый «дословный» перевод. То, что машина не может, в отличие от профессионального переводчика-человека, предоставить высокохудожественный или хотя бы просто совершенно связный текст, безусловно, плохо, особенно для пользователей, не знающих язык. Но зато любой здравомыслящий человек, обнаружив предложенную ему нелепицу, попытается разобраться, в чем же дело и как должно быть правильно. Профессиональный же переводчик явно не станет писать бессвязный текст даже в том случае, если он неправильно его понял. Он художественно обработает его так, как посчитает нужным, и заказчик, естественно, не сможет отфильтровать ошибки сразу, а возможно, и потом. Риск возникновения ошибок повышается в случае узкоспециализированного перевода, когда требуется не просто высококлассный переводчик, но и хороший специалист в конкретной области, будь то теоретическая экономика, математическое моделирование или что-либо другое.

На самом деле, как показывает опыт, автоматизированными переводчиками надо пользоваться, в определенном смысле, как словарями-подсказчиками, то есть для создания чернового варианта текста, подлежащего в дальнейшем корректированию пользователем — специалистом в данной области. Для этого удобно разбивать экранную страницу на две части: одновременно в одной из них будет находиться содержимое исходного текста, в другой — переведенного.

В ноябре 1999 года компьютерный журнал «*PC Expert*» провел сравнительное тестирование самых известных систем машинного перевода: *Power Translator* (компания Lernout&Haspie), *Reverso/PROjectMT* (компания ПРОМТ), *Systran* (компания Systran) и *Transcend* (компания Transparent Language) для двух языковых пар (англо-французской и французско-английской). За пределами России системы компании ПРОМТ продаются под торговой маркой Reverso/PROjectMT. Было проведено обширное тестирование с использованием текстов по информатике, философии, медицине, юриспруденции, религии (использовались тексты из Библии). В ходе тестирования также переводились выдержки из художественных произведений (любовные романы и законы Мэрфи).

В табл. 5.1 показаны оценки, которые получили системы в результате тестирования. Оценки выставлялись по ряду параметров (таких как качество перевода, состав и качество словарей, способность систем к развитию и встраиванию в другие системы, а также скорость работы). Каждому параметру был присвоен весовой коэффициент, определяющий важность данного параметра. Так, качеству перевода как наиболее важному параметру был присвоен максимальный коэффициент, равный 4.

Таблица 5.1. Результаты сравнительного тестирования автоматизированных систем перевода

Параметр/Система	Power Translator	Reverso	Systran	Transcend
Англо-французское направление				
Open Sources	3,3	4,0	3,7	3,0
Перевод любовных романов	3,3	4,0	3,7	3,0
Перевод законов Мерфи	3,3	4,0	3,3	3,0
Перевод Библии	2,0	2,0	2,0	2,0
Перевод отрывков из книги «Sociéte du Spectacle»	3,0	3,5	4,0	4,0
Средняя оценка перевода	3,0	3,5	3,3	3,0
Словарь по информатике	1,3	3,8	1,9	1,6
Словарь по информатике журнала «JO»	1,7	2,0	0,4	1,1
Юридический словарь	4,5	4,5	4,1	3,1
Словарь по философии	1,3	1,7	0,4	0,7
Морской словарь	4,6	4,7	4,6	3,2
Медицинский словарь	3,2	4,1	3,4	0,9
Средняя оценка словаря	2,8	3,5	2,5	1,8
Франко-английское направление				
Фразы	3,5	4,0	4,0	3,0
Перевод новостей	3,5	3,5	3,8	3,3
Перевод отрывков из книги «Societé du Spectacle»	4,0	3,5	4,0	3,5
Средняя оценка	3,5	4,0	4,0	3,0

Сводные результаты проведенного сравнительного анализа по рассмотренным системам автоматизированного перевода приведены в табл. 5.2.

Таблица 5.2. Сводные результаты сравнительного тестирования автоматизированных систем перевода

Параметр/ Система	Power Translator	Reverso	Systran	Transcend	Коэффициент
Качество перевода	3,2	3,8	3,7	3,0	4
Словарь	2,8	3,5	2,5	1,8	2
Способность системы к развитию	4,0	4,0	2,0	2,0	1
Скорость перевода	0,4	5,0	2,2	2,9	1
Эргономика/ Встраиваемость	3,0	4,0	3,5	1,0	1
Итоговая оценка	2,9	3,9	3,0	2,4	

PROMT (ПРОМТ) – единственная российская компания из представленных. Безусловно, любое сравнительное исследование в той или иной мере субъективно, а порой и не беспристрастно, но все же работа серьезного независимого журнала представляет определенный интерес. Интерес не в том плане, кто «лучше», а кто «хуже», общеизвестен тот факт, что мировым лидером, владеющим львиной долей рынка перевода, является Systran. В России же практически монополистом выступает ПРОМТ. Все сильно зависит от условий, в которых происходит работа с программой. Данное исследование полезно с точки зрения предоставления информации:

- во-первых, о том, какие переводы может выполнять автоматизированный переводчик;
- во-вторых, насколько качественно или некачественно он может это делать в среднем;
- в-третьих, на что, на какие критерии и проблемы нужно обращать особое внимание, пользуясь такой системой.

В определенном смысле, недостатком данного исследования для нас является выбор направления перевода: англо-французское. Но в данном случае это не столь критично, поскольку, вне зависимости от выбранного направления, картина представлена вполне адекватно. К сожалению, аналогичного подробного и качественного отечественного исследования пока нет и не предвидится по объективным причинам, прежде всего потому, что в России, как уже говорилось, PROMT реальных конкурентов не имеет.

Рассмотрим основные характеристики пакета PROMT. Последняя версия пакета PROMT представлена четырьмя вариантами системы:

- PROMT 98 Гигант;
- PROMT Internet;
- MAGIC GOODDY;
- Pokcet PROMT.

PROMT 98. Версия 4.0. Профессиональная система перевода

Направления перевода:

- английский – русский – английский;
- немецкий – русский – немецкий;
- французский – русский – французский.

Весной 1999 года ПРОМТ освоила новое направление перевода: итало-русское. Все, кто сталкивается с проблемами перевода с итальянского языка на русский, теперь могут приобрести новую итало-русскую систему.

PROMT 98 – профессиональная 32-разрядная система для автоматизированного перевода документов, наследующая все лингвистические и интерфейсные разработки систем машинного перевода семейства STYLUS. Система реализована в технологии «Гигант», что позволяет работать в одной оболочке с несколькими язы-

ковыми направлениями перевода. Она состоит из нескольких основных модулей, которые могут быть связаны с помощью специального приложения «Интегратор»:

«Интегратор» обеспечивает выполнение основных функций системы PROMT 98 и облегченный запуск модулей, реализован в виде плавающей панели на панели управления или в виде значка на панели задач.

Модули целенаправленно ориентированы на профессиональное решение конкретных задач и предоставляют следующие основные возможности:

- *File Translator* — профессиональное приложение для пакетной обработки большого количества документов;
- *WebView* — революционное решение для Интернета;
- *PROMT* — уникальная лингвистическая среда переводчика;
- *QTrans* — дополнительный переводчик для небольших текстов, передаваемых из Clipboard или набираемых с клавиатуры.

Таким образом, PROMT 98 — это гибкая интегрированная среда, обеспечивающая следующие возможности:

- интеграцию 4 специализированных модулей в одной системе;
- полнофункциональные кнопочные меню переводчика в последних выпусках Microsoft Office.

PROMT Internet

PROMT Internet — это интегрированный пакет для работы в Интернете. Обеспечивает перевод с английского, немецкого, французского языков на русский язык и обратно. А также, как упоминалось ранее, теперь поддерживает итальянское направление.

PROMT Internet объединяет 3 системы в одном пакете:

- модуль перевода в Microsoft Internet Explorer;
- WebTranSite 98 — удобный переводчик как в броузере, так и в большинстве приложении Windows;
- WebView — броузер с синхронным переводом Web-страниц.

Также он реализует несколько полезных функций для работы в Интернете, как то: поиск в Интернете, открытие Web-узла. Пользователям, уже знакомым с пакетом и использующим его, будет интересно, что не только не пропал, но, напротив, был усовершенствован комплекс уже существовавших и функционировавших удобных программ:

- *WebTranSite 98* — обеспечивает мгновенный перевод в Netscape Navigator 3.0, 4.0 и Microsoft Internet Explorer 3.0, 4.0 и позволяет переводить в Word, Excel и множестве других программ, включая перевод справки;
- *WebView* — это броузер с синхронным переводом Web-страниц.

Основные его функции:

- синхронный перевод Web-страниц;
- переход по ссылкам как в окне оригинала, так и в окне перевода;

- сохранение в виде файла как оригинала, так и перевода;
- вывод на печать содержимого оригинала и перевода;
- копирование в Clipboard содержимого оригинала и перевода;
- возможность изменения направления перевода;
- возможность изменения списка используемых словарей;
- работа со словами, не требующими перевода;
- работа со списком незнакомых слов;
- возможность формирования запроса к поисковым серверам на родном языке.

Также в нем доступны основные функции Microsoft Internet Explorer, что облегчает и ускоряет работу с переводчиком.

Поиск в Интернете. Вам достаточно сформировать запрос на родном языке, а система переведет его на указанный язык и отправит на один из наиболее популярных поисковых серверов. Три уровня формирования запросов — «Простой», «Сложный», «Профессиональный» — помогут вам быстро создать запрос практически любой сложности, даже если вы никогда не делали этого раньше.

Открыть WWW*-узел.* Эта функция позволяет:

- открыть WWW-страницу, используя броузер, имеющийся на вашем компьютере (например, Internet Explorer или Netscape Navigator);
- открыть и перевести WWW-страницу, используя броузер-переводчик WebView.

Подключение дополнительных специализированных словарей. Качество перевода можно значительно улучшить, если использовать при переводе текстов:

- *специализированные словари,* разработанные для системы перевода PROMT;
- *пользовательские словари,* созданные самим пользователем в системе PROMT.

В некотором смысле, PROMT является *самообучающейся системой,* и в этом одно из главных ее достоинств. Суть состоит в том, что пользователь по ходу перевода может исправлять не устраивающие его варианты, слова, словосочетания, добавляя правильный вариант в словарь, и PROMT в дальнейшем будет распознавать такой же кусок текста так, как надо пользователю.

В комплект поставки PROMT Internet входят англо-русский и русско-английский словари с терминологией Интернета, предназначенные для перевода текстов в Интернете. Другие специализированные словари могут быть приобретены отдельно.

Коллекция специализированных словарей, предлагаемых компанией ПРОМТ, представлена в табл. 5.3. Около названия каждого словаря, представляющего специализацию перевода, в скобках стоят направления перевода[1]. Их имеет смысл принять к сведению как базовый, или минимальный вариант, поскольку ПРОМТ постоянно совершенствует и расширяет возможности своей системы, в том числе и за счет словарей и новых направлений перевода.

[1] Используются следующие обозначения: а — английский, р — русский, н — немецкий, ф — французский.

Таблица 5.3. Коллекция специализированных словарей ПРОМТ

Группа	Словари
Коммерция	Коммерция (а—р—а, н—р—н, р—ф)
	Информатика (а—р, н—р)
	Юридический (а—р—а, н—р—н, р—ф)
Наука	Информатика (а—р—а, н—р)
	Математика (а—р—а)
	Физика (а—р—а)
	Химия (а—р—а)
	Биология (а—р)
	Медицина (а—р—а)
Промышленность	Автомобильный (а—р, н—р)
	Строительство (а—р)
	Добыча нефти и газа (а—р—а)
	Машиностроение (а—р—а)
	Химия (а—р—а)
	Горно—технический (а—р—а)
	Металлургия (а—р)
	Полиграфия (а—р)
Техника	Электротехника и энергетика (а—р—а)
	Телекоммуникации (а—р—а)
	Военно—политический (а—р—а)
	Морской (а—р—а)
	Аэрокосмический (а—р—а)
	Авиационный (а—р—а)
	Информатика (а—р, н—р)
Домашняя	Автомобильный (а—р, н—р)
	Информатика (а—р)
	Бытовая техника (а—р)
	Кино и масс—медиа (а—р)
	Спорт (а—р)
	Кулинария (а—р)
	Музыка (а—р)
	Религия (а—р—а)
	Парфюмерия и косметика (а—р, ф—р)
	Путешествия (а—р—а)

При переводе конкретного текста имеет смысл подключать несколько наиболее подходящих словарей, выбирая приоритет каждого из них; например, для

экономиста, исследующего сегмент строительного рынка, это может выглядеть примерно так:

1) коммерческий;
2) юридический;
3) строительство.

Magic Gooddy

Magic Gooddy — это новый мультимедийный англо-русско-английский переводчик компании ПРОМТ. Gooddy быстро переводит фрагменты текста, который перетаскивается мышью на его изображение.

Редактор этого переводчика предоставляет как стандартные, так и сравнительно новые, малореализуемые возможности:

- подготовить и отправить письмо по электронной почте;
- перевести и сохранить текстовый файл;
- создать письмо с анимацией;
- перевести текст, набранный с клавиатуры.

Также он может распознавать команды, отдаваемые ему голосом. Словарный запас Gooddy на настоящий момент насчитывает более 500 тыс. слов. Этим переводчиком удобно пользоваться, например, «в домашних условиях», создавая не очень громоздкие, но нестандартно оформленные документы.

Pocket PROMT (1.0)

POCKET PROMT (1.0) — 32-разрядная интегрированная система перевода для операционной системы Windows CE.

POCKET PROMT обеспечивает связный перевод текстов с английского языка на русский и обратно. Система построена на новейших лингвистических разработках компании ПРОМТ. Она поддерживает возможности, предоставляемые портативными компьютерами. Это очень удобно тем, кто вынужден находиться в частых разъездах и одновременно связан с переводом больших или сложных текстов.

5.2. Обработка документов средствами текстовых процессоров

5.2.1. Основные понятия и характеристики текстовых процессоров

Сейчас работа с текстовыми документами производится при помощи персональных компьютеров, что является удобным и надежным методом. Программное обеспечение, предназначенное для обработки документов с помощью компьютера, условно разделяют на две категории: текстовые редакторы и текстовые процессоры. Формально будем различать их следующим образом.

Текстовые редакторы представляют собой простейшие программы, предназначенные для самого примитивного редактирования текста. Как правило, они не имеют развитых средств форматирования. Классическим примером текстового редактора является, например, уже упоминавшийся ранее Windows Notepad.

Текстовые процессоры содержат гораздо более развитые средства создания и оформления документов. В качестве примеров рассмотрим наиболее широко применяемые на данный момент в мировой практике, и особенно в российской экономической науке, текстовые процессоры.

Если говорить об общемировой практике, то основными инструментами для подготовки текстовых документов являются процессоры Word компании Microsoft; Word Perfect фирмы Corel; а также пакет StarOffice, разработанный немецкой компанией StarDivision GMBH.

Word Perfect — широко распространенный ранее на Западе текстовый процессор. Во многих небольших офисах он использовался для решения практически всех задач. Для поклонников Word Perfect самое знаменательное событие — появление комплекта Word Perfect Office 2000, представляющего собой модернизацию Word Perfect 8. Радикальной перестройки фирма Corel не планирует, однако собирается дополнить его приложением Trellix для работы с документами в Web. Trellix превращает документы WordPerfect в привлекательно выглядящие Web-страницы, автоматически снабжая их шрифтами, цветом и навигационными кнопками, позволяющими разбивать длинные документы на удобные для работы части. Пользователям, которые создают технически сложные документы, понравятся и некоторые другие усовершенствования, в том числе возможность оперативного перехода на такие элементы, как таблицы или примечания внутри документа. Бесспорным плюсом является наличие функции RealTimePreview, которая в удобной форме показывает, как те или иные предполагаемые изменения шрифтов, выравнивания и других параметров форматирования могут повлиять на внешний вид документа.

Остальные функции помогают наладить взаимодействие между продуктами Microsoft и Corel. Пользователи WordPerfect смогут работать с расширением Word (*.doc), как с установленным по умолчанию собственным форматом файлов. По заверению фирмы Corel, приложения ее комплекса будут также с высокой степенью надежности импортировать файлы из соответствующих разделов MS Office.

Разработчики StarOffice не выделяют, как таковой, текстовый процессор, а рассматривают весь пакет в комплексе. В последнее время ведется довольно активное обсуждение пакета StarOffice, особенно в связи с операционной системой Linux. Многим кажется, что именно такого продукта не хватало долгое время ОС Linux для того, чтобы наконец утвердиться на рабочем столе экономиста, бухгалтера или секретаря и распространиться по всем компьютерам в офисах многочисленных мелких фирм по всему миру. StarOffice хвалят за сходство с MS Office, за то, что разработчики честно стали использовать фактический стандарт вместо того, чтобы обременить мир еще одним. Это полный стандартный набор программ для офиса, совместимый с большим числом различных форматов документов, включая (начиная с версии 5.0) и формат MS Word 2000.

По замыслу разработчиков, пакет StarOffice должен содержать практически все, что может потребоваться в повседневной работе на компьютере. В него входят диспетчер файлов StarDesktop, текстовый редактор StarWriter, редактор Web-страниц и по совместительству Web-броузер StarWriter / Web, электронная таблица StarCalc, база данных StarBase, редактор диаграмм StarChart, графический редактор StarImage, редактор презентаций StarImpress, программы работы с электронной почтой и новостями StarMail и StarDiscussion. Начиная с версии 4.0, все компоненты пакета интегрированы в единую большую программу.

Встроенная поддержка русского языка в StarOffice 5.0 — последней версии — не предусмотрена. Однако StarCalc, StarImage, StarWriter и StarMail поддаются некоторому подобию русификации. Для этого необходимо русифицировать среду работы StarOffice — графическую среду X Window операционной системы Unix, что зачастую является непростой задачей. Также не имеется важных элементов поддержки русификации: проверки грамматики, орфографии и пр. Поэтому в настоящее время в России этот текстовый редактор, как и остальные приложения StarOffice 5.0, практически не распространен.

5.2.2. Microsoft Word 2000 и его новые функциональные возможности

Word 2000 является наиболее часто используемым в нашей стране текстовым процессором, поскольку представляет собой приложение MS Office 2000 — разработки компании Microsoft, проводящей политику активного внедрения своих продуктов в России за счет поддержки русского языка и создания русифицированнных версий. В ноябре 1992 г. компания открыла представительство по странам СНГ, в задачи которого входят развитие рынка программного обеспечения на территории бывшего СССР, а также внедрение и локализация новейших технологий.

Справедливости ради, надо отметить, что MS Word является самым популярным текстовым редактором не только в среде MS Windows, но и в среде операционной системы MacOS на компьютерах фирмы Apple, хотя для российского пользователя этот факт представляет исключительно теоретический интерес.

Первые версии MS Word появились в середине 80-х годов. В то время этот текстовый процессор работал в среде MS DOS. Последняя DOS-версия этого продукта носила номер 5.0 и могла работать в графическом режиме. Первая Windows-версия появилась с выходом Windows 3.0. Долгое время MS Word находился «в тени» Word Perfect, но с выпуском версии 6.5 этот текстовый процессор занял лидирующее положение на рынке. Все Windows-версии этого продукта отличаются хорошей интеграцией с операционной средой и другими продуктами MS Office — пакетом офисных программ Microsoft. Начиная с 1995 года, все версии MS Word являются 32-битными приложениями. Версия 7.0 (Word 95) была первым текстовым процессором, который полностью использовал возможности новой ОС. Следующая версия, 8.0, имела ряд улучшений, основными из кото-

рых были усовершенствованные средства интеграции и настройки, а также новый формат файлов DOC. Также там впервые появилась возможность работы с форматом HTML, что позднее стало приобретать все большую важность.

Microsoft Word 2000 с самого начала разрабатывался как инструмент, с помощью которого пользователи в любой стране мира могли бы легко и просто создавать не только обычные печатные документы, но и сообщения электронной почты, а также документы для публикации вWorld Wide Web. HTML выбран в Word 2000 в качестве одного из основных форматов создаваемых файлов, что способствует дальнейшему закреплению ведущего положения этого приложения в области создания Web-документов и сообщений электронной почты.

По сравнению с предыдущими версиями Word 2000 имеет также ряд преимуществ, носящих не столь радикальный характер: улучшенные средства автоматической проверки орфографии, табличные и графические инструменты, значительно модернизированый Web-инструментарий. Некоторые из новшеств не очень приглянулись давним пользователям MS Word, хорошо освоившим особенности предыдущей версии, но это, скорее, вопрос вкуса и привычки. В любом случае, для тех, кто уже знаком с MS Word, нововведения должны быть интересны, поэтому далее мы рассмотрим основные привнесенные изменения.

Новые возможности Word 2000 по сравнению с предыдущими версиями

HTML как равноправный формат хранения файлов. Все имеющиеся в Word функции и средства (за исключением возможности создания версий документа и использования паролей) сохраняют свое действие после преобразований из формата Word DOC в формат HTML и обратно. Это означает, что Word 2000 предоставляет пользователям возможность создавать документы в формате Word, сохранять их в формате HTML и снова открывать в Word, используя для их обработки все функции, доступные в Word.

Принцип «Click-n-Type» (*«Щелкнуть и ввести»*). Пользователи могут начинать вводить текст в любом месте страницы, предварительно поместив туда курсор и дважды щелкнув кнопкой мыши. Эта функция упрощает разметку сложных документов.

Функция «Собрать и вставить». Пользователи могут скопировать в буфер обмена MS Office до 12 блоков информации, каждый из которых затем можно будет в произвольном порядке вставить нужное число раз в любой документ MS Office. Эта новая функция упрощает сбор информации, полученной из различных источников.

Система настраиваемых меню. В меню отображаются теперь только те функции и команды, с которыми пользователи работают наиболее часто. Меню можно легко развернуть, отобразив в них весь список команд; кроме того, развертывание меню может производиться автоматически в соответствии с правилами технологии. По идее, меню, подстраивающиеся к нуждам пользователя, должны способствовать устранению неразберихи, однако в данном случае это как раз тот вари-

ант, когда вы можете быть еще больше сбиты с толку, — особенно это почувствуют пользователи, привыкшие к старой (и все еще действующей) системе Office-меню.

Темы. Темы создаются и согласовываются между приложениями Word и Microsoft FrontPage, последнее из которых является специализированным средством создания и управления Web-узлами. Благодаря этому упрощается создание и редактирование с помощью Word и FrontPage согласованных по внешнему виду Web-страниц.

Автоматическое определение языка. Благодаря автоматическому определению языка, на котором пользователь производит ввод текста, в Word 2000 правильно осуществляется выбор средств проверки орфографии и грамматики, списка автозамены и других инструментов проверки правописания. Кроме того, были несколько скорректированы задаваемые вручную языковые настройки, что сделало их более доступными и удобными в работе.

Помимо совершенно новых функций, в Word 2000 были расширены и улучшены некоторые имевшиеся до этого в MS Word возможности.

Word 2000: усовершенствованные возможности

Мастер Web-страниц. В Word 2000 пользователи могут легко создавать профессионально оформленные Web-страницы и многостраничные Web-узлы при помощи специального мастера.

Предварительный просмотр Web-страниц. В процессе работы в Word 2000 можно просматривать Web-страницы в используемом по умолчанию броузере даже без их предварительного сохранения.

Оглавление. В Word 2000 генерация оглавлений может выполняться автоматически с использованием нескольких параметров, позволяющих работать в режиме интерактивного просмотра, в том числе с использованием рамок.

Гиперссылки. Word 2000 имеет усовершенствованный интерфейс гиперссылок, облегчающий создание, удаление и редактирование гиперссылок в документах.

Таблицы. Теперь поддерживаются вложенные таблицы (таблицы, размещаемые внутри ячеек таблицы), перемещаемые таблицы (текст в этом случае обтекает таблицу), размещение таблиц в любом месте страницы (в том числе друг рядом с другом) и объединение строк заголовка (строки заголовка должны включать в себя первую строку таблицы). Пользователи могут применять функцию обтекания текста вокруг графических изображений, находящихся внутри ячейки таблицы. Графические объекты Word 2000, размещенные внутри ячейки таблицы или же вне ее, обладают одинаковыми свойствами.

HTML-форматирование таблиц. В Word 2000 поддерживаются HTML-параметры форматирования таблиц, включая возможность создания вложенных таблиц, использование функции автоподбора, перемещение таблицы в любое место страницы, задание интервалов между ячейками и способа заполнения ячеек и правильное оформление трехмерных границ в формате HTML.

Новые возможности автозамены. В предыдущих версиях приложения Word исправление ошибок в написании слов с помощью функции автозамены проводилось на базе использования специального списка. Теперь функция автозамены может по ходу дела исправлять и те явные ошибки в написании слов, которые не указаны в списке для автозамены.

Средства проверки орфографии и грамматики. В Word 2000 оба этих популярных средства поверки правописания получили дальнейшее развитие благодаря добавлению новых слов в словарь, совершенствованию логики предлагаемых вариантов замены и уменьшению числа «ложных» флагов грамматических ошибок.

Средства расширения функциональных возможностей Word. Макросы

Текстовый процессор MS Word всегда отличался большим количеством функций и, главное, мощными возможностями настройки. Встроенный в текстовый процессор язык программирования VBA (Visual Basic for Applications) помогает довольно сильно расширить потенциал этого продукта за счет написания макросов — сценариев, позволяющих автоматизировать многие операции. Активизировать написание макроса можно через меню Сервис ▸ Макрос (рис. 5.5).

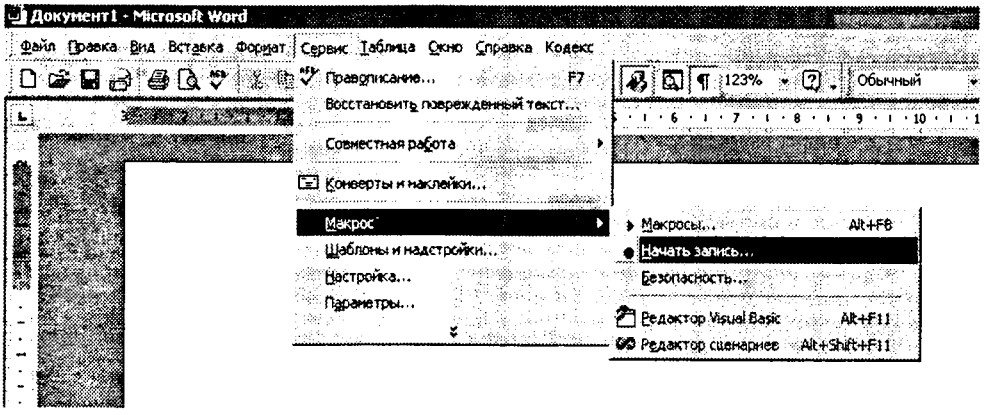


Рис. 5.5. Работа с макросами в среде MS Word

Использование макросов создает огромные функциональные возможности, но может представлять и потенциальную опасность. Использование VBA для создания макросов в Word и других приложениях MS Office позволяет:

- автоматизировать выполнение часто повторяющихся сложных операций;
- производить интеграцию между различными программными продуктами;
- настроить Word для редактирования специальным образом оформленных текстов.

Процессор Word способен сохранять макросы внутри документа, а при открытии или закрытии файла автоматически запускать их. Однако здесь есть один очень серьезный недостаток — с макросами могут распространяться вирусы.

Для того чтобы подготовить пользователя к возможным неприятностям, рассмотрим *примерный алгоритм работы* Word-*макровирусов*. Этот алгоритм хорошо знают и используют для борьбы с вирусами фирмы, выпускающие антивирусные программы.

Большинство известных Word-вирусов при запуске переносят свой код (макросы) в область глобальных макросов документа («общие» макросы). Для этого они используют команды копирования макросов MacroCopy, Organizer.Copy либо редактор макросов — вирус вызывает его, создает новый макрос, вставляет в него свой код, который и сохраняет в документе.

Первый способ проникновения вирусов заключается, в целом, в изменении общего шаблона Word NORMAL.DOT. При выходе из Word глобальные макросы (включая макросы вируса) автоматически записываются в файл *.DOT глобальных макросов (практически всегда это NORMAL.DOT). Таким образом, при следующем запуске Word вирус активизируется в момент загрузки глобальных макросов, то есть сразу.

Затем вирус переопределяет (или уже содержит в себе) один или несколько стандартных макросов (например, FileOpen, FileSave, FileSaveAs, FilePrint) и перехватывает таким образом команды работы с файлами. При вызове этих команд вирус заражает файл, к которому идет обращение. Для этого вирус конвертирует файл в формат DOT (что делает невозможными дальнейшие изменения формата файла, то есть конвертирование в какой-либо иной формат) и записывает в файл свои макросы, включая Auto-макрос.

Таким образом, если вирус перехватывает макрос FileSaveAs, то заражается каждый файл *.DOC, сохраняемый через перехваченный вирусом макрос. Если перехвачен макрос FileOpen, то вирус записывается в файл при его считывании с диска.

Второй способ внедрения вируса в систему используется значительно реже — он базируется на так называемых Add-in-файлах, то есть файлах, являющихся служебными дополнениями к Word. В этом случае NORMAL.DOT не изменяется и внешне все выглядит, вроде бы, нормально, но Word при запуске загружает макросы вируса из файла (или файлов), определенного как Add-in. Этот способ практически полностью повторяет заражение глобальных макросов, за тем исключением, что макросы вируса хранятся не в NORMAL.DOT, а в каком-либо другом файле.

Возможно также внедрение вируса в файлы, расположенные в каталоге STARTUP, — Word автоматически подгружает файлы-шаблоны из этого каталога, но такие вирусы пока не встречались.

Эти способы внедрения в систему представляют собой некоторый аналог резидентных DOS-вирусов. Вирус постоянно находится внутри Word и заражает все файлы.

Третий способ. Аналогом нерезидентности являются макровирусы, которые не переносят свой код в область системных макросов, — для заражения других файлов-документов они либо ищут их при помощи встроенных в Word функций работы с файлами, либо обращаются к списку последних редактированных файлов (Recently used file list). Затем такие вирусы открывают документ, заражают его и закрывают.

Периодические вспышки «эпидемий» заставили задуматься о последствиях, к которым могут привести макровирусы, а также о важности защиты от них. Далее приводится несколько рекомендаций по защите от макровирусов.

ВНИМАНИЕ

Заранее используйте встроенную функцию защиты от макровирусов.

Работая с Word 2000, выберите в меню Сервис ▸ Макрос ▸ Безопасность (Tools ▸ Macro ▸ Security), отметьте закладку Уровень безопасности (Security Level) и задайте опцию Высокая (High) или Средняя (Medium). После этого нажмите OK. На практике предпочтительнее выбирать опцию Средняя, поскольку Высокая может автоматически не пропускать макросы, содержащиеся в документе, без сообщения об этом. Тогда, если по каким-либо причинам пользователь не догадывается о существовании макросов в документе, он будет долго бороться с возникающими неприятностями, неправильно идентифицировав их причину. С другой стороны, включение опции Высокая практически гарантированно спасет вас от проникновения макровирусов, так что решайте в каждом конкретном случае.

ВНИМАНИЕ

Проявляйте осмотрительность при открытии документов, содержащих макросы.

Когда вы открываете файл, содержащий макрос, на экране появляется диалоговое окно. Если вы не уверены в безопасности макроса, то нажмите одну из кнопок — Отключить макросы (Disable Macros) или Не открывать документ (Do not open). В первом случае Word откроет документ с атрибутом «только для чтения» — чтобы его сохранить после редактирования, отметьте пункты Файл ▸ Сохранить как (File ▸ Save As). Редактор Word позволяет сохранить документ после редактирования, но удаляет из него макрос. Будьте внимательны: возможно, вместе с удалением макроса вы потеряете некоторые удобства и возможности по редактированию документа.

ВНИМАНИЕ

Осторожно относитесь к письмам, приходящим по электронной почте.

Следует иметь точные сведения об источнике вашей корреспонденции. Если вы не знаете, чьи это документы, то не открывайте их.

ВНИМАНИЕ

Неплохо иметь на вашем компьютере антивирусную программу.

Если вы все же боитесь заразиться каким-либо ужасным вирусом, то установите одну из существующих антивирусных программ. Но при этом надо учитывать, что и здесь есть обратная сторона медали:

- во-первых, не все антивирусные программы способны обнаружить вирусы Word;
- во-вторых, особенности их работы иногда могут раздражать, потому что работа с документом ведется медленнее, периодически приходится отвлекаться на диалог с антивирусной программой, не всегда удачный, и т. д.

Поэтому универсальное средство — сохранять бдительность. А также быть в курсе последних событий и следить за новой информацией, касающейся защиты от вирусов, которые распространяются через макросы, и за ситуацией, сложившейся в данный момент.

5.3. Работа с текстовыми документами в среде MS Word

Функциональные возможности текстового процессора MS Word, отобранные для рассмотрения в настоящем параграфе, далеко не исчерпывают всей их обширной совокупности. В основном мы остановимся на вопросах, так или иначе касающихся базовых навыков и умений, которые требуются пользователям при решении задач создания и оформления профессиональной документации. К таковым относятся:

- экранный интерфейс и настройки процессора MS Word;
- создание документов;
- ввод текста и редактирование;
- сохранение документа.

5.3.1. Экранный интерфейс и настройки процессора MS Word

Запуск процессора Word

Для того чтобы запустить процессор Word, необходимо выполнить команду Пуск ▸ Программы ▸ Microsoft Word. На экране появится окно процессора Word с открытым в нем пустым документом, которому по умолчанию присваивается имя Документ1 (рис. 5.6). Одновременно Word загружает служебный файл NORMAL.DOT, который является файлом загружаемого по умолчанию шаблона документа Пустой документ. В нем определены стандартные параметры документов в Word — шрифты и размеры символов, ширина полей на странице и пр. Установленные в шаблоне по умолчанию параметры, в принципе, можно изменить, но на практике этим пользуются не часто.

Компания Microsoft является одним из мировых лидеров по разработке пользовательских интерфейсов, и Word — это один из лучших примеров современного и развитого интерфейса. Этот текстовый процессор является Windows-приложением, и его интерфейс стандартен для этой операционной системы. Word 2000 имеет так называемый SDI (Single document interface) интерфейс — для каждого редактируемого документа открывается отдельное окно с меню и панелями инструментов.

Управление приложением осуществляется с помощью клавиатуры и мыши. Многим командам соответствует заданное сочетание клавиш, так называемые «горячие» клавиши. Элементы управления активизируются нажатием на них левой кнопки мыши, в дальнейшем мы будем называть такое действие «щелкнуть мы-

шью». При нажатии правой кнопки мыши появляется контекстное меню с командами, соответствующими элементу, над которым нажата кнопка. Когда курсор мыши задерживается над каким-либо управляющим элементом, то появляется окно подсказки с кратким описанием функций этого элемента.

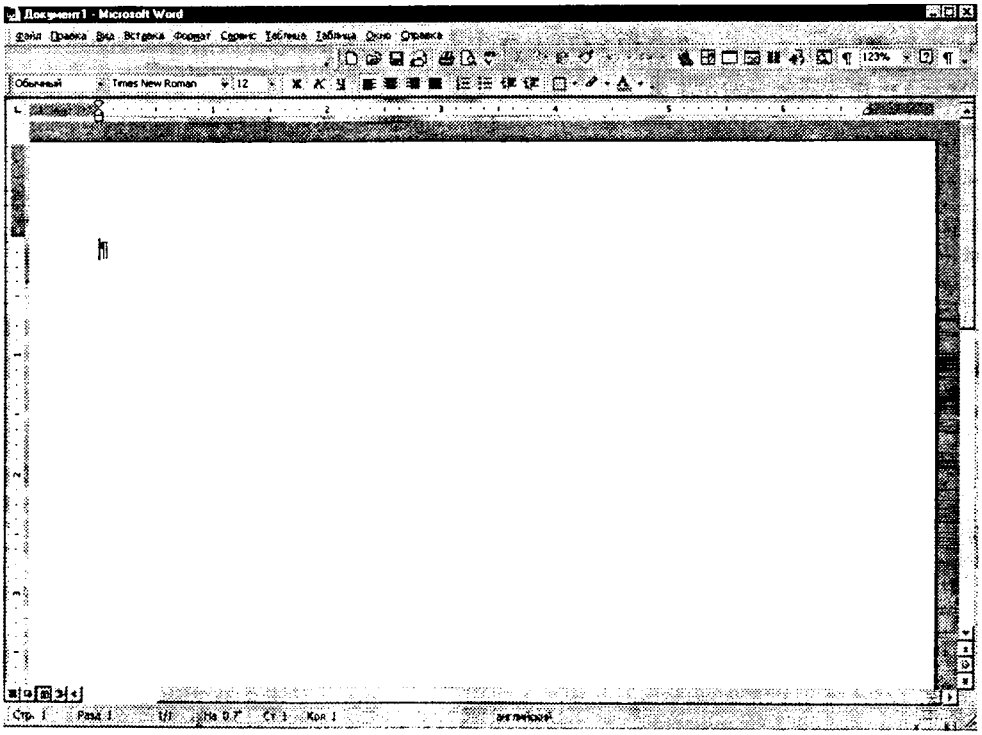


Рис. 5.6. Экранный интерфейс процессора Word 2000

При стандартных настройках окно MS Word состоит из следующих элементов:

- *Заголовок окна* — стандартный для Windows-окна. Слева вверху в строке заголовка находится значок Word, рядом — имя загруженного документа, а затем — имя програмы: Microsoft Word. Если щелкнуть мышью на значке Word, то откроется системное меню, команды которого позволяют установить подходящие размеры окна программы, свернуть или закрыть окно. То же самое можно сделать с помощью трех кнопок, расположенных на правом краю строки заголовка. Щелчок на левой кнопке обеспечивает сворачивание окна, щелчок на средней — его увеличение или уменьшение, а щелчок на правой кнопке — закрытие, то есть Word завершает работу.
- *Строка меню* расположена под строкой заголовка. Меню со списком команд открываются нажатием на них левой кнопки мыши. Word 2000 имеет «интеллектуальное» меню, которое автоматически настраивается в зависимости от последних используемых команд и метода работы пользователя. Сначала меню показывает последние использованные и наиболее часто используемые

команды. Далее оно «подстраивается» под конкретного пользователя в зависимости от того, какие команды он активизирует чаще всего. Каждая из команд тоже активизируется мышью.

- *Панели инструментов.* Ниже строки меню располагается область панелей инструментов. По умолчанию на экране отображаются панели Стандартная и Форматирование. Панель инструментов состоит из управляющих элементов — кнопок, областей ввода и выбора, что позволяет быстро выполнять команды и настройки, не обращаясь к меню. С помощью настроек можно создать или активизировать дополнительные панели инструментов и/или настроить находящиеся на них элементы управления.
- *Рабочая область* — поле, в котором отображается и редактируется документ. Вверху и слева рабочую область могут ограничивать горизонтальная и вертикальная линейки. Справа и снизу рабочая область ограничена полосами прокрутки. Конкретный вид рабочей области зависит от режима работы редактора.
- *Строка состояния.* Внизу окно программы замыкает строка состояния. Это информационная строка, в которой указывается различная информация о документе и режимах редактора, как то: число страниц, текущее положение в документе, язык, фоновые выполняющиеся операции и т. п.

Для того чтобы иметь некоторое представление о возможностях Word, кратко опишем содержание *встроенных меню*:

- Файл. Это меню содержит команды создания, открытия, сохранения и печати документа;
- Правка. Здесь содержатся команды работы с выделением части документа, буфером обмена, поиска и отмены изменений;
- Вид. Этим меню переключаются режимы отображения документа, наличие панелей инструментов и линеек, масштаб;
- Вставка. С помощью этого меню осуществляется вставка в документ различных объектов;
- Формат. Команды этого меню позволяют менять параметры форматирования текста: шрифта, абзацев, фона и др.;
- Сервис. Содержит различные сервисные операции: проверку орфографии, совместную работу над одним документом, настройку параметров программы и др.;
- Окно. Это меню дает возможность управлять параметрами окна с документом;
- Справка. Через это меню становится доступной система помощи текстового процессора.

После краткого описания меню имеет смысл так же коротко рассмотреть существующие панели инструментов. В процессоре Word 2000 в списке панелей инструментов их насчитывается 16. Пользователь по своему усмотрению может вызывать на экран и убирать любую панель, создавать собственные. Внутри панелей инструментов собраны элементы управления, представляющие операции, выпол-

нение которых необходимо при работе с документами. Таким образом, панель инструментов обеспечивает простой структурированный доступ к важнейшим функциям Word. Управление отображением панелей инструментов осуществляется в меню Вид ▸ Панели инструментов (рис. 5.7). По этой команде появляется полный список панелей инструментов. Если перед названием панели установлен флажок, то эта панель отображается на экране. Выключение приводит к скрытию панели.

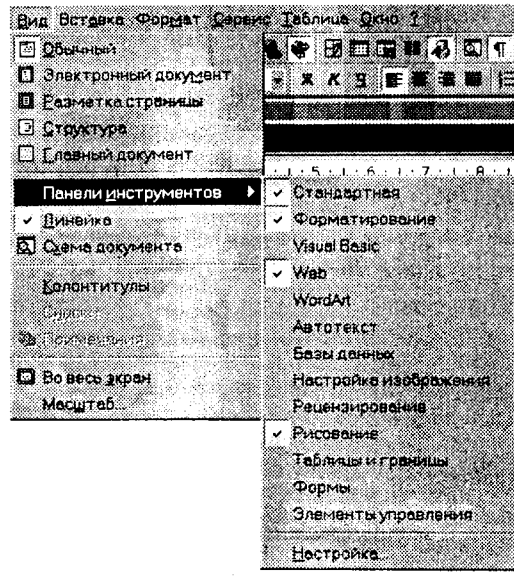


Рис. 5.7. Управление отображением панелей инструментов

Дадим краткую характеристику панелей инструментов:

- Стандартная — содержит элементы управления файловыми операциями, редактирования и экранного отображения;
- Форматирование — элементы управления форматированием документа;
- Visual Basic — предоставляет доступ к средствам создания и редактирования макросов и Web-сценариев;
- Web — элементы управления для навигации в Web-структурах данных;
- Web-компоненты — содержит команды для оформления Web-страниц и форм;
- Word-Art — это доступ к стандартному набору художественных элементов оформления документа;
- Автотекст — предоставляет быстрый доступ к средствам настройки различных функций автоматизированного изменения текста документа;
- База данных. Обеспечивает стандартное управление базами данных как внутренних таблиц Word, так и перенесенных Access;
- Буфер обмена. Команды работы с объектами буфера обмена;

- Настройка изображения. Команды, управляющие параметрами растровых изображений;
- Рамки. Команды для управления фреймами — прямоугольными областями документа;
- Рецензирование. Предоставление работы с различными версиями документа, внесение и удаление изменений без изменения исходного текста;
- Рисование. Команды, управляющие параметрами векторных изображений, выполняющие простые графические операции;
- Таблицы и границы. Создание и оформление страниц, а также дополнительные средства оперирования с данными таблицы;
- Формы. Команды, позволяющие подготовить документ к одному из видов публикации;
- Элементы управления. Набор готовых компонентов ActiveX для создания Web-страниц и форм.

Кроме того, что мы можем по желанию отображать на экране или скрывать любую панель инструментов, Word предоставляет возможность убирать или дополнительно отображать целые встроенные меню, отдельные команды меню, элементы управления панелей инструментов. Это можно проделывать как при помощи того же меню Вид ▸ Панели инструментов ▸ Настройка, но удобнее, пожалуй, пользоваться меню Сервис ▸ Настройка. При этом вызывается то же диалоговое окно (рис. 5.8), в котором вы можете точно так же выбирать отображение панелей инструментов, параметры его, а кроме того, выбрав в окне панель Команды, оперировать с меню, в том числе создавая новые пользовательские меню, содержащие команды по вашему усмотрению, а также отображая на экране и размещая удобным образом те

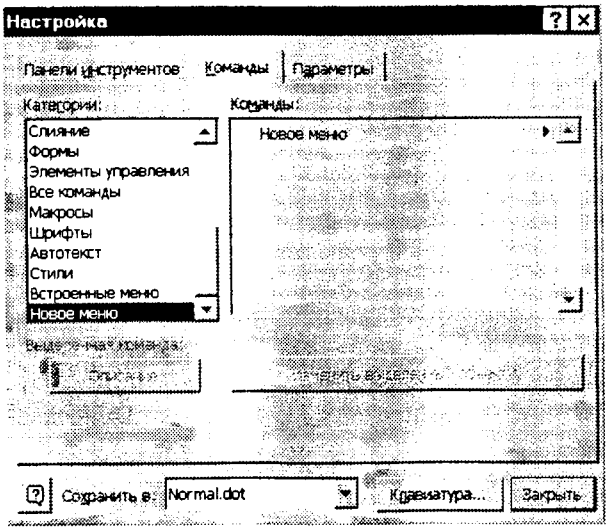


Рис. 5.8. Отображение элементов управления панелей инструментов и настройка меню

стандартные команды, которые часто используются и должны быть «под рукой». Выбрав на панели Команды в списке Категории ▸ Встроенные меню, вы получаете возможность продублировать либо, наоборот, удалить из строки меню любое встроенное меню.

Одно из удобств работы с процессором Word заключается в том, что многие команды можно вызвать различными способами: через меню, панели инструментов, «горячей» клавишей, из контекстного меню. Процессор имеет средства настройки, позволяющие привязать к любой команде различные способы активизации. Поэтому мы не будем рассматривать все возможные способы выполнения какой-либо операции, а возьмем за основу оперирование с командами меню, иногда дублируя их элементами управления панелей инструментов. Иначе это потребовало бы привлечения несоизмеримо большого объема информации, в том числе лишней, тем более что «горячие» клавиши, например, чаще всего претерпевают настройку под конкретного пользователя. Здесь лишь укажем заранее, какие клавиши и сочетания их, предусмотренные в Word по умолчанию, несут функциональную нагрузку или позволяют существенно ускорить работу по подготовке документа, так что менять их непрактично.

Клавиша Spase (Пробел) осуществляет вставку интервалов между словами. Необходимо помнить, что не стоит злоупотреблять этой функцией для того, чтобы перейти в нужную позицию в строке путем многократного нажатия на Пробел. Прежде всего потому, что большинство шрифтов, используемых при набивке текста, — пропорциональные. Это означает, что размеры символов каждого шрифта пропорциональны интервалам между ними. Поскольку это величины переменные, то при замене шрифта соответственно изменится ширина интервалов, и вместо нужной позиции вы окажетесь в несколько иной.

Клавиши Ctrl+Shift+Spase осуществляют так называемый неразрывный пробел. При этом слова, разделяемые неразрывным пробелом, визуально выглядят стоящими отдельно, но Word распознает их как единый комплекс, не разрывая его и не перенося автоматически отдельные слова комплекса на следующую строку. Это необходимо при вводе, например, длинных названий, состоящих из нескольких слов.

Клавиша Enter выполняет переход на начало нового абзаца. Это необходимо строго соблюдать, ни в коем случае не нажимая Enter в конце каждой строки. Word предусматривает автоматический переход на следующую строку, как только вы достигнете границы печатной области. Эти правила имеют принципиальное значение для последующего форматирования, поскольку в Word различаются стили абзаца и стили символа с соответствующими методами форматирования. Задаются параметры форматирования символов в меню Формат ▸ Шрифт. Параметры форматирования абзацев — в том же меню Формат ▸ Абзац. Word предоставляет широкие возможности выбора шрифтов, их комбинаций, оформления и анимации. Что касается форматирования абзацев — вы можете оперировать с предлагаемыми параметрами по своему усмотрению; единственное, на что обратим внимание прежде всего — на то, что лучше сразу установить Запрет висячих строк на панели окна Положение на странице, а также помнить, что если вы задаете Отступ первой строки, не нужно задавать Интервал перед (после) абзаца.

Клавиши Shift+Enter выполняют так называемый мягкий перевод строки. При этом строка прерывается до подхода к границе печатной области, но новый абзац не начинается. Этим пользуются, например, при набивке «шапок» деловых писем.

Клавиша Caps Lock позволяет вводить текст ПРОПИСНЫМИ буквами вместо строчных, и наоборот.

Клавиша Insert переключает два режима ввода и редактирования текста: *режим замены* и *режим вставки*. В режиме замены текстовые фрагменты заменяют редактируемый текст посимвольно, то есть каждый символ нового текста заменяет собой один символ прежнего. А в режиме вставки старый текст не меняется, а новый как бы вставляется в него, начиная с местонахождения курсора, раздвигая прежний текст. Как правило, более удобен режим вставки, за исключением тех случаев, когда работа ведется со стандартными формами, где необходимо редактировать единообразно куски содержимого.

Клавиши Delete и Backspase удаляют выделенные фрагменты текста, графические объекты, а также символы. Delete удаляет символы, расположенные справа от курсора ввода, а Backspase — соответственно, слева.

Клавиша Tab осуществляет переход к следующей позиции табуляции. Чаще всего ею пользуются при работе с таблицами, а также для форматирования документов, подготавливаемых как Web-документы.

Клавиша Home позволяет перейти в начало строки из любой позиции в середине ее, а клавиша End — в конец строки.

Клавиши Ctrl+Home и Ctrl+End обеспечивают переход, соответственно, в самое начало и самый конец документа из любой позиции в середине него.

Клавиши Ctrl+← и Ctrl+→ позволяют двигаться по тексту не через символ, как просто ← и →, а через слово, используя пробел для распознавания отдельных слов.

Помимо этого существуют различные другие встроенные сочетания клавиш, более или менее функциональные, а также возможность изменять заданные сочетания. Для этого пользуются тем же меню Сервис ▸ Настройка, где нажимают на кнопку Клавиатура и вызывают диалоговое окно, в котором каждой команде можно поставить в соответствие наиболее удобное для конкретного пользователя сочетание клавиш.

5.3.2. Создание документа

Под созданием документа мы будем понимать здесь так называемое создание нового документа, то есть научимся инициировать создание документа через меню Файл двумя способами: с применением или без применения шаблонов.

Создание нового документа без использования шаблона

Запустив MS Word и произведя в случае необходимости соответствующие настройки, можно приступать к созданию нового документа. Для этого нужно, например, щелкнуть мышью на крайнем левом значке панели инструментов Стандартная, при этом на экране появится пустой документ, которому по умолчанию будет дано имя ДокументN, где N — порядковый номер данного документа. В этом случае к нему будет применен стандартный шаблон NORMAL.DOT.

Более корректным методом является создание нового документа через меню Файл ▸ Создать. В этом случае откроется диалоговое окно, содержащее стандартно несколько панелей, из которых активной изначально является панель Общие (рис. 5.9). В ней отображается хотя бы один значок типа документа — Новый документ. Если вы хотите создать обычный документ, то просто нажмите OK, и новый документ будет открыт с шаблоном NORMAL.DOT.

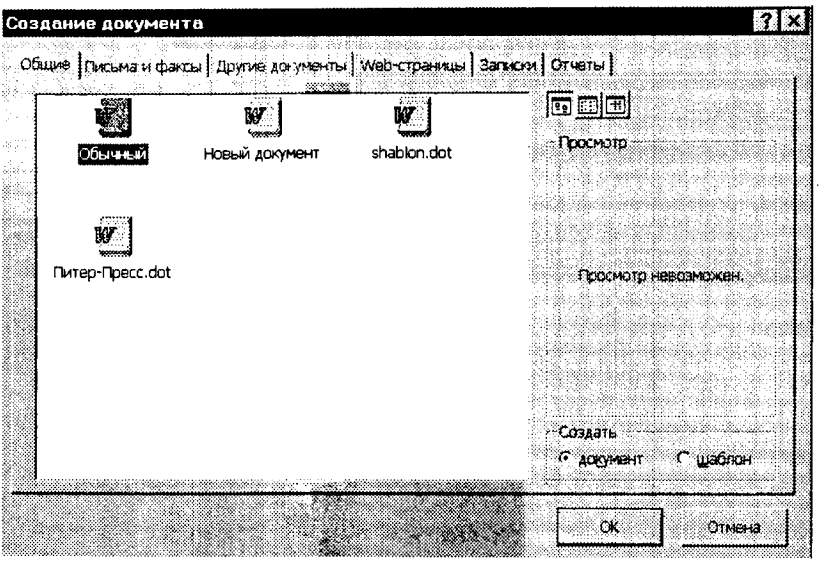


Рис. 5.9. Создание нового документа на основе шаблона Normal.dot

Создание документа на основе существующего шаблона

Второй способ — это создание нового документа на основе одного из существующих шаблонов. Это достигается при помощи того же меню Файл ▸ Создать, но в данном случае мы выбираем другую панель открывшегося диалогового окна. Разные панели соответствуют разным типам документов (рис. 5.10).

Отметим, что создание документа на основе шаблона:

- во-первых, проще для начинающего пользователя, поскольку в шаблоне заложены стандартные для данного типа документа методы форматирования текста, страниц, графические элементы;
- во-вторых, удобнее в случае стандартных документов, которые могут иметь определенный вид, и чем разрабатывать документ «с нуля», удобнее воспользоваться готовым стандартом;
- в-третьих, необходимо при создании документа «под заказ», когда заказчик обязывает пользоваться конкретным шаблоном, который зачастую им же или для него и разработан. Чаще всего это происходит в случае последующей публикации в печатном виде и декларируется издательством.

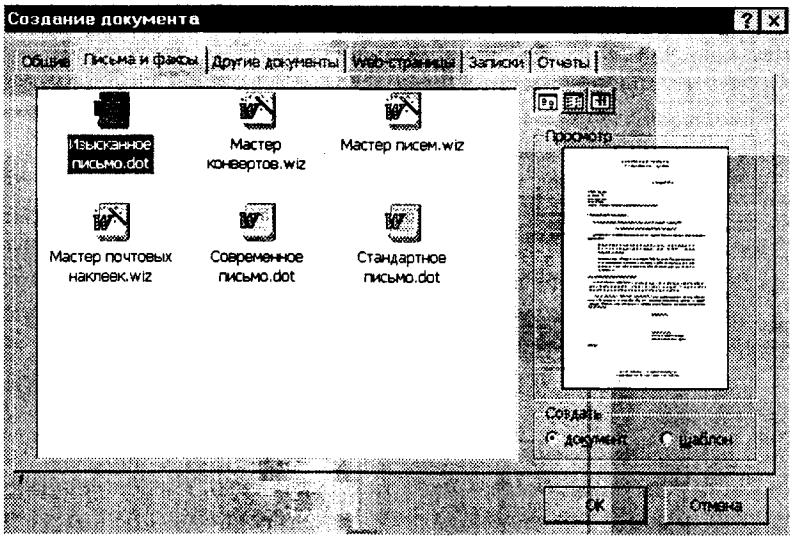


Рис. 5.10. Создание документа на основе шаблона

Для того чтобы создать документ на основе одного из предлагаемых шаблонов, нужно, прежде всего, выбрать наиболее подходящую панель диалогового окна, а затем, щелкнув мышью на одном из значков, соответствующих разным документам одного типа, просмотреть вид документа, который будет создан в случае применения данного шаблона. Просмотр осуществляется в окне справа. Если вид будущего документа устраивает, значит, шаблон подходит, и остается щелкнуть на кнопке OK. При этом в рабочей области появится соответствующий стандартный бланк, содержащий отформатированный текст и пустые поля для ввода авторского текста. Для завершения создания документа надо ввести свой текст в эти поля.

Если вид будущего документа вас почему-либо не устраивает, то необходимо рассматривать следующий шаблон и т. д. Word предлагает самые разнообразные шаблоны, так что выбор широк. Тем не менее, может сложиться такая ситуация, что пользователя не удовлетворяет ни один из них. В этом случае Word предоставляет возможность создавать собственные шаблоны как «с нуля», так и на основе существующих. Для этого в правом нижнем углу соответствующей панели диалогового окна надо щелкнуть переключатель Шаблон, а не Документ.

Режимы отображения

Приступая к работе над текстом документа, необходимо выбрать один из режимов работы, наиболее подходящий для данного случая. Word предлагает четыре режима отображения документов. Переключение в тот или иной режим осуществляется в меню Вид либо четырьмя кнопками в левой части горизонтальной полосы прокрутки.

Режимы отображения документов:

Normal (*Обычный режим*). Он представляет собой действительно обычный режим работы с текстовыми документами, в котором представлена только содержатель-

ная часть документа со всеми элементами форматирования шрифта и абзацев, но не страниц. Постраничная организация представлена упрощенно, что, как минимум, ухудшает визуальное восприятие содержимого документа. Зато гораздо быстрее происходят ввод текста и иные операции с большими документами на малопроизводительных компьютерах, Word реже «зависает».

Web Layout (*Web-документ*). Этот режим удобен при оформлении документов, предназначенных для публикации в Интернете. Текст представляется развернутым на всю страницу. Этот режим совершенно не предоставляет возможности увидеть документ в том виде, в котором будет представлена его твердая копия, но учитывает особенности форматирования электронных документов, такие, как резко ограниченные возможности форматирования, непредсказуемость его внешнего вида при открытии на другой платформе (здесь рекомендуем вспомнить про формат PDF).

Print Layout (*Печатный документ*). Исторически сложилось так, что в нашей литературе этот режим чаще называют *режимом разметки*. Существенной разницы нет, поскольку в обоих названиях заложена основная суть этого режима: в нем максимально реализован упоминавшийся ранее принцип WYSIWYG. При работе над текстом документа мы можем адекватно представлять себе, как те или иные исправления будут выглядеть при печати. Поэтому, как правило, пользователи предпочитают заниматься оформлением документа именно в этом режиме. Исключение составляет, как уже говорилось, обработка больших документов на маломощной технике.

Outline (*Структура*). В этом режиме документ отображается с дополнительным выделением его структуры: абзацев, заголовков разных уровней, списков. Он очень удобен при работе с большими и имеющими сложную структуру документами. Безусловно, набивать текст в нем не стоит, но можно заниматься окончательным редактированием структуры: перемещать разделы и главы, подчинять их другому уровню. Можно выделить так называемый главный документ и работать с ним, не открывая вложенных.

5.3.3. Ввод текста и редактирование

После того как мы создали пустой документ, можно приступать к вводу текста. Если документ был создан на основе шаблона, то специальных средств ввода и форматирования текста, как правило, не требуется либо они специфические для каждого шаблона, поэтому будем рассматривать ввод и редактирование применительно к документу, создаваемому «с нуля».

Задание параметров страницы

Прежде чем приступать непосредственно к вводу текста, полезно заранее установить некоторые удобные для данного документа параметры форматирования.

Например, удобно сразу задать параметры страниц. Для этого выберите в меню Файл ▸ Параметры страницы. Откроется соответствующее диалоговое окно (рис. 5.11), содержащее четыре вкладки.

Выбрав вкладку Поля, вы можете назначить границы полей страницы и, соответственно, печатной области, в пределах которой будет помещаться текст.

Вкладка Размер бумаги позволяет задать страницам документа формат, отличный от стандартного А4, а также переопределить ориентацию с книжной на альбомную.

Вкладка Источник бумаги отвечает за правила подачи бумаги при печати.

При помощи вкладки Макет можно задавать различные специальные режимы просмотра параметров форматирования страниц создаваемого документа.

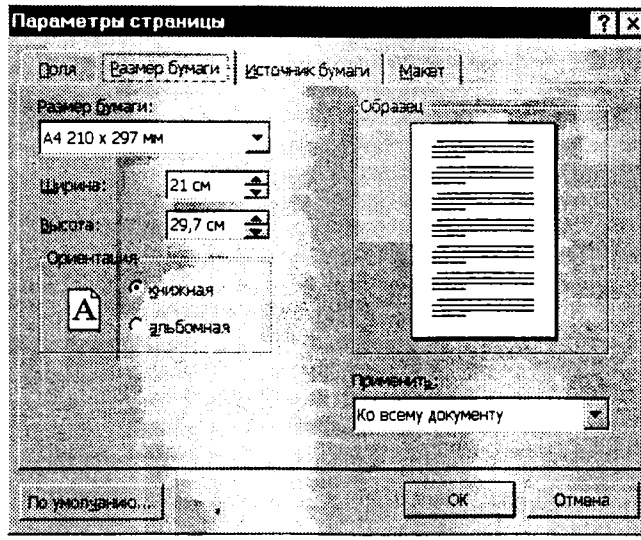


Рис. 5.11. Задание параметров страницы

Другие параметры документа

Можно заранее установить и другие параметры работы с документом, выбрав в меню Сервис ▸ Параметры. Тем самым вызывается диалоговое окно (рис. 5.12), содержащее десять вкладок. Перечислять их все смысла не имеет, поскольку пользователь сам при ближайшем ознакомлении выберет то, что его интересует.

Единственное, на что сразу обратим внимание, — это вкладка Сохранение, поскольку в ней находятся важные для обеспечения сохранности документа и удобства работы с ним функции. Особый практический интерес представляют:

- Разрешить быстрое сохранение — его лучше не разрешать и отключить, так как оно может нести довольно неприятные и неожиданные неудобства;
- Автосохранение каждые — его, наоборот, лучше включить, задав надлежащую периодичность сохранения. Эта функция спорна и часто вызывает нарекания по тому поводу, что она элементарно не срабатывает. Например, при «зависании» Word почти наверняка все сохраненные автоматически данные будут утеряны. Поэтому лучше по ходу работы с документом периодически «вручную» сохранять его через Файл ▸ Сохранить. Тем не менее, включение этой функции не повредит.

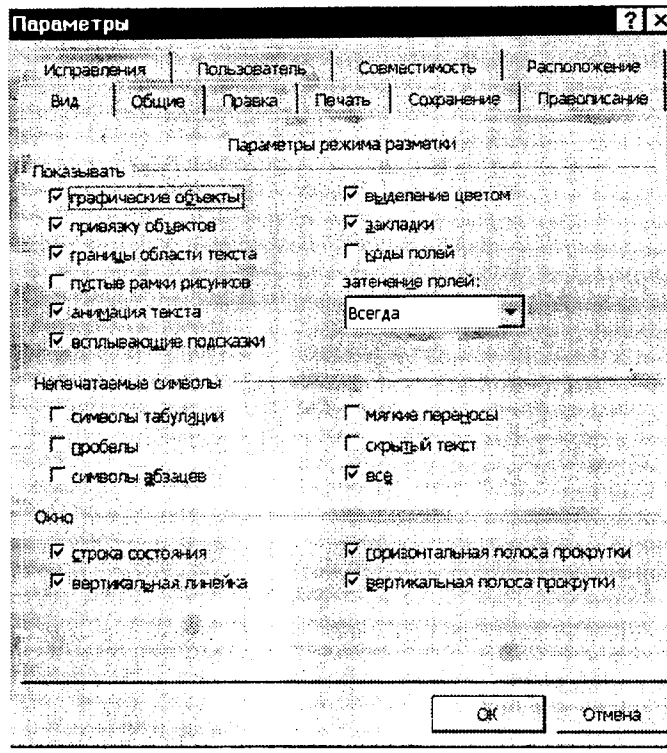


Рис. 5.12. Диалоговое окно настройки параметров документа

Непосредственно ввод текста с клавиатуры особых сложностей не вызывает, надо лишь быть предельно внимательным и набивать символы в правильном порядке. Каждый символ в Word имеет свой ASCII-код, которым он представлен в электронных документах, поэтому любая ошибка ввода адекватно воспринимается и фиксируется. Word имеет замечательное свойство: он помнит все производимые с документом действия и предоставляет возможность отказа от них, а также последующего возврата к ним. Отмена действий осуществляется при помощи меню Правка ▸ Отменить «указание совершенного действия» либо синей полукруглой стрелкой влево на панели инструментов Стандартная. Возврат к ним производится из того же меню Правка ▸ Повторить «указание совершенного действия» либо синей полукруглой стрелкой вправо на панели инструментов Стандартная.

Большой интерес для пользователя представляют специальные средства ввода и редактирования, и не только текста, но и различных графических объектов, составляющих неотъемлемую часть оформления сложной документации. В широком смысле, весь Word представляет собой единое мощное средство автоматизации подготовки документов. Но он содержит в себе определенные специальные средства, позволяющие практически полностью автоматизировать вставку объектов различной природы. Остановимся на некоторых из них.

Автозамена, автотекст, автоформат

Мы намеренно объединили эти три функции, потому что вызов каждой из них осуществляется в одном и том же диалоговом окне. Word, как уже указывалось ранее, предусматривает дублирование вызова многих команд. Так, например, данное диалоговое окно можно вызвать, как минимум, тремя разными способами:

- через меню Вставка ▸ Автотекст ▸ Автотекст;
- через меню Формат ▸ Автоформат ▸ Параметры;
- через меню Сервис ▸ Автозамена.

Разница лишь в том, какая панель этого окна будет активной по умолчанию.

- Функция Автозамена позволяет существенно сократить время для ввода текста и объем набивки. Этой функцией удобно пользоваться при замене ввода часто повторяющихся длинных последовательностей символов (и не только букв) на ввод коротких комбинаций. Word 2000 содержит довольно большой встроенный словарь автозамены в рамках проверки правописания и замены случайных опечаток. Словарь подлежит корректировке и добавлению новых комбинаций автозамены. Для этого необходимо на соответствующей панели диалогового окна ввести подлежащие замене и заменяющие их сочетания символов, а затем выбрать опцию Добавить. Можно выделить в самом документе заменяемый текст, а затем активировать диалоговое окно, набрать заменяющую комбинацию вручную и щелкнуть Добавить.
- Автотекст дает возможность автоматически вводить заложенные в его словаре блоки текста. При использовании функции автотекста пользователь может выбрать из предлагаемого ему словаря и вставить в документ любой текстовый фрагмент. Также Word предлагает всплывающей подсказкой добавить какой-либо фрагмент, если пользователь введет первые четыре его символа. Можно добавлять новые текстовые блоки в словарь, либо набивая их непосредственно в диалоговом окне и выбирая опцию Добавить, либо аналогичным образом, но предварительно выделив подлежащий добавлению текст, и лишь затем, вызвав диалоговое окно, щелкнуть Добавить.
- Автоформат, при соответствующих настройках, подразумевает автоматическое форматирование вводимого текста. Он также позволяет автоматически отформатировать весь документ по своему усмотрению, если задать в меню Формат ▸ Автоформат команду Сразу весь документ, с указанием типа документа.

Вставка специальных символов

Под специальными символами в Word подразумеваются различные символы, не предусмотренные для ввода с клавиатуры по той причине, что пользуются ими очень редко. Эти символы имеют так называемые расширенные ASCII-коды. Поскольку ими пользуются редко, не имеет смысла закладывать их коды в память. Иногда все же возникает необходимость оперирования с такими символами при редактировании специальных текстов. Для этого в Word предусмотрена

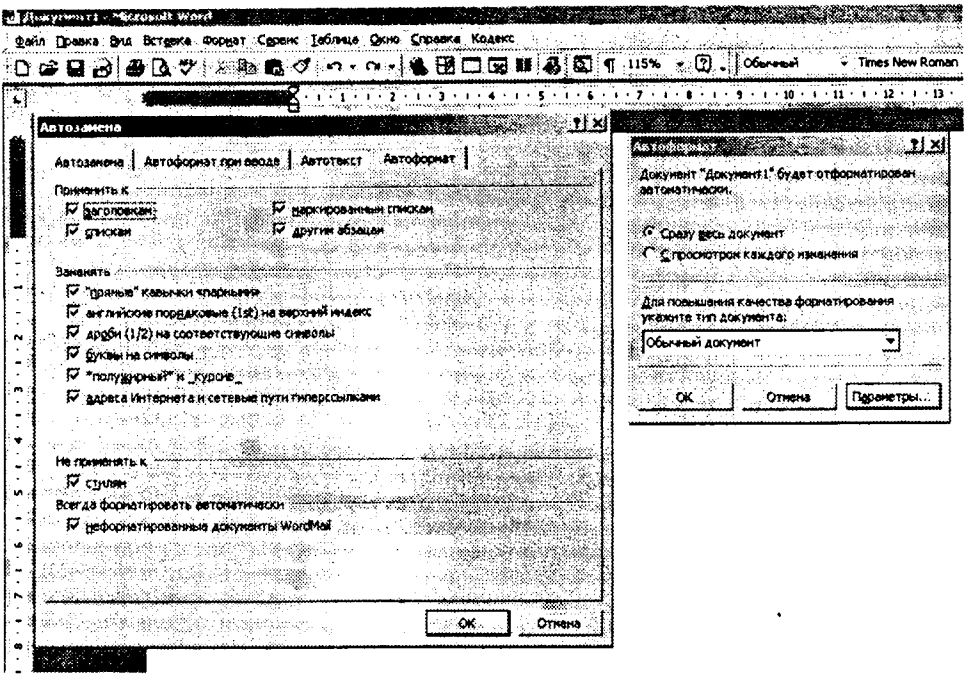


Рис. 5.13. Диалоговое окно Автозамена, вызванное через меню Формат

команда меню Вставка ▸ Символ. При этом появляется диалоговое окно, содержащее две панели: Символы и Специальные символы. Они существенно отличаются друг от друга тем, что панель Специальные символы содержит небольшой список редко употребляемых символов с соответствующими им предусмотренными комбинациями клавиш. Панель Символы предлагает многообразие символов, соответствующих определенному Шрифту и, более того, в некоторых шрифтах, еще и Набору, соответствующему определенному специфическому алфавиту, или языку. Обе панели предусматривают возможность организации «горячих» клавиш, соответствующих любому символу, в том случае, если вы пользуетесь этим символом достаточно часто. Для этого в диалоговом окне надо щелкнуть на кнопке Клавиша и задать подходящую комбинацию клавиш.

Стили

Наиболее мощным средством автоматизации ввода и редактирования в Word являются стили. Существует два подхода к форматированию (оформлению) документа. Первый, исторически самый старый, состоит в том, что редактор, форматируя текст, задает для каждого элемента параметры шрифтов, абзацев и т. п. Такой подход называется *физическим форматированием*, так как редактор практически задает физические параметры представления текста: параметры шрифта, размеры на бумаге. Но физическое форматирование имеет много недостатков:

- редактор вынужден вручную следить за единством стилевого оформления документа. Например, выбирать для всех списков один и тот же шрифт;

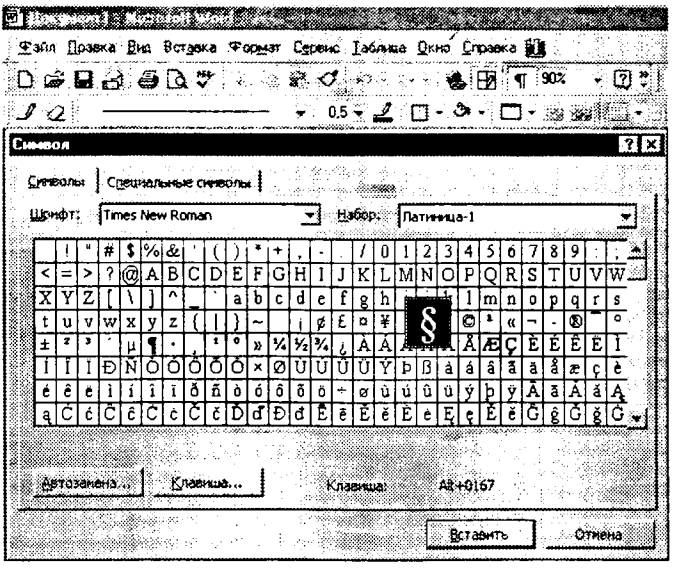


Рис. 5.14. Вставка специальных символов

- затруднено изменение документа. Например, если нужно поменять шрифт заголовков, то придется пройти по всему документу, внося изменения;
- текстовый процессор, имея только информацию о физическом форматировании, не имеет информации о структуре документа и не может предоставить таких дополнительных услуг, как, например, автоматическое составление оглавления.

Поэтому в настоящее время любое средство подготовки документов поддерживает второй подход к форматированию: *логическое форматирование*. В MS Word для поддержки такого форматирования и служит механизм стилей. При использовании стилей оператор выделяет в документе логические блоки: заглавия, абзацы, перечисления и др., и для каждого создает стиль — описание физических параметров, которые используются для отображения этого логического элемента. Редактируя текст, оператор для каждого логического элемента документа указывает, какому они принадлежат стилю.

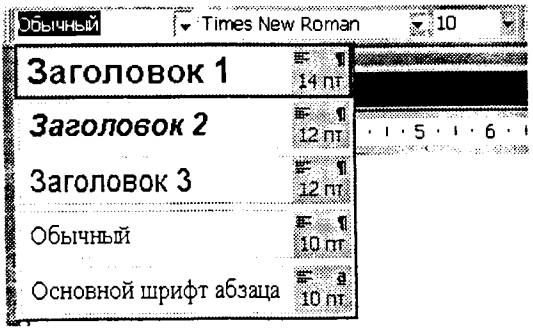


Рис. 5.15. Задание стиля

Например, набивая заголовок самого верхнего уровня, нужно указать стиль Заголовок 1. Word, распознав этот стиль, автоматически поместит этот элемент в оглавление. Если вы захотите, чтобы все заголовки отображались прописными буквами, вам достаточно будет только поменять параметры стиля. Если этот документ будет преобразован в Web-страницу, то конвертор из информации о стиле сгенерирует правильные HTML-теги. Если бы вы просто указали для заголовка больший размер шрифта и выравнивание посередине, то во всех перечисленных случаях вам пришлось бы проделывать все это вручную, долго и не всегда оптимальным образом.

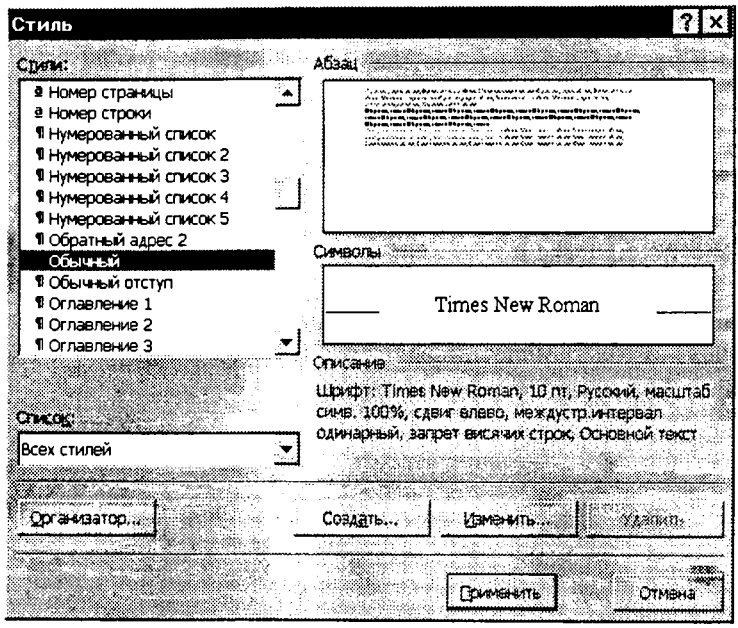


Рис. 5.16. Окно управления стилями

Поэтому при правильной работе с MS Word нужно по возможности избегать физического форматирования и стараться использовать механизм стилей для логического форматирования. Частным случаем реализации в Word функций логического форматирования являются ранее упомянутые шаблоны.

5.3.4. Сохранение документа

Необходимости аккуратного и правильного сохранения документа было уделено много внимания ранее, поэтому здесь остановимся на общих ошибках, чаще всего допускаемых при первом опыте работы с Word. Самая распространенная ошибка — сохранение не в том каталоге или не в той папке, в которой требуется. Для того чтобы избежать этого, надо при появлении диалогового окна, вызываемого из меню Файл ▸ Сохранить как, обратить внимание, куда предлагает сохранить ваш доку-

мент Word. По умолчанию он предлагает сохранить его в Мои документы. Если это вас устраивает, то нажмите Сохранить, а если нет, то прежде перейдите в нужный каталог.

Другая распространенная ошибка — случайное сохранение под тем именем, которое предлагается по умолчанию, не запомнив его. Word предлагает для сохранения имя, состоящее из первой фразы текста. Чтобы этого не произошло, надо в том же диалоговом окне задать то имя, под которым вы будете потом оперировать с документом, потому что в противном случае его можно вообще не найти через Поиск.

Наконец, та ошибка, о которой уже говорилось ранее, — сохранение не в том формате. В Word это не так часто случается, поскольку по умолчанию предлагается формат DOC.

5.3.5. Управление настройками MS Word

Рассмотренные в настоящем разделе инструменты позволяют пользователю создать стройную, целостную и непосредственно ориентированную на решение его профессиональных задач среду работы. Очевидно, что разработка такой среды требует (помимо определенной квалификации) еще и немалых трудозатрат. Цели их уменьшения в Word служит *организатор настроек*. Он может быть вызван с помощью меню Сервис ▸ Шаблоны и надстройки..., далее в появившемся диалоговом окне нужно нажать кнопку Организатор.

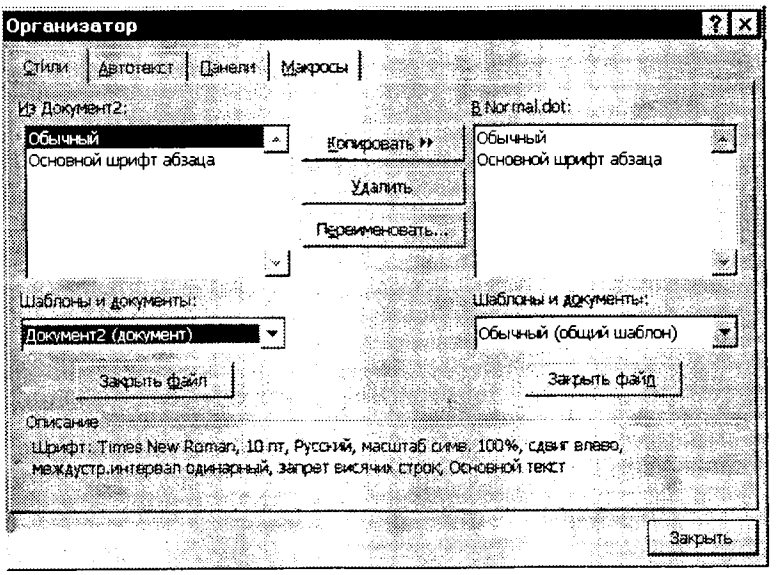


Рис. 5.17. Окно организатора настроек

В частности, окно, показанное на рис. 5.17, демонстрирует технику экспорта настроек стилей между документами.

5.4. Решение задач оформления экономической документации средствами MS Word

Остановимся более подробно на специфике применения текстового процессора Word, проявляющейся при работе с документами экономического содержания. Безусловно, многие из задач, возникающих при оформлении экономической документации, являются общими по существу для документов любого типа независимо от их отраслевого «происхождения». Основная специфика, как правило, заключается в том, что при оформлении экономических документов необходимо уделять повышенное внимание как форме, так и содержанию табличных, графических и формульных данных. Также немаловажно аккуратно соблюдать систему стилей. Последнее предъявляет дополнительные требования к уровню владения пользователем соответствующими инструментами. Рассмотрим несколько практических примеров.

5.4.1. Ввод формул

Даже в относительно несложных экономико-математических текстах невозможно обойтись без формул, содержащих знаки суммирования, интегрирования, надстрочные и подстрочные индексы нескольких уровней. Формулы в Word представляют собой OLE-объекты, погружаемые в основной текст. При обращении пользователя к формуле (допустим, по щелчку мышью) вызывается специальная программа — Редактор формул (Microsoft Equation). На рис. 5.18 показана панель инструментов, активизирующаяся при вызове Microsoft Equation.

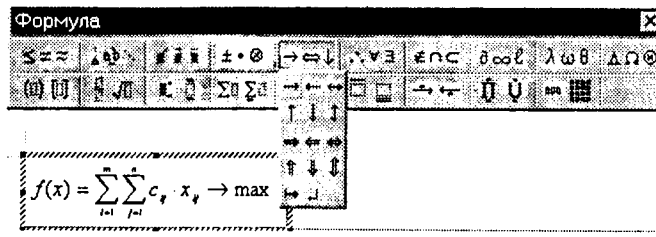


Рис. 5.18. Пример редактирования формулы средствами Microsoft Equation

Исходная вставка формулы в текст происходит через меню Вставка ▸ Объект, далее в появившемся диалоговом окне Вставка Объект выбирается объект Microsoft Equation[1].

СОВЕТ

Если создаваемый текст содержит много формул, то имеет смысл с помощью средств настройки, описанных в 5.3.1, вынести пиктограмму вставки формулы на панель инструментов.

[1] Предполагается, чтобы Microsoft Equation был установлен в процессе установки Word.

ПРИМЕЧАНИЕ

Вообще говоря, Microsoft Equation является не единственным средством работы с формулами из среды Word. В качестве альтернативы ему обычно предлагается редактор формул MathType, с которым также можно работать по OLE-технологии.

5.4.2. Расположение и автонумерация формул в тексте

Читателю, которому хоть раз приходилось сталкиваться с книгами математического содержания, знакомы общие принципы организации изложения в них. Как правило, значимые формулы располагаются отдельно на строке, в ее центре. При этом в правой части строки в круглых скобках располагается порядковый номер формулы, который в дальнейшем используется для ссылок на нее. Пример такого текста показан на рис. 5.19.

Известно, что

$$(\sin(x))' = \cos(x), \qquad (1)$$

а

$$(\cos(x))' = -\sin(x). \qquad (2)$$

На основании (1) и (2) может получить

$$(\sin(x)\cdot\cos(x))' = (\sin(x))'\cdot\cos(x) + \sin(x)\cdot(\cos(x))' = \cos^2(x) - \sin^2(x). \quad (3)$$

Рис. 5.19. Пример текста, содержащего формулы

Перечислим задачи, которые могут возникнуть при оформлении текстового фрагмента.

- *во-первых*, нужно добиться такого форматирования строки, содержащей формулу, при котором сама формула выравнивается по центру, а ее номер — по правому краю (очевидно, что простым центрированием абзаца такого эффекта добиться не удастся);
- *во-вторых*, желательно обеспечить автонумерацию формул. Это означает, что если мы захотим вставить между существующими строками еще одну с какой-то новой формулой, например, напоминающей читателю, как считается производная от произведения функций, то должен произойти автоматический сдвиг номеров, начиная от вставленной строки;
- *в-третьих*, следует «научить» систему самостоятельно менять текст ссылок на формулы при автоматическом изменении их номеров.

Теперь рассмотрим подробнее те способы, которые предлагает нам текстовый процессор MS Word для решения данных задач.

Форматирование строки с формулой

Для задания различных видов выравнивания для различных частей строки можно применять задание пользовательской (настраиваемой) табуляции (команда меню Формат ▸ Табуляция...).

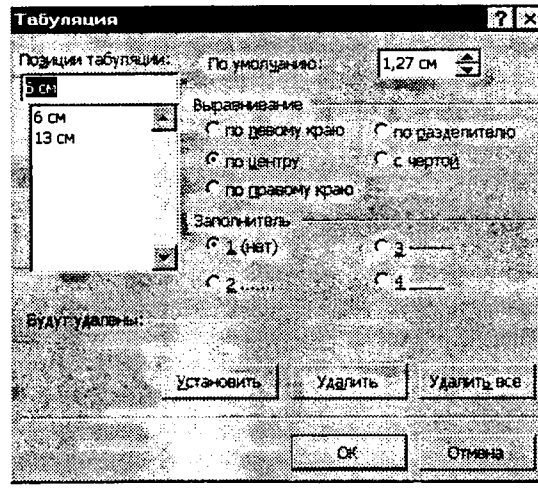


Рис. 5.20. Определение позиций табуляции

В диалоговом окне, показанном на рис. 5.20, задаются позиции табуляции и соответствующие им способы выравнивания. После настройки табуляции пользователь получает возможность с помощью клавиши Tab перемещаться (внутри абзаца) к заданным позициям и вводить относительно них те или иные элементы текста (формулы, их номера и т. д.). Заметим, что позиции и виды выравнивания для табуляции могут задаваться непосредственно щелчком мыши по *линейке выравнивания* (рис. 5.21).



Рис. 5.21. Определение позиций табуляции с помощью линейки выравнивания

СОВЕТ

Заметим, что в свете того, чтобы было сказано о стилях, описанные настройки табуляции имеет смысл оформить как еще один стиль и применять его ко всем подобным абзацам, содержащим отдельно вынесенные формулы.

Задание автонумерации формул

При описании метода решения данной задачи мы коснемся такого мощного инструмента текстового процессора Word, как *поля*[1]. Сразу оговоримся: глубокое и обстоятельное изучение этого средства в рамках данной книги невозможно, поэтому

[1] Термин «поле» является многозначным. В дальнейшем он встретится нам при рассмотрении электронных таблиц и баз данных, но будет использоваться в качественно ином смысле.

заинтересованному читателю следует обратиться непосредственно к подробным практическим руководствам по работе с MS Word. Отметим лишь, что поля в Word представляют собой элементы, которые внедряются в основной текст и могут автоматически изменяться системой по тем или иным правилам. Например, с помощью полей DATE и TIME в текст можно вставить текущие дату и время.

Специальное поле SEQ позволяет вести в документе списки автонумерации. Чтобы его вставить в текст, нужно воспользоваться функцией меню Вставка ▸ Поле....

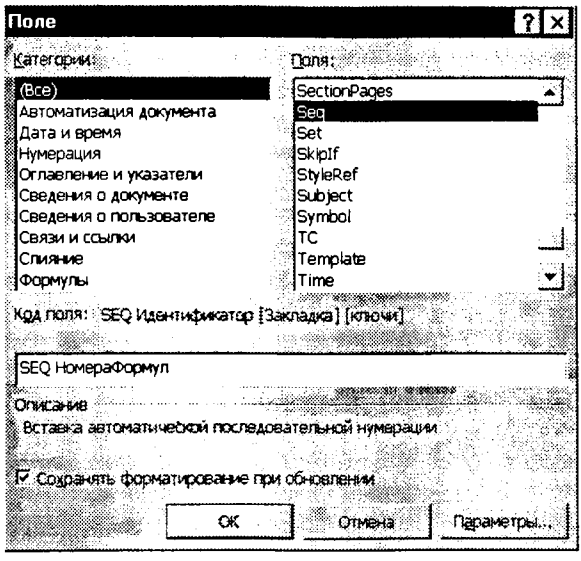


Рис. 5.22. Вставка поля SEQ

Окно диалога, показанное на рис. 5.22, служит для выбора типа вставляемого поля. За кодом поля следуют управляющие параметры. В случае поля SEQ необходимо также задать Идентификатор списка автонумерации: поскольку в документе могут вестись параллельно несколько таких списков, то Идентификатор нужен для того, чтобы отличать их друг от друга. В нашем случае списку дано имя НомераФормул.

Очевидно, что поле SEQ должно быть вставлено в то место текста, где мы хотим видеть автоматически меняющийся номер. Таким образом, в рассматриваемом нами примере каждая строка с формулой внутри скобок, расположенных справа, должна содержать это поле.

При возникновении события, ведущего к изменению последовательности номеров (например, при вставке новой нумерованной строки между существующими), их автоматического перевывода **не происходит!** Для того чтобы его добиться, нужно выделить фрагмент текста, содержащий поля, и нажать кнопку F9.

Автоматически изменяющиеся ссылки на номера формул

Очевидно, что номера формул нужны не сами по себе, а для того, чтобы на них можно было ссылаться в тексте, что, в свою очередь, означает необходимость

обеспечить при работе с документом автоматическое преобразование не только самих номеров, но и ссылок на них. Для того чтобы это происходило, следует:

- оформить номера формул как специальные элементы, на которые можно сослаться. К таковым в Word относятся *закладки*;
- ссылки на номера формул должны быть реализованы с помощью специальных элементов Word, называемых *перекрестными ссылками*.

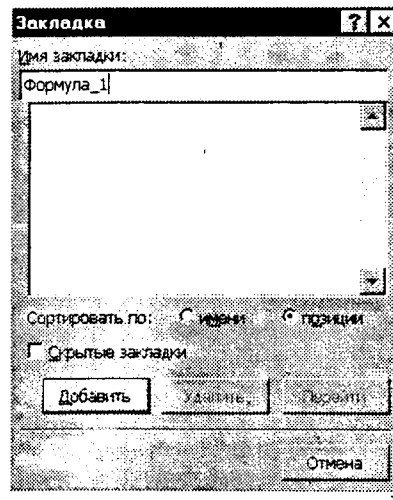


Рис. 5.23. Создание закладки

В результате процесс решения рассматриваемой задачи распадается на два принципиальных этапа:

1. Создание закладки, указывающей на номер формулы:
   - выделить номер формулы;
   - выполнить команду меню Вставка ▸ Закладка...;
   - в окне диалога, показанном на рис. 5.23, задать имя создаваемой закладки;
   - нажать кнопку Вставить.
2. Определение перекрестной ссылки, указывающей на закладку:
   - установить курсор на то место, где должна располагаться ссылка на формулу;
   - выполнить команду меню Вставка ▸ Перекрестная ссылка...;
   - в окне диалога (рис. 5.24) установить тип объекта, на который задается ссылка, — **закладка**, указать, что ссылка происходит на **текст закладки**, и, наконец, выбрать нужное имя закладки в списке Для какой закладки;
   - нажать кнопку Вставить.

Для перевывода значений перекрестных ссылок их так же, как и формулы, следует выделить и нажать кнопку F9.

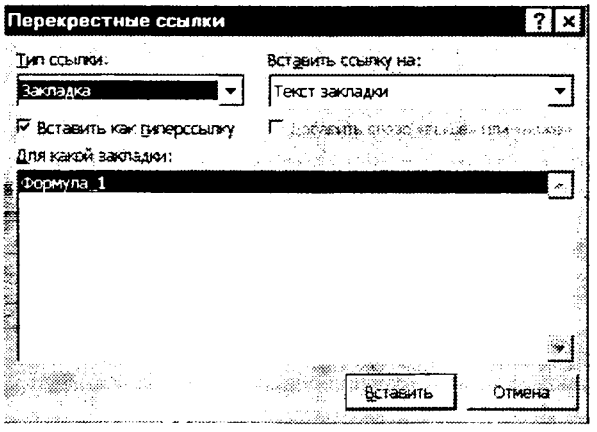


Рис. 5.24. Определение перекрестной ссылки на закладку

ПРИМЕЧАНИЕ

Как нетрудно догадаться, описанный механизм задания автонумерации и ссылок может быть использован не только при работе с формулами, но и с любыми другими объектами, которые требуют учета внутри текста.

5.4.3. Оформление финансово-экономической документации

В завершение разговора об экономических приложениях текстового процессора MS Word рассмотрим пример комплексного применения функциональных возможностей данной программы при создании документа экономического характера.

Рассмотрим некоторый условный пример. Допустим, перед нами стоит задача разработки документации **по инвестиционному проекту завершения строительства станции метро «Адмиралтейская»**.

Очевидно, что содержательная сторона данного вопроса, потенциально оказывающего влияние на социально-экономическую ситуацию в большом городе, несопоставима по своей сложности и важности с задачами «технического» характера. Однако не вызывает сомнений и тот факт, что высокое качество оформления документа значительно ускоряет и повышает уровень работы с ним, позволяет принимать на его основе более обоснованные и продуманные решения.

Процесс оформления документации по данному инвестиционному проекту распадается на ряд этапов (подзадач).

Структуризация документа

В содержательном плане текст описания инвестиционного проекта распадается на ряд частей. Для их корректного выделения удобно использовать механизм стилей. Применим для общего названия инвестиционного проекта стиль Заголовок 1. Далее подразделы будем выделять стилями Заголовок N, соблюдая принцип вложен-

ности, или иерархической организации структуры документа: чем ниже уровень раздела, тем больше N — номер его оглавления. А ключевые моменты, на которых **необходимо** акцентировать внимание, — специальными стилями, имеющими физическое форматирование «**жирный**» или «*курсив*». Сам основной текст будем набирать стилем Обычный. К соблюдению стилей советуем отнестись особенно внимательно, поскольку это понадобится при окончательном форматировании документа, и не только для автоматической сборки оглавления, как в нашем простейшем случае.

Необходимо отметить, что выделение с помощью стилей названий частей (разделов) документа является формой его структурирования, позволяющей в дальнейшем эффективно работать с его отдельными частями. На рис. 5.25 представлен создаваемый документ в режиме Схема документа. Данный режим вызывается командой меню Вид ▸ Схема документа либо щелчком мыши по пиктограмме.

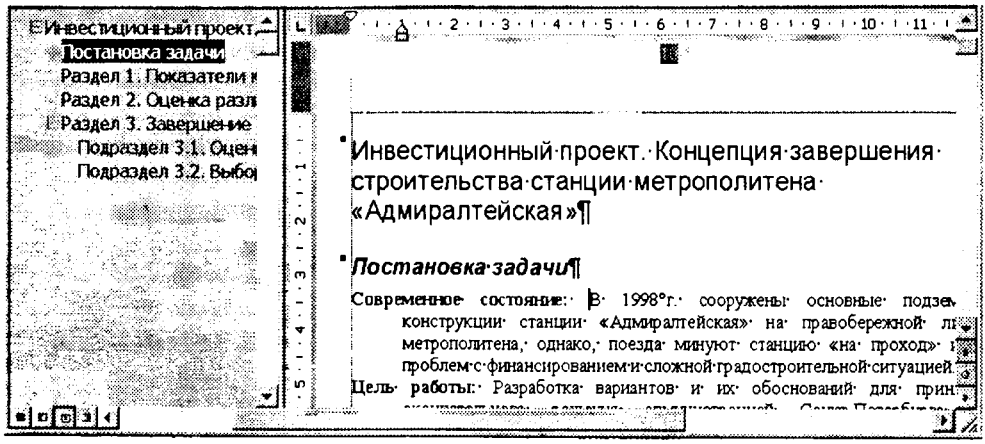


Рис. 5.25. Работа в режиме Схема документа

Как видно из рис. 5.25, стандартное окно Word в режиме Схема документа разбивается на две части. Левая часть содержит «дерево» со структурой документа, причем уровень объекта в данном дереве определяется тем, каким стилем Заголовок N он был выделен. Соответственно, при выборе того или иного пункта в левой части окна в правой происходит автоматическое позиционирование к тому месту, где начинается текст выбранного пункта.

Оформление табличных данных

Целью работ по рассматриваемому проекту является разработка для администрации Санкт-Петербурга вариантов принятия окончательного решения по размещению вестибюля и наклонного хода построенной подземной станции «Адмиралтейская» на правобережной линии метрополитена (с необходимым экономическим обоснованием).

Для реализации этой цели были разработаны наиболее важные показатели, отражающие значимость всех вариантов, и весовые коэффициенты каждого из них, или баллы. Все они, для удобства восприятия, сведены в единую табл. 5.4.

Таблица 5.4. Показатели и баллы оценки вариантов размещения вестибюльного комплекса

№ пп.	Критерии	Показатели критериев	Баллы
	Социальные критерии		
1.	Безопасность пешеходных подходов	Высокая	10
		Средняя	5
2.	Степень удобства входа пассажиров на станцию (время входа, лестницы, количество эскалаторов, протяженность пешеходных коридоров подземной станции)	Высокая	10
		Средняя	5
		Низкая	3
3.	Затраты времени на пересадку между метро и наземным транспортом	Нормативные	5
		Выше нормативных	1
4.	Влияние процесса строительства на условия проживания, инженерные сети, уличное движение	Не влияет	5
		Влияет	3
		Существенно влияет	1
	Градостроительные критерии		
1.	Объемы сноса зданий	Отсутствуют	5
		Незначительные	3
		Значительные	1
2.	Объемы расселения жителей	Отсутствуют	5
		Незначительные	4
		Средние	3
		Значительные	1
3.	Объемы освобождения нежилого фонда	Незначительные	3
		Средние	2
		Большие	1
	Нормативные критерии		
1.	Обеспечение нормативных требований эксплуатации пешеходных путей сообщения пассажиров	Обеспечены	5
		Имеются сложности	3
	Экономические критерии		
1.	Стоимость строительства	Минимальная из всех вариантов	20
		Средняя	10
		Максимальная	3
2.	Продолжительность строительства	Минимальная	10
		Средняя	5
		Максимальная	3

Операции по созданию и оформлению таблиц выполняются из меню Таблица либо из панели инструментов Таблицы и границы. Результат вставки табличной информации в разрабатываемый документ показан на рис. 5.26.

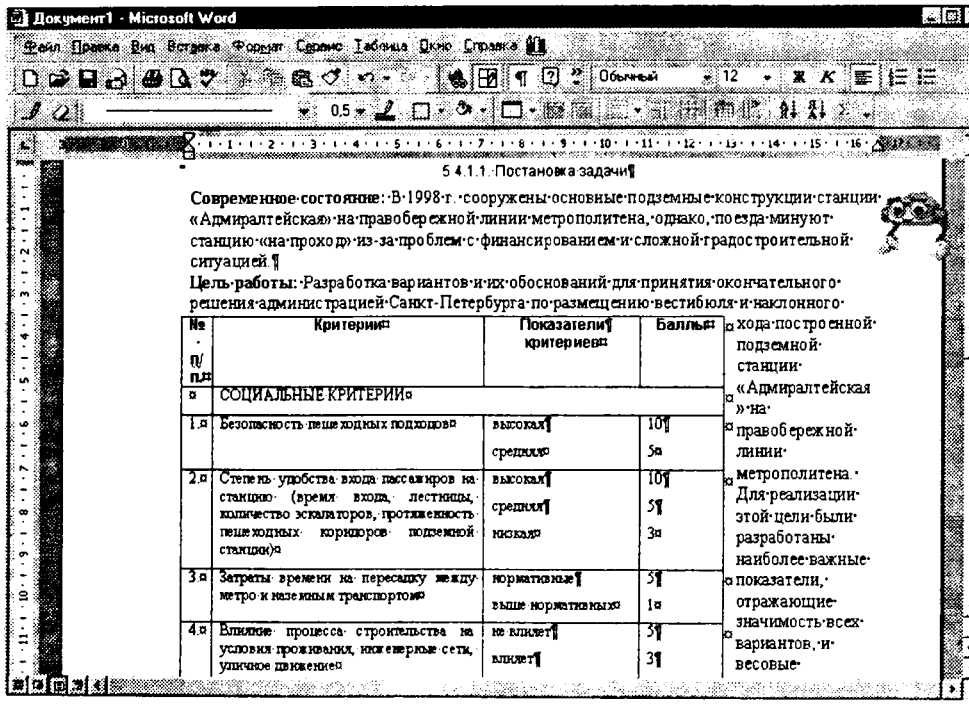


Рис. 5.26. Создание «плавающей таблицы»

При желании пользователь может не рисовать таблицу на всю ширину печатной области, а Word 2000 предоставляет новую возможность задания различных комбинаций таблицы и текста. Это осуществляется следующим образом: необходимо поместить курсор мыши в пределах таблицы, активизировать меню Таблицы ▸ Свойства таблицы. В появившемся диалоговом окне на вкладке Таблица надо задать Обтекание текстом.

Графическое представление данных

Сводные данные по оценке возможных вариантов строительства, получаемые на основе табл. 5.4, представлены в табл. 5.5.

Для решения задач аналитического характера очень часто оказываются полезными не столько табличные, сколько визуальные формы представления данных, такие как графики, столбчатые и круговые диаграммы, поверхности и т. п. MS Word предоставляет пользователю достаточно продвинутые средства для их создания.

Таблица 5.5. Сводные данные по оценке вариантов строительства

№ пп.	Критерии		Варианты 1	2	3	4	5	6	7	Примечания
1	Стоимость строительства		20	10	3	10	20	10	3	Крайние значения могут различаться на 5%
2	Продолжительность строительства		10	5	3	5	5	5	5	
3	Объемы сноса зданий		3	3	5	5	5	1	5	
4	Объемы расселения жителей		3	3	1	4	5	5	4	
5	Объемы освобождения нежилого фонда		2	2	2	3	3	1	3	
6	Влияние процесса строительства	На инженерные сети	3	1	1	1	1	3	3	
		На уличное движение	3	3	1	1	5	5	3	
7	Степень удобства входа пассажиров на станцию		5	3	3	5	5	5	3	
8	Безопасность пешеходных подходов		5	5	5	10	5	5	5	
9	Затраты времени на пересадку между метро и наземным транспортом		5	5	5	5	3	5	1	
10	Обеспечение нормативных требований эксплуатации пешеходных путей сообщения пассажиров		3	3	3	3	3	3	5	
11	Возможность коммерческого использования ссоружений		5	5	5	3	5	5	3	
12	Сводное количество баллов по каждому варианту		67	48	37	55	65	53	43	

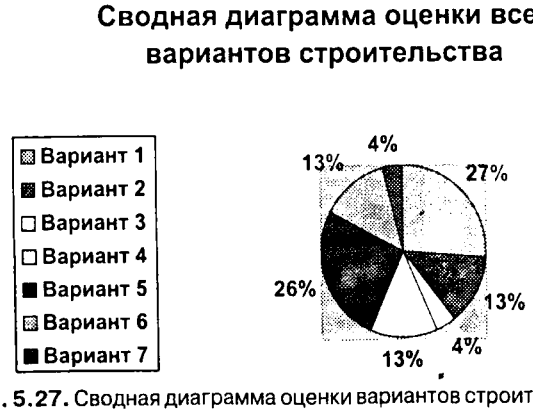


Рис. 5.27. Сводная диаграмма оценки вариантов строительства

Сводная диаграмма была построена при помощи меню Вставка ▸ Рисунок ▸ Диаграмма. При этом активизируется Microsoft Graph, предоставляющий широкие возможности по созданию различного вида диаграмм, из которых для экономистов наибольший интерес, пожалуй, представляют График, Круговая диаграмма и Поверхность. Изменение параметров и вида диаграммы осуществляется в активном окне Microsoft Graph в меню Диаграмма ▸ Параметры диаграммы. В частности, на рис. 5.27 представлена круговая диаграмма, построенная на основе данных, содержащихся в табл. 5.5.

Создание списков

При оформлении документов экономического содержания очень часто приходится иметь дело с текстовыми фрагментами с так называемой регулярной структурой. Например, это может быть нумерованное или ненумерованное перечисление альтернатив чего-либо. В Word для их наименования используется термин *списки*. Примерами списков могут служить следующие два фрагмента.

Оценка социально-экономического эффекта от реализации проекта по двум приоритетным вариантам завершения строительства станции «Адмиралтейская» показала:

1. По условиям возврата в городской бюджет средств, затраченных на строительство, проект является для города неокупаемым.
2. Проект имеет важное социальное значение для жителей города, для создания предпосылок экономического развития в историческом центре, а также для ликвидации кризисной перегрузки участков метрополитена в центре города.
3. По условиям суммарной бюджетной и общественной эффективности проекта срок его окупаемости составляет 5 лет.

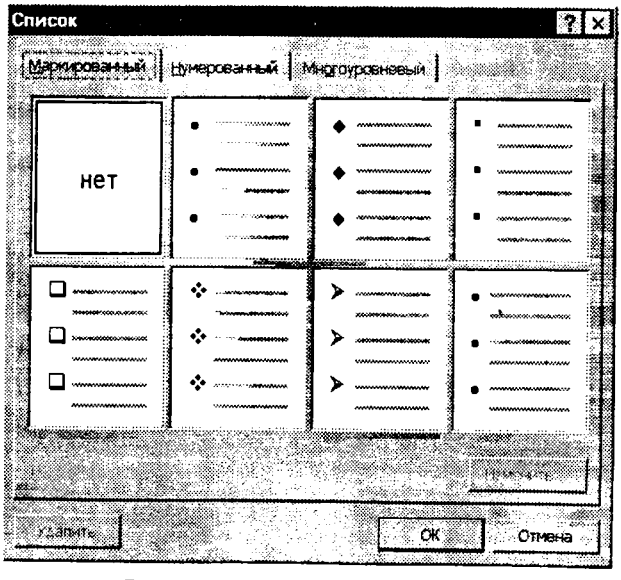


Рис. 5.28. Окно выбора типа списка

В результате сравнения вариантов признаны приоритетными два:

- наземный вестибюль на углу Малой Морской ул. и Кирпичного пер., со сносом 5-этажного здания (43 млн. дол.);
- подземный вестибюль на Большой Морской ул. между Невским пр. и аркой Главного штаба (37 млн. дол.).

При оформлении последних двух подразделов применялись нумерованный и маркированный списки, созданные при помощи меню Формат ▸ Список, где можно выбирать подходящие разновидности списков и наилучший вид маркеров для него, а также задавать свои параметры для списков (рис. 5.28).

Формирование оглавления документа

Теперь, когда основная работа по разработке инвестиционного проекта завершена, осталось только собрать воедино перечень всех его разделов с тем, чтобы максимально облегчить поиск необходимой информации. Word предоставляет прекрасную возможность *автоматически собрать оглавление*. Особенно удобно это в том случае, когда экономический документ функционирует в основном в электронном виде. Тогда вы можете, щелкнув мышью на номере страницы соответствующего раздела в оглавлении, автоматически перейти на эту страницу документа. Не зря в самом начале разработки проекта мы акцентировали внимание на стилях. Если выбор стилей Заголовок N, основного текста, прочих параметров форматирования был выполнен аккуратно, то сборка оглавления не должна вызвать проблем. Для вставки оглавления в текст документа следует:

- выполнить команду меню Вставка ▸ Оглавление и указатели...;
- в появившемся диалоговом окне выбрать вкладку Оглавление (рис. 5.29);
- в списке Вид выбрать вариант оформления оглавления и нажать кнопку OK.

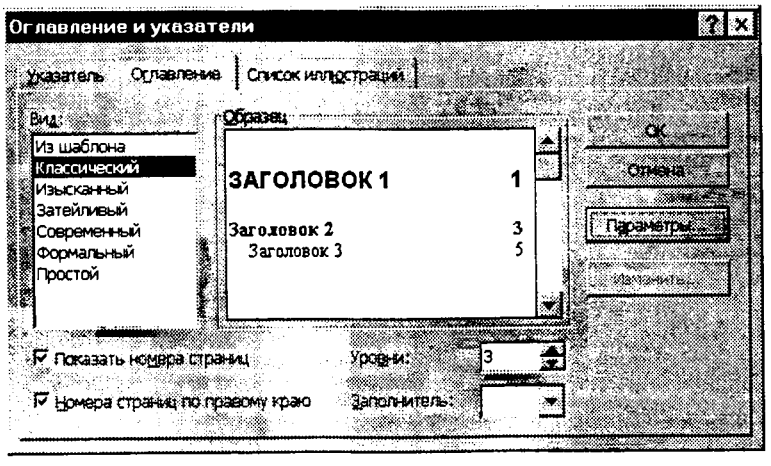


Рис. 5.29. Окно выбора стиля оглавления документа

ПРИМЕЧАНИЕ

Как нетрудно заметить, стили оформления оглавления, представленные в списке Вид, являются «базовыми». При желании пользователь может изменить их настройки (см. кнопка Изменить).

Пример оглавления, которое может иметь наш гипотетического проект, представлен на рис. 5.30.

Инвестиционный проект. Концепция завершения строительства станции метрополитена «Адмиралтейская»

Раздел 1. Показатели критериев и баллы оценки вариантов размещения вестибюльного комплекса 45

Раздел 2. Оценка различных вариантов строительства 47

Раздел 3. Завершение разработки и оформления инвестиционного проекта 48

Подраздел 3.1. Оценка социально-экономического эффекта 48

Подраздел 3.2. Выбор конкретного варианта размещения вестибюльного комплекса 49

Рис. 5.30. Собранное оглавление документа

Ключевые понятия

- текстовый процессор;
- способы ввода информации;
- распознавание текстовой и голосовой информации;
- TrueType;
- шрифт;
- программы автоматизированного перевода;
- PostScript;
- форматы текстовых документов;
- ASCII;
- PDF-формат;
- символ, специальный символ;
- шаблон;
- стиль;
- автотекст;
- макрос;
- список;
- оглавление.

Контрольные вопросы

1. Назовите основные этапы обработки текстовых документов. В чем заключается их специфика?
2. В чем отличие растровых шрифтов от векторных?
3. Перечислите форматы текстовых документов и их отличительные особенности.
4. Сформулируйте основное функциональное предназначение формата PDF.
5. В чем специфика обработки документа системами автоматизированного перевода?
6. Охарактеризуйте преимущества текстового процессора перед текстовым редактором.
7. Каким образом осуществляется работа в Word с документами, находящимися не в формате DOC?
8. Каковы принципиальные отличия Word 2000 от предыдущих версий?
9. Понятие макросов, их преимущества и недостатки.
10. Каковы отличия логического форматирования от физического?
11. Чем отличается стиль абзаца от стиля символа?
12. Что такое «плавающая таблица» в Word 2000?
13. Как построить график функции средствами MS Word?

Литература

1. Microsoft Office 2000: справочник/Под ред. Ю. Колесникова. СПб.: Питер, 1999.
2. Герцог К. Word 2000 без проблем. М.: БИНОМ, 1999.
3. Стоцкий Ю. Самоучитель Office 2000. СПб.: Питер, 1999.
4. PC World. 1999–2000 гг.

Глава 6. Процессоры электронных таблиц

В этой главе мы остановимся на классе программного обеспечения, который имеет крайне важные и содержательные экономические приложения. Более того, с высокой степенью уверенности можно утверждать, что в настоящее время в области экономики и финансов чаще всего применяются именно программы данного класса. Это — *процессоры электронных таблиц*. Активное развитие программ данного вида началось с наступлением эпохи персональных компьютеров, хотя в той или иной форме они существовали и раньше. Среди наиболее известных программных систем, предназначенных для ведения электронных таблиц, могут быть названы SuperCalc, Lotus-1-2-3, Quattro Pro, MS Excel, а в главе 5, к примеру, упоминался такой программный продукт, как StarCalc. MS Excel в силу своего массового распространения в России выбран нами в качестве базового представителя программ данного класса. На его примере будут изложены практические вопросы работы с электронными таблицами.

6.1. MS Excel: общая характеристика и функциональные возможности

6.1.1. Понятие электронной таблицы

Начиная разговор об Excel, прежде всего хочется обратить внимание на тот качественный скачок, который происходит при переходе от электронного аналога текстовых документов к электронным таблицам. Ценность любой информации в значительной мере определяется качеством ее организации, и, более того, существенная доля затрат на обработку информации связана с приданием ей той или иной логической структуры. Особенностью *электронных таблиц* является то, что в них структурирование информации начинается непосредственно на этапе ввода данных: с самого начала своего существования в машинной форме они привязываются к структурным подразделениям таблиц — *ячейкам*.

Как показал опыт развития прикладного программного обеспечения, идея таблично ориентированной организации данных, несмотря на кажущуюся простоту, оказалась весьма плодотворной.

Основное назначение процессоров электронных таблиц — обработка таблично организованной информации (данных, представленных в виде строк и столбцов чи-

сел), проведение расчетов на ее основе и обеспечение визуального представления хранимых данных и результатов их обработки (в виде графиков, диаграмм и т. п.).

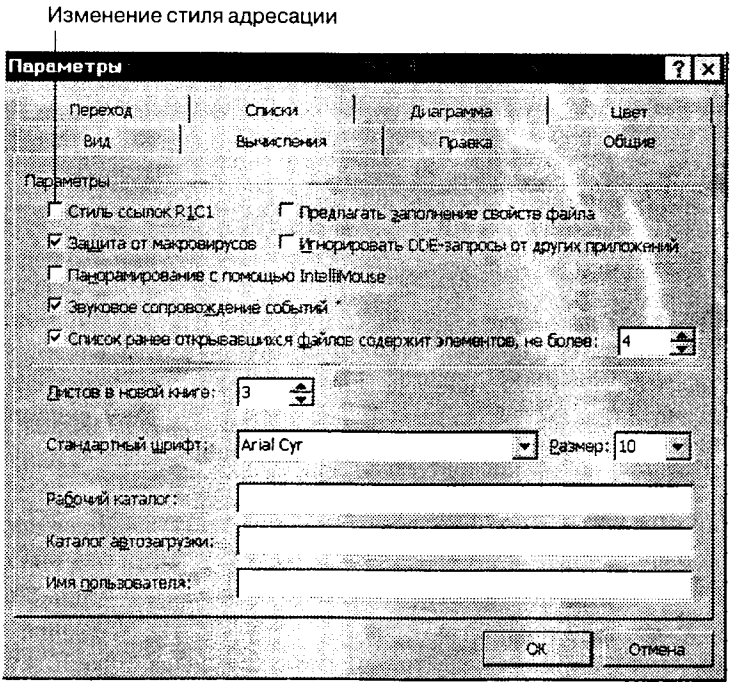


Рис. 6.1. Задание типа ссылок на ячейки

Структурно основным объектом Excel является *рабочая книга*, которая сохраняется как целостный объект в едином файле, имеющем по умолчанию расширение *.xls. Книга делится на *листы*, а листы, в свою очередь, — на *ячейки*. В некоторых случаях логика работы с данными требует задания ссылок между несколькими рабочими книгами. Получающаяся в результате связанная совокупность книг обычно называется *рабочим пространством* или *рабочей средой*. Сведения о настройках рабочего пространства могут быть сохранены в файле специального формата. По умолчанию он имеет расширение *.xlw.

Лист рабочей книги Excel имеет матричную структуру. По умолчанию каждый рабочий лист имеет (максимально) 256 столбцов с именами от A до IV[1] и 16 384 строки с соответствующими номерами. Имя столбца и номер строки, которым одновременно принадлежит ячейка, однозначно определяют ее адрес: A1 — адрес ячейки, находящейся в столбце A и первой строке. Excel поддерживает и другую систему адресации (стиль ссылок), когда нумеруются и строки, и столбцы (в этом случае сначала указывается номер строки, а затем номер столбца. R1C1 — адрес той же ячейки: строка № 1 — Row 1 и столбец № 1 — Column 1.). Изменить стиль ссылки можно с помощью команды Параметры меню Сервис, выбрав вкладку Общие.

[1] То есть столбцы могут именоваться либо одной, либо сочетанием из двух латинских литер.

Очевидно, что данная система адресации позволяет однозначно идентифицировать ячейки в пределах отдельного листа. В расширенном формате адрес ячейки имеет вид

[Имя_файла_рабочей_книги]<имя_листа>!<адрес_ячейки>,

например: **[Книга1.xls]Лист1!A1**

Ячейкам (блокам ячеек) также можно присваивать *собственные имена* и использовать их для ссылок на ячейки наряду с адресами:

1. Выделить ячейку (блок ячеек).
2. Выбрать пункт меню Вставка ▸ Имя ▸ Определить.
3. Указать имя ячейки (блока ячеек), начинающееся с буквы.

Использование содержательных имен вместо «слепых» адресов при работе с ячейками, как правило, является элементом хорошего стиля работы: хотя их присвоение на начальном этапе требует дополнительных трудозатрат от пользователя, в последующем они позволяют лучше ориентироваться в больших массивах информации со сложными взаимозависимостями.

Принципиальным решением, реализованным во всех табличных процессорах, является то, что ячейки электронных таблиц могут содержать данные произвольного типа (по мере необходимости пользователь может загружать в них самую различную информацию). В MS Excel допускаются данные числового, текстового, логического типа, а также даты, массивы, OLE-объекты и др. Наконец, что наиболее существенно, ячейка может содержать формулу, задающую математические или иные операции над данными из других ячеек.

MS Excel является одной из составляющих интегрированной программной системы автоматизации офисной деятельности Microsoft Office. Он хорошо взаимодействует с другими программными приложениями, входящими в ее состав. Прежде всего это касается возможности создавать собственные программные расширения, работающие в среде Excel. Принципиальным свойством процесса создания таких программных надстроек является то, что как в Excel, так и в других программных продуктах, входящих в семейство MS Office, для этого используется среда программирования Visual Basic for Applications (VBA), что, в конечном счете, открывает широкие возможности для их последующей тесной интеграции.

Важнейшей особенностью процессоров электронных таблиц является их способность обеспечивать автоматический пересчет и обновление связей при вводе или изменении данных. Как только вводятся новые данные, электронная таблица мгновенно проводит перерасчет по ранее заданным формулам, и информация моментально обновляется. Эта особенность таблиц с успехом используется для анализа многовариантных ситуаций или ответа на вопросы типа «Что будет, если...?».

6.1.2. Общая характеристика интерфейса MS Excel

Вид главного окна Excel, которое показывается пользователю непосредственно после запуска программы, представлен на рис. 6.2.

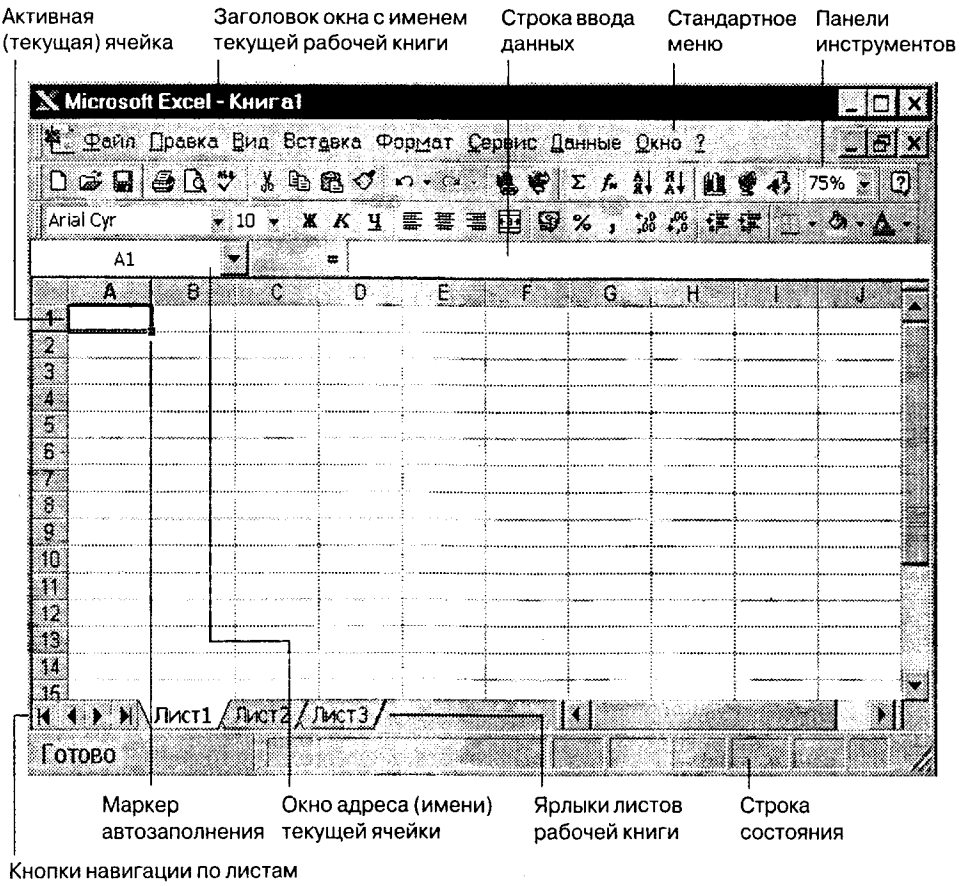


Рис. 6.2. Стандартный вид окна электронной таблицы

Среди основных интерфейсных элементов окна могут быть названы:

- *строка меню* и расположенные на ней меню основных режимов (иерархические раскрывающиеся списки);
- *панели инструментов*;
- *строка ввода данных*;
- *окно адреса активной (текущей) ячейки*;
- *ярлыки листов рабочей книги и кнопки навигации по ним*;
- *линейки прокрутки*.

Панели инструментов — значки, кнопки, раскрывающиеся списки, благодаря которым возможен быстрый доступ к часто используемым командам и настройкам. По умолчанию, если не сделано определенных установок, Excel выводит на экран Стандартную панель инструментов и Панель форматирования. Чтобы узнать название кнопки или пиктограммы, нужно подвести к ней указатель мыши — Excel выдаст подсказку, чтобы получить полную информацию о назначении кнопки, можно воспользоваться справкой — ?.

Чтобы вывести на экран дополнительные панели инструментов, нужно воспользоваться меню Вид ▸ Панели инструментов..., где выбрать необходимую панель.

Строка ввода данных — в ней отображается содержимое активной ячейки.

Строка состояния — отображает сведения о выбранной команде или выполняемой операции, а также справочную информацию о включенных функциональных клавишах.

Очевидно, что в каждый конкретный момент нам визуально может быть доступна лишь некоторая часть данных электронной таблицы. В связи с этим особое значение в плане оптимизации технологии работы с данными приобретают навигационные возможности программы.

Перемещаться по листу рабочей книги можно с помощью:

- клавиатуры (соответствующие комбинации клавиш и стрелки управления курсором);
- мыши (полосы прокрутки);
- команды меню Правка ▸ Перейти. В поле Ссылка указать адрес нужной ячейки, чтобы перейти к определенной ячейке нужного рабочего листа.

Основные возможности по перемещению внутри отдельного листа описаны в табл. 6.1.

Таблица 6.1. Клавиши перемещения по листу рабочей книги Excel

Стрелки управления курсором	На одну ячейку в направлении стрелки
Ctrl+ стрелки управления курсором	Переход к соответствующей части листа, содержащей данные
PageUp, Page Down	К верхней и нижней частям экрана
Ctrl+End, Ctrl+Home	К нижнему или верхнему левому углу рабочего листа
Home	К крайней слева ячейке листа
End	К последнему столбцу в строке

Для перемещения между листами рабочей книги можно либо воспользоваться кнопками навигации (рис. 6.3), либо непосредственно выбрать с помощью мыши ярлычок нужного листа.

В заключение разговора об общих принципах организации пользовательского интерфейса Excel отметим, что в нем, как и в подавляющем большинстве других Windows-приложений, одни и те же операции могут быть выполнены несколькими способами:

- с помощью команд главного меню;
- с помощью функций, присвоенных пиктограммам панелей инструментов;
- с помощью команд контекстных меню, активизируемых по щелчку правой клавиши мыши;
- нажатием специальных клавиатурных комбинаций.

В случае электронных таблиц умелое сочетание различных методов управления позволяет достигнуть существенного повышения скорости и эффективности работы.

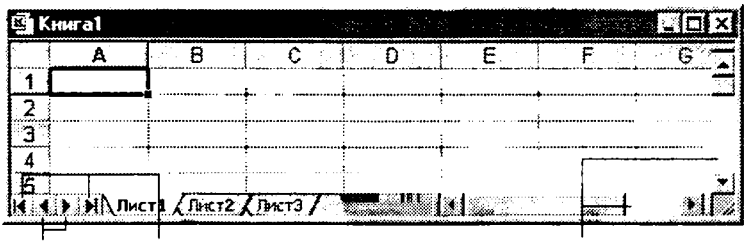


Рис. 6.3. Перемещение между листами рабочей книги и по экрану

6.1.3. Технология ввода данных в MS Excel

Как уже отмечалось ранее, ячейка предназначена для того, чтобы хранить различные значения различных типов. Она имеет уникальный адрес, может иметь имя, может иметь и менять значения.

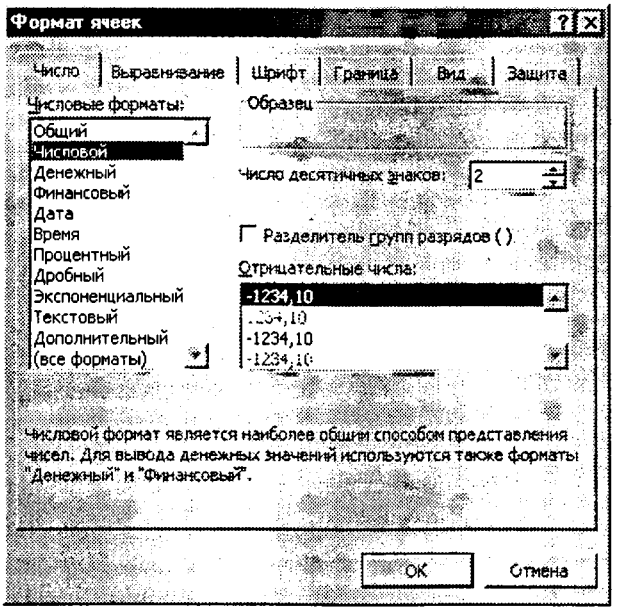


Рис. 6.4. Управление форматом ячеек

Интервал (блок, диапазон) ячеек — прямоугольная область смежных или несмежных ячеек. Он задается указанием адреса верхней левой и правой нижней ячеек, разделенных символом двоеточие. Например: **A2:G4**.

ПРИМЕЧАНИЕ

Выделение несмежного диапазона ячеек производится с помощью мыши при нажатой клавише Ctrl.

Ячейки имеют заданный *формат*, который можно установить из меню Формат командой Ячейки. Эта команда имеет несколько вкладок, из вкладки Число можно выбрать категорию значений в ячейке и задать код формата, в том числе можно создать пользовательский формат.

Чтобы ввести данные в ячейку, следует:

1. Сделать ячейку активной — поместить курсор в нужную ячейку и один раз щелкнуть левой клавишей мыши.
2. Ввести данные прямо с клавиатуры в *режиме редактирования ячейки* (курсор ввода находится внутри ячейки).

Или:

1. Дважды щелкнуть левой клавишей мыши.
2. Ввести данные в *режиме редактирования строки формул* (курсор ввода находится в строке формул над рабочим листом), текст будет автоматически появляться и в строке формул.
3. Завершить ввод одним из трех способов:
   - нажатием клавиши Enter;
   - щелчком мыши вне рабочего пространства изменяемой ячейки;
   - щелчком мыши по «зеленой галочке» в строке формул.

Для того чтобы изменить (отредактировать) уже введенные данные, следует:

- очистить ячейку (клавиша Delete) и повторить ввод данных;
- исправить (отредактировать) содержимое ячейки, для этого:
  - войти в режим редактирования, дважды щелкнув левой клавишей мыши или нажать функциональную клавишу F2;
  - изменить данные.

В ячейки можно вводить данные различных *типов*: текст, числа, даты, время, формулы (которые могут включать в себя числа, знаки арифметических, логических или других действий, производимых с данными из других ячеек, адреса ячеек или их имена, имена встроенных функций), функции (частный случай формулы).

В качестве текста можно вводить различные комбинации букв и цифр. Текст по умолчанию выравнивается по левому краю, а числа — по правому.

При работе в Excel часто бывает так, что расчеты проводятся быстрее, чем ввод исходных данных. Excel предусматривает различные средства контроля ввода данных, например, можно указать диапазон значений, в котором должно быть заключено число.

В Excel существует несколько способов, которые позволяют упростить и ускорить процесс ввода данных:

- повторный ввод (копирование) уже существующих данных;
- автозаполнение;
- ввод прогрессий.

Для повтора (копирования) существующих данных следует:

1) выделить ячейку;
2) подвести указатель мыши к маркеру автозаполнения и перетащить рамку выделения по всем ячейкам.

Аналогичного результата можно добиться, если воспользоваться меню Правка ▸ Заполнить и указать направление копирования.

Автозаполнение — заполняет выделенные ячейки выбранными (либо созданными дополнительно) последовательностями.

Рассмотрим пример. Пусть необходимо заполнить диапазон ячеек названиями месяцев (Январь, Февраль и т. д.). Для этого нужно:

1. Ввести в первую ячейку слово — Январь.
2. Перетащить маркер автозаполнения в нужном направлении (вверх, вниз, влево, вправо).

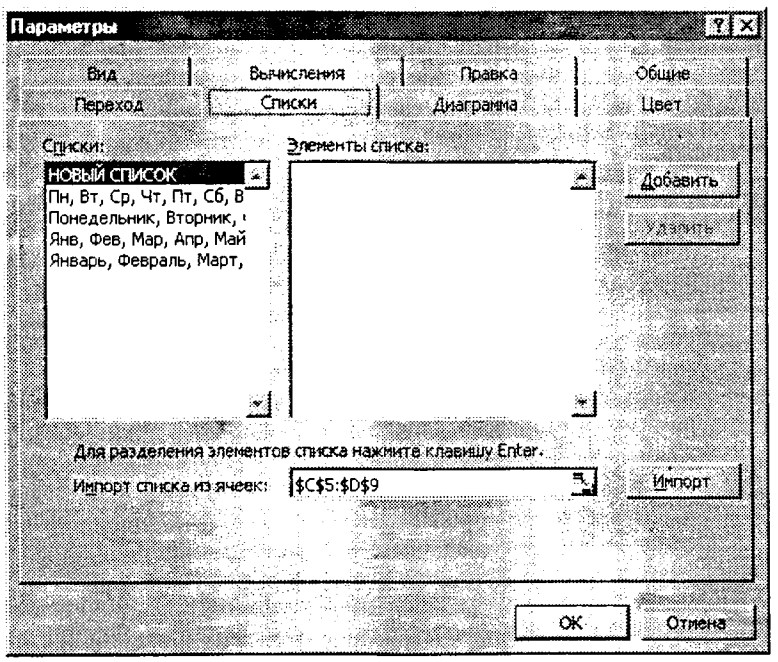


Рис. 6.5. Создание нового списка для автозаполнения

Для того чтобы создать собственную последовательность для автозаполнения, нужно:

1. Выполнить команду меню Сервис ▸ Параметры и выбрать вкладку Списки (рис. 6.5).
2. В элементе управления Списки сделать активным пункт Новый список.

3. В поле Элементы списка ввести последовательность строк, которые должны образовать новый список. При этом каждый элемент списка вводится с новой строки.
4. Процесс формирования нового списка завершить нажатием кнопки Добавить.

С помощью автозаполнения можно, набрав любой (не обязательно первый) элемент последовательности, вставить все остальные (причем произвольное количество раз).

Функция *ввода прогрессий* позволяет оперативно создавать последовательности из возрастающих (убывающих) по арифметическому или геометрическому закону чисел, дат и т. п.

Для создания прогрессии нужно:

1. Ввести значение первого элемента прогрессии в выбранную базовую ячейку.
2. Выделить блок ячеек, который должны будут занять члены прогрессии (либо в дальнейшем придется указать значение последнего элемента).
3. Выполнить команду меню Правка ▸ Заполнить ▸ Прогрессия.
4. В появившемся диалоговом окне Прогрессия указать тип и параметры формируемой последовательности значений (рис. 6.6).

В частности, на рис. 6.6 приводится пример создания геометрической прогрессии из 10 членов с первым членом 3 и основанием 5.

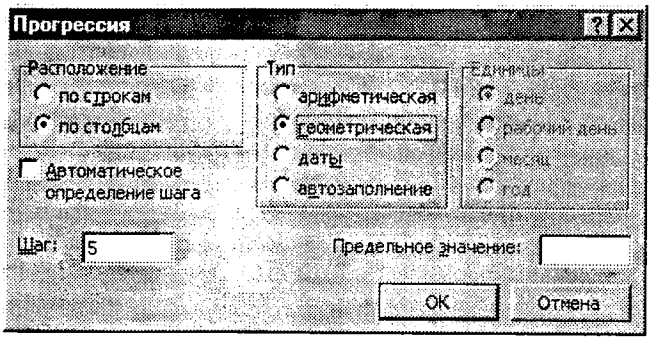


Рис. 6.6. Задание параметров создаваемой прогрессии

В результате получим диапазон ячеек, заполненный членами геометрической прогрессии, показанный на рис. 6.7.

6.1.4. Формулы, функции, мастер функций

Формула — это выражение, состоящее из операндов, которое используется для выполнения вычислений над имеющимися данными. Каждая формула начинается с символа равно (=). В качестве операндов могут быть: числа, логические значения, ссылки на адреса ячеек (диапазон ячеек), встроенные функции, которые соединяются с помощью символов операций — сложения, вычитания, умножения,

деления, возведения в степень. Сама формула — это тоже значение (результат вычислений), которое хранится в ячейке рабочего листа.

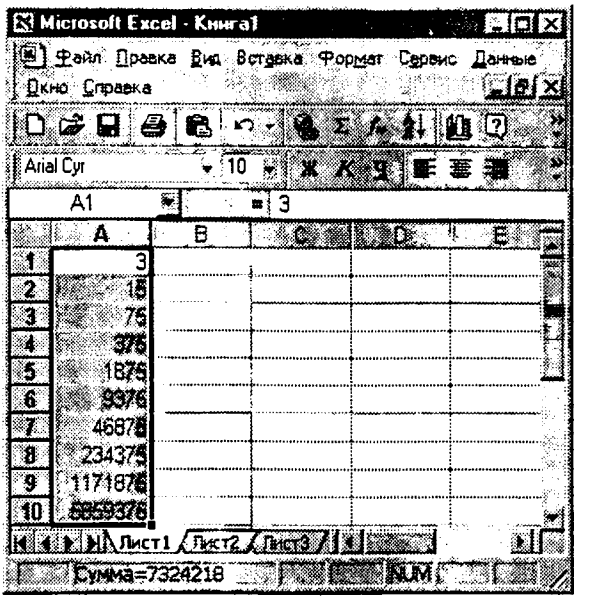


Рис. 6.7. Диапазон, заполненный геометрической прогрессией

Excel выполняет вычисления последовательно, но некоторые операции имеют более высокий приоритет и выполняются раньше других:

1. Возведение в степень и выражения в круглых скобках.
2. Умножение и деление.
3. Сложение и вычитание.

Технические возможности Excel позволяют создавать достаточно сложные формулы. Однако подлинная мощь Excel как программного средства реализуется через широкий набор встроенных функций, предназначенных для выполнения самых различных вычислительных и логических процедур.

Несмотря на их разнообразие, все функции имеют одинаковый стандартный формат: имя функции и находящийся в круглых скобках перечень аргументов, разделенных точками с запятой. Например:

```
=ОКРВВЕРХ(B3;2).
```

Существенную помощь пользователю в процессе конструирования выражений, содержащих формулы, может оказать Мастер функций. Вызвать его можно одним их следующих способов:

- меню Вставка ▸ Функция;
- кнопка Вставка функции на панели Стандартная.

В Excel редактирование формул производится так же, как и редактирование любых введенных данных:

- выделить ячейку, содержащую редактируемую формулу, нажать клавишу F2, внести изменения, нажать клавишу Enter;
- быстрая правка — двойной щелчок по ячейке, нажать клавишу F2, внести изменения, нажать Enter.

Копируются формулы так же, как и другие данные:

- выделить ячейку, меню Правка ▸ Копировать (или комбинация клавиш Ctrl+C); выделить целевую ячейку (ячейки), то есть область, куда надо поместить копируемые данные; меню Правка ▸ Вставить (или Ctrl+V);
- быстрое копирование: выделить ячейку, содержащую формулу, и перетащить границу выделенной ячейки в новое место, удерживая нажатой клавишу Ctrl.

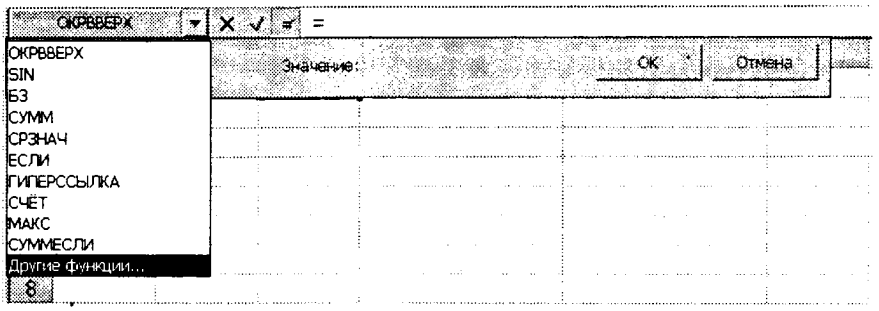


Рис. 6.8. Ввод функции

Принципиальным при копировании и перемещении формул является вопрос о преобразовании содержащихся в них ссылок на другие ячейки. Очевидно, что в зависимости от внутренней логики выражений в определенных случаях адреса должны оставаться неизменными, а в других ситуациях для пользователя будет более удобным и предпочтительным, чтобы их пересчитали с учетом относительного изменения местоположения. Для решения данной задачи в процессорах электронных таблиц поддерживается система относительных и абсолютных ссылок.

Абсолютная ссылка — это не изменяющийся при копировании и перемещении формулы адрес ячейки, содержащий исходные данные. В качестве признака абсолютной ссылки в адресе используется знак $. Различают:

- *полную абсолютную ссылку* (знак $ ставится и перед именем столбца, и перед номером строки, например **A8**). В этом случае при копировании и перемещении адрес ячейки не меняется;
- *частичную абсолютную ссылку* (знак $ ставится либо перед номером строки, либо перед именем столбца, например **A$8** или **$A8**. В данном случае при копировании и перемещении неизменной остается только одна из координат).

Для быстрого изменения типа адресации в формулах используется функциональная клавиша F4 (при ее последовательном нажатии вид адреса, находящегося в строке ввода данных, попеременно меняется с относительного на абсолютный, с абсолютного на частично абсолютный и т. д.).

Обратим внимание на то, что при работе с экономическими приложениями к необходимости применения абсолютной адресации пользователь приходит, если так можно выразиться, естественным путем. В качестве хрестоматийного примера в литературе обычно приводится задача расчета заработной платы с помощью электронных таблиц, где в формулах для суммы налога содержится абсолютная ссылка на ячейку, содержащую ставку налогообложения.

В случае *относительных ссылок* при копировании и перемещении формул происходит пересчет адресов ячеек, на которых они указывают, с учетом относительного изменения места этих формул на листе. Как нетрудно заметить, по умолчанию используется именно относительная адресация.

6.1.5. Графические возможности Excel

Среди задач, решаемых с помощью процессоров электронных таблиц, особое место занимает графическое представление хранимой информации и результатов ее обработки. Наглядность и легкость восприятия диаграмм позволяет принять быстрое и обоснованное решение по дальнейшей обработке данных числового типа, так как даже самые простые диаграммы могут помочь оценить имеющиеся данные лучше, чем изучение каждой ячейки рабочего листа. Графическое представление также может помочь обнаружить ошибку в данных (просто неверно введенные данные) и даже на первом этапе изучения данных поможет раскрыть важные зависимости, скрытые в больших массивах чисел.

Существенным достоинством Excel как современного программного инструмента является то, что при работе с ним при изменении исходных данных, по которым построены график или диаграмма, автоматически изменяется и их изображение.

С помощью Excel можно строить диаграммы разных типов. Некоторые из них могут быть «объемными» (они выглядят очень эффектно и помогают подчеркнуть различия между разными наборами данных).

В зависимости от места расположения и особенностей построения и редактирования различают два вида диаграмм:

- *внедренные диаграммы* — помещается на том же рабочем листе, где и данные, по которым они построены;
- *диаграммы в формате полного экрана* на новом рабочем листе.

Оба типа диаграмм связаны с данными рабочего листа и автоматически обновляются при изменении данных.

Удобным средством для создания графических представлений в Excel является Мастер диаграмм, который вызывается специальной кнопкой на панели Стандартная или из меню Вставка ▸ Диаграмма (рис. 6.9).

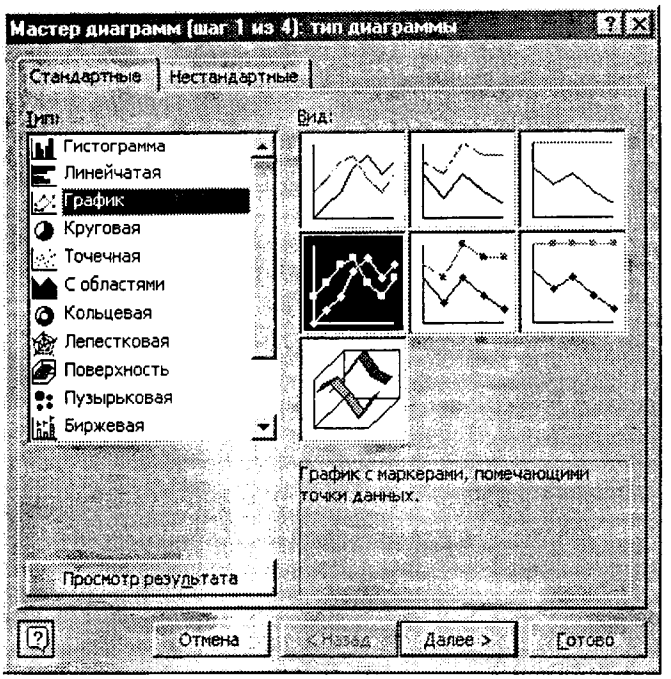


Рис. 6.9. Окно мастера диаграмм

При выборе нужного параметра на четвертом шаге Мастера диаграмм (рис. 6.10) определяется, будет ли создан специальный диаграммный лист или внедренная диаграмма.

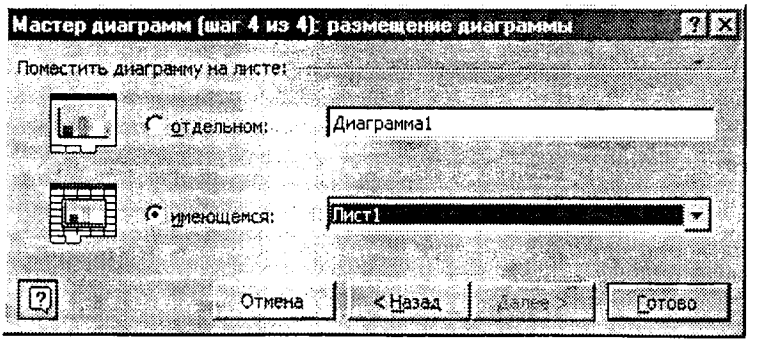


Рис. 6.10. Мастер диаграмм, шаг 4 — построение внедренной диаграммы

Правильно выбранный способ представления данных на диаграмме и удачное форматирование очень важны для качественного анализа данных.

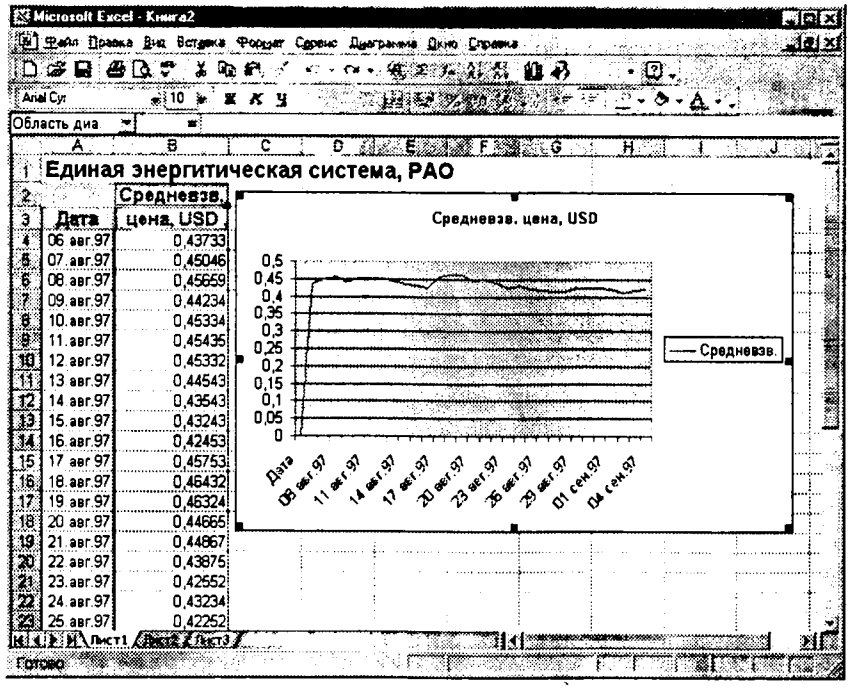


Рис. 6.11. Представление данных в виде графика

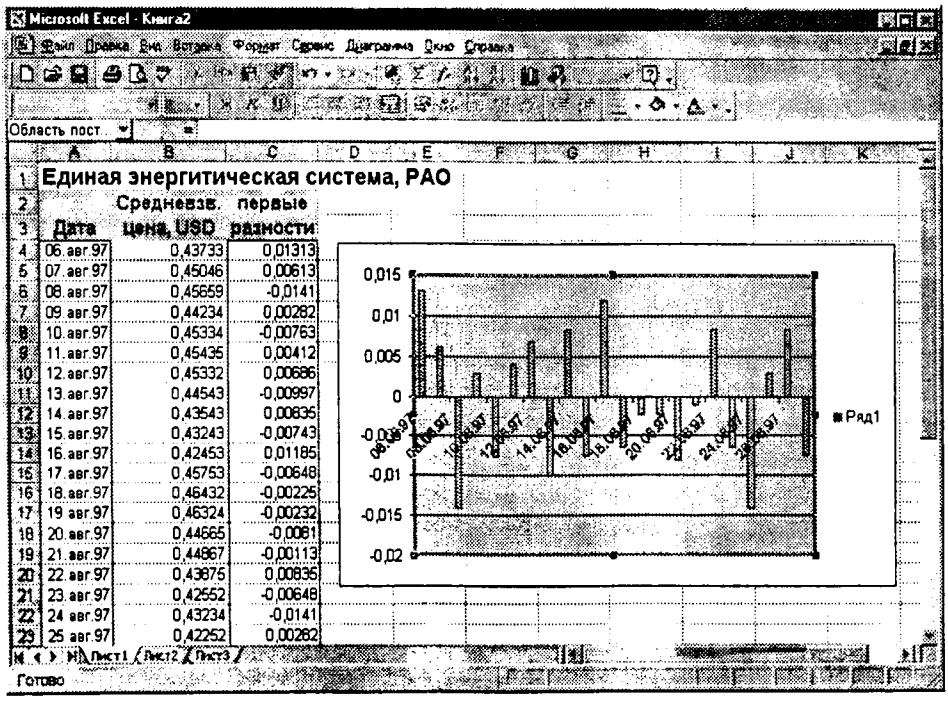


Рис. 6.12. Представление данных в виде гистограммы

Внедренную диаграмму можно перемещать по экрану и изменять ее размеры. Для этого нужно щелкнуть мышью на любой ее части и перетащить в нужное место рабочего листа. Чтобы изменить размер диаграммы, ее нужно выделить (щелчок мышью), а затем перетащить один из маркеров изменения размеров.

Уже созданную диаграмму можно усовершенствовать (дополнительно добавить название, отформатировать уже имеющийся текст в диаграмме, дать название осям или изменить их масштаб).

Перед тем как *редактировать диаграмму*, ее нужно открыть в окне (двойной щелчок мышью по диаграмме переводит ее в режим правки). Далее щелкнуть правой кнопкой мыши по тому элементу диаграммы, который нужно редактировать (например, оси или текст), и, наконец, в контекстном меню выбрать соответствующий пункт и заполнить поля.

6.2. Экономико-математические приложения MS Excel

Основная цель настоящего параграфа — продемонстрировать, что MS Excel представляет собой не просто удобное средство для выполнения математических и логических операций, а мощный и универсальный инструмент по решению достаточно серьезных задач, возникающих в сфере экономики и финансов.

Среди типичных экономико-математических приложений Excel могут быть названы:

- структуризация и первичная логическая обработка данных;
- статистическая обработка данных, анализ и прогнозирование;
- проведение финансово-экономических расчетов;
- решение уравнений и оптимизационных задач.

6.2.1. Средства структуризации и первичной обработки данных

Ранее уже отмечалось, что к принципиальным преимуществам электронных таблиц как формы организации данных относится гибкое сочетание возможностей, которые дает привязка информации к ячейкам таблицы, со свободой принятия решений о том, какую именно ячейку выбрать для размещения информации. Однако данная «свобода» имеет и обратную сторону: зачастую за нее приходится платить на дальнейших этапах работы, когда возникают проблемы с выполнением тех или иных операций по автоматизированной обработке данных. Не секрет, что логика таких операций предполагает наличие жестко формализованной структуры у исходной информации. Одним из способов решения этой проблемы в Excel являются списки.

Список — это содержащаяся в рабочем листе Excel таблица, данные в строках которой имеют однородную структуру, то есть в каждом столбце списка располагают-

ся данные одного типа (число, текст, дата и т. п.[1]). Для наименования ячейки, относящейся к некоторой строке списка и содержащей данные соответствующего типа, используется термин *поле*. Поля списка могут иметь имена, которые должны располагаться в первой строке таблицы.

К традиционным задачам первичной (предварительной) логической обработки данных относятся *сортировка* и *выборка по заданному критерию* (фильтрация).

В Excel возможна сортировка по одному или нескольким ключевым полям, в том числе с подведением общих итогов и промежуточных итогов по группам записей. Для того чтобы отсортировать данные, следует:

- выполнить команду меню Данные ▸ Сортировка;
- в появившемся диалоговом окне выбрать параметры сортировки. Как видно из рис. 6.13, с их помощью можно задать три уровня ключей, которые будут использоваться при упорядочении данных, направления упорядочения для каждого уровня (возрастание/убывание), а также указать наличие (отсутствие) у полей списка имен.

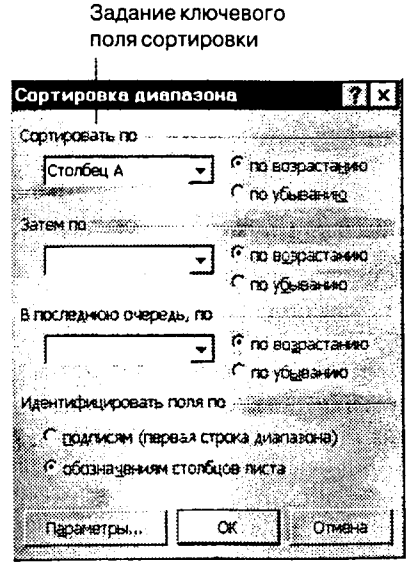


Рис. 6.13. Задание параметров сортировки данных

Если из общего списка необходимо выбрать лишь часть информации, соответствующую некоторым условиям, то следует провести фильтрацию данных. Для простых условий фильтрации часто бывает достаточно средств Автофильтра.

[1] Более подробно вопросы работы с массивами, имеющими «жесткую» табличную структуру, будут рассмотрены в главе 7, посвященной базам данных. Также отметим, что здесь присутствует определенная неоднозначность в применении термина «таблица», который используется и в контексте «электронная таблица», и при определении понятия «список».

Ограничимся кратким описанием последовательности действий:

1. Выделить всю область списка (включая заголовок).
2. Выполнить команду меню Данные ▸ Фильтр ▸ Автофильтр, после чего в строке заголовков появятся кнопки с раскрывающимися списками значений.
3. С помощью появившихся элементов управления кнопок могут быть заданы стандартные критерии отбора (Все, Первые 10..., все строки с конкретным значением), см. рис. 6.14.

	A	B	C	D	E	F
1	Россия	6,8649	3	21,	10 янв	Пн
2	(Все)	1,287477	8,9	6,99	11 янв 00	Вт
3	(Первые 10...)	6,429249	6,9	9,26	12 янв 00	Ср
4	(Условие...)	3,769638	8,0	8,71	13 янв 00	Чт
5	Германия	3,725437	7,3	3,44	14 янв 00	Пт
6	Россия	9,632333	6,9	0,00	15 янв 00	Сб
7	Франция	1,253713	1,9	0,02	16 янв 00	Вс
8	Германия	4,829935	7,2	6,35	17 янв 00	Пн
9	Россия	6,904143	7,6	0,80	18 янв 00	Вт
10	Франция	7,698115	7,0	8,53	19 янв 00	Ср
11	Германия	5,304424	2,0	6,36	20 янв 00	Чт

Рис. 6.14. Задание фильтра

Для отбора строк списка по сложным критериям, например таким, которые содержат вычисляемые выражения, а также для размещения результатов фильтрации в произвольном месте рабочего листа служит команда меню Данные ▸ Расширенный фильтр.

Сводные таблицы — средство обработки и представления данных, намного превосходящее по возможностям и удобству использования традиционные списки с промежуточными итогами. Это интерактивная таблица на рабочем листе, позволяющая подытожить большие объемы данных, выбрав подходящий метод вычислений (а не только суммирование!). Для построения сводной таблицы используются запросы к внешним базам данных. Для этого применяется техника запросов по образцу, и в данном случае списки обязательно должны содержать имена полей в первой строке.

Сводная таблица создается с помощью Мастера сводных таблиц (команда меню Данные ▸ Сводная таблица...). Режим мастера состоит из четырех последовательных этапов, в ходе которых пользователь в режиме диалога задает параметры построения сводной таблицы:

1) первый и второй шаги — выбор источника данных;
2) третий — создание макета сводной таблицы: определение полей сводной таблицы и их ориентации, выбор функции суммирования (сведения) для полей данных (рис. 6.15);
3) четвертый — определение размещения сводной таблицы.

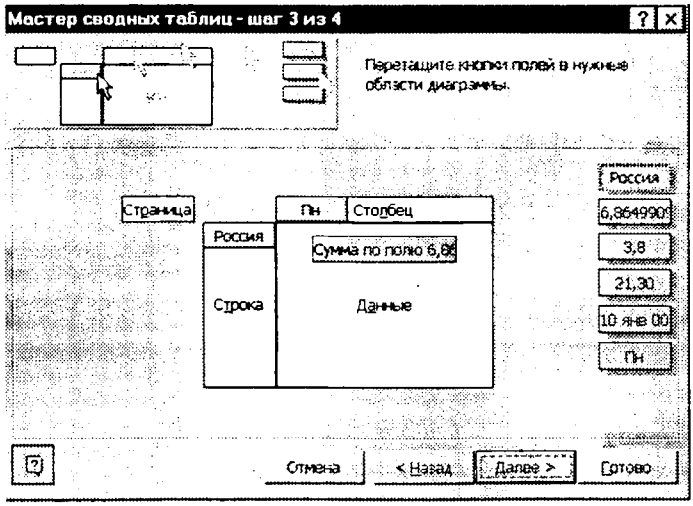


Рис. 6.15. Построение сводной таблицы в режиме мастера — шаг 3

6.2.2. Статистическая обработка данных и прогнозирование

Функции, реализующие статистические методы обработки и анализа данных, в Excel реализованы в виде специального программного расширения — надстройки Пакет анализа, которая входит в поставку данного программного продукта и может устанавливаться (или не устанавливаться) по желанию пользователя.

Установка надстройки Пакет анализа производится точно так же, как и установки прочих надстроек, то есть через меню Сервис ▸ Надстройки, после чего в диалоговом окне Надстройки необходимо пометить пункт Пакет анализа и нажать кнопку OK (рис. 6.16).

Установка пакета анализа

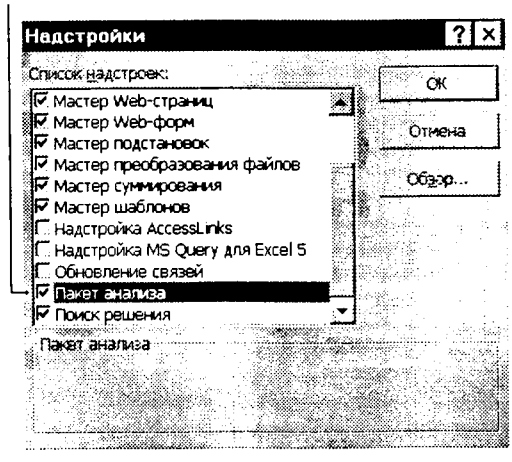


Рис. 6.16. Установка пакета анализа

Если процесс установки завершается успешно, то в меню Сервис появляется еще один пункт — Анализ данных (рис. 6.17), а также при создании формул становится доступной новая группа функций — статистические.

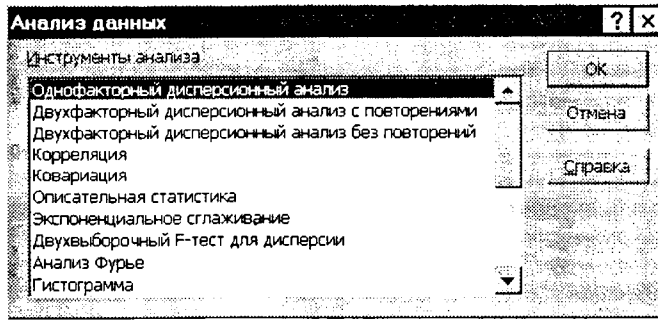


Рис. 6.17. Окно Анализ данных, вызываемое из меню Сервис ▸ Анализ данных

Проблема изучения взаимосвязей различного рода показателей является одной из важнейших в экономическом анализе. В конечном счете, основное содержание любой экономической политики может быть сведено к регулированию экономических переменных, осуществляемому на базе выявленной тем или иным образом информации об их взаимовлиянии.

Целью статистического исследования является обнаружение и исследование соотношений между статистическими (экономическими) данными и их использование для изучения, прогнозирования и принятия решений.

Любые экономические данные представляют собой количественные характеристики каких-либо экономических объектов. Они формируются под действием множества факторов, не все из которых доступны внешнему контролю. Неконтролируемые факторы могут принимать случайные значения из некоторого множества значений и тем самым обусловливать случайность данных, которые они определяют. Стохастическая природа экономических данных обусловливает необходимость применения специальных статистических методов для их анализа и обработки.

Поэтому фундаментальными понятием статистического анализа являются понятия вероятности и случайной величины.

Конечно, Excel не предназначен для комплексного статистического анализа и обработки данных (в отличие от специального статистического программного обеспечения, такого как STATISTICA, Eviews, TSP, SPSS, Microfit и др.). Однако и на базе электронных таблиц можно провести некоторую статистическую обработку данных. В частности, в рамках Excel с помощью команд, доступных из окна Анализ данных (рис. 6.17), можно провести:

- описательный статистический анализ (Описательная статистика);
- ранжирование данных (Ранг и персентиль);
- графический анализ данных (Гистограмма);

- прогнозирование данных (Скользящее среднее, Экспоненциальное сглаживание);
- регрессионный анализ (Регрессия) и др.

Термин *«регрессия»* широко применяется в научной литературе для обозначения так называемой статистической зависимости между двумя (несколькими) сериями значений каких-либо величин. Определение «статистическая» предполагает, что рассматриваемая зависимость реализуется как некоторая общая тенденция, от которой возможны случайные отклонения в ту или иную сторону.

Практические методы определения параметров регрессии (или, как еще говорят, регрессионного анализа) базируются на достаточно сложном математическом аппарате, составляющем предмет таких дисциплин, как математическая статистика, многомерный статистический анализ и др.

В табл. 6.2 приведены *статистические функции*, позволяющие пользователю реализовывать операции регрессионного анализа (выявления зависимостей между рядами данных) непосредственно на рабочем листе электронной таблицы.

Таблица 6.2. Статистические функции для регрессионного анализа

Функция	Назначение функции и ее аргументы	Тип
ЛИНЕЙН	Определяет параметры линейного тренда для заданного массива	Встроенная
	ЛИНЕЙН(знач.У; знач. Х; константа; стат.)	
ТЕНДЕНЦИЯ	Определяет предсказанные значения в соответствии с линейным трендом для заданного массива (метод наименьших квадратов)	Встроенная
	ТЕНДЕНЦИЯ(знач.У; знач. Х; новые знач.Х; константа;)	
ПРЕДСКАЗ	Определяет предсказанное значение функции в заданной точке на основе линейной регрессии	Встроенная
	ПРЕДСКАЗ(Х; знач.У; знач. Х)	
НАКЛОН	Определяет коэффициент для независимой переменной в уравнении парной регрессии	Встроенная
	НАКЛОН(знач.У; знач. Х)	
ОТРЕЗОК	Определяет отрезок, отсекаемый на оси ординат линией линейной регрессии	Встроенная
	ОТРЕЗОК(знач.У; знач. Х)	
КВПИРСОН	Определяет квадрат коэффициента корреляции Пирсона	Встроенная
	КВПИРСОН(знач.У; знач. Х)	
ПИРСОН	Определяет коэффициент корреляции Пирсона (степень линейной зависимости между двумя множествами данных)	Встроенная
	ПИРСОН(массив 1; массив 2)	
СТОШУХ	Определяет стандартную ошибку предсказанных значений У для каждого Х	Встроенная
	СТОШУХ(знач.У; знач. Х)	
РОСТ	Аппроксимирует данные экспоненциальной кривой	Встроенная
	РОСТ(знач.У; знач. Х; новые знач.Х; константа;)	

Перечисленные функции имеют очень широкий спектр экономических приложений. Например, в качестве иллюстрации техники использования статистических функций Excel рассмотрим *задачу построения парной линейной регрессии между доходностью отдельно взятой акции и доходностью рыночного портфеля.*

Сформулированная проблема основывается на подходе к описанию поведения финансового рынка с помощью так называемой модели CAPM (Capital assets pricing model — модель оценивания финансовых активов). Не вдаваясь в подробности ее описания[1], заметим, что одним из базовых ее допущений является предпосылка о возможности описания связи между доходностью акции и доходностью рыночного портфеля[2] с помощью уравнения:

$$Y=\alpha+\beta X+e,$$

где Y — доходность акции; X — доходность рыночного портфеля; e — ошибка регрессионного уравнения.

Коэффициент β в данном уравнении (с точки зрения геометрической интерпретации он является угловым коэффициентом графика) служит важнейшей характеристикой риска изучаемого актива, о котором в зависимости от значения β можно заключить:

- $\beta=1$ — риск ценной бумаги равен рыночному (рыночный риск — стандартное отклонение доходности рыночного портфеля), а доходность изменяется точно так же, как и рыночная;
- $\beta>1$ — риск ценной бумаги выше рыночного. В результате этого доходность ценой бумаги изменяется в том же направлении, что и рыночная, но в большей степени. Таким свойством, как правило, обладают ценные бумаги компаний, оперирующих в энергично развивающихся высокотехнологичных отраслях;
- $0\leq\beta<1$ — риск ценной бумаги ниже рыночного, а ее доходность изменяется в том же направлении, что и рыночная, но в меньшей степени. Это часто встречается у компаний в традиционных отраслях, особенно связанных с пищевой промышленностью и коммунальным хозяйством;
- $\beta<0$ — данный теоретически возможный вариант означает, что доходность такой ценной бумаги изменяется в направлении, противоположном изменению рыночной доходности.

Если в качестве исходной информации мы имеем серии значений доходностей акций и доходностей рыночного (индексного) портфеля по периодам, то, применяя функцию НАКЛОН, мы можем легко получить оценку значения β-коэффициента (рис. 6.18). Заметим, что для оценки значения свободного члена линейного уравнения (α) применяется функция ОТРЕЗОК.

---

[1] Заинтересованный читатель сможет более подробно ознакомиться с моделью CAPM, допустим, в Первозванский А. А., Первозванская Т. Н. Финансовый рынок: расчет и риск. М.: «Инфра», 1994.

[2] Рыночным портфелем называют совокупность всех акций, обращающихся на рынке.

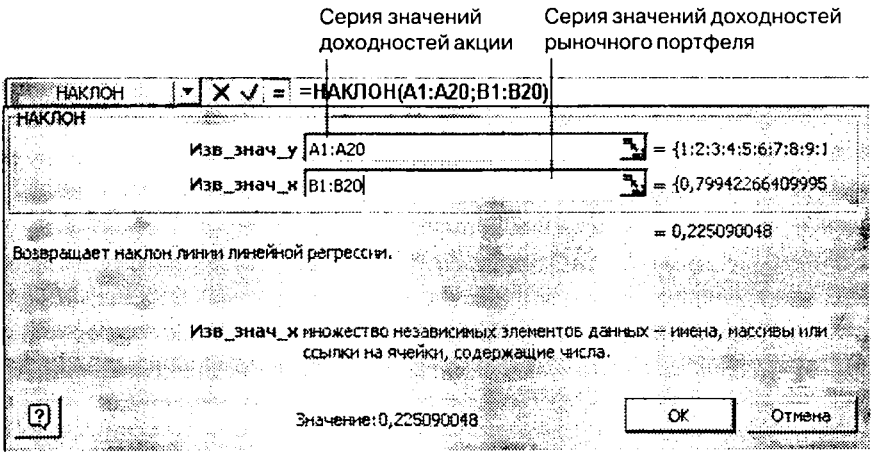


Рис. 6.18. Определение значения β-коэффициента с помощью функции НАКЛОН

График для фактических значений и построенной регрессии приведен на рис. 6.19.

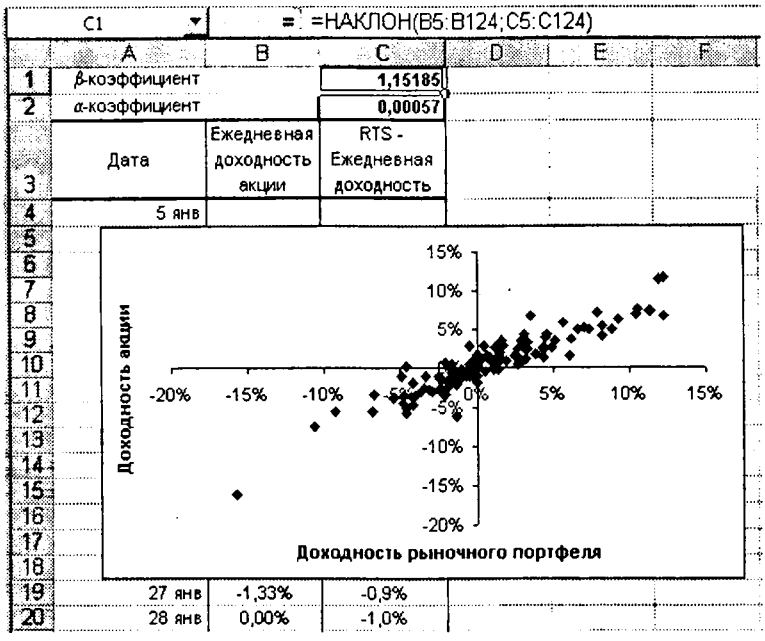


Рис. 6.19. Построение коэффициентов для акций компании «Лукойл» за период с 5.01.99 г. по 10.11.99 г. с помощью статистических функций НАКЛОН и ОТРЕЗОК

В случае, если мы хотим провести полный статистический анализ имеющихся серий значений и, в частности, проверить статистические характеристики (значимость) полученных β-коэффициентов и всего уравнения регрессии в целом, нужно воспользоваться упомянутой выше надстройкой Пакет анализа (команда меню Пакет анализа ▸ Регрессия). Соответствующие результаты приведены на рис. 6.20.

	A	B	C	D	E	F
1	ВЫВОД ИТОГОВ					
2						
3	*Регрессионная статистика*					
4	Множественный R	0,926181421				
5	R-квадрат	0,857812025				
6	Нормированный R-квадрат	0,856607042				
7	Стандартная ошибка	0,018084007				
8	Наблюдения	120				
9						
10	Дисперсионный анализ					
11		*df*				
12	Регрессия	1	0,232809441	0,232809441	711,8873352	8,25457E-52
13	Остаток	118	0,038589693	0,000327031		
14	Итого	119	0,271399134			
15						
16		*Коэффициенты*	*тандартная ошибк*	*t-статистика*	*P-Значение*	*Нижние 95%*
17	Y-пересечение	0,000568307	0,001677923	0,338696378	0,735440411	-0,002754435
18	Переменная X 1	1,151854621	0,043170993	26,6812169	8,25457E-52	1,066364395



Рис. 6.20. Построение коэффициентов для акций компании «Лукойл» за период с 5.01.99 г. по 10.11.99 г. с помощью надстройки Пакет анализа

6.2.3. Финансово-экономические расчеты

Еще одна важная сфера приложения MS Excel связана с осуществлением финансовых расчетов. Финансовые вычисления включают в себя всю совокупность методов и расчетов, используемых при принятии управленческих решений, — от элементарных арифметических операций и до сложных алгоритмов построения многокритериальных моделей, позволяющих получить оптимальные характеристики коммерческих сделок в зависимости от различных условий их проведения. Проведение подобных вычислений — трудоемкая процедура, требующая определенной математической подготовки, а также использования большого количества справочных материалов.

Считается, что как наука коммерческая арифметика начала формироваться «на заре новой истории» в Венеции, являвшейся в то время одним из крупнейших торговых центров Европы, хотя некоторые ее элементы встречаются уже в древнеегипетских манускриптах.

В силу исторически сложившегося выбора строительства централизованно планируемой социалистической экономики коммерческая арифметика в России не получила должного развития в послереволюционный период.

С переходом же к рыночным отношениям потребность в финансовых вычислениях вновь возросла. Они стали необходимы для успешного проведения любой коммерческой сделки. В комплексе с современными методами анализа и моделирования финансовых ситуаций финансовые вычисления перерастают в новое влиятельное направление организации и управления предпринимательской деятельностью — финансовый менеджмент.

Владение методами современных финансовых вычислений становится одной из основных составляющих в профессиональной подготовке предпринимателя, менеджера, банковского работника, экономиста.

На данный момент стандартный курс финансовых вычислений включает в себя следующие основные темы:

- логика финансовых операций (временная ценность денег, операции наращения и дисконтирования и т. д.);
- простые проценты (операции наращения и дисконтирования, налоги, инфляция, замена платежей); сложные проценты (то же и эквивалентность ставок, операции с валютой и т. п.);
- денежные потоки;
- анализ эффективности инвестиционных проектов;
- оценка финансовых активов.

Возрождение финансовой и страховой математики происходит в нашей стране в своеобразных условиях. С одной стороны, в мировой финансовой науке в течение XX века интенсивно развивались различные математические методы расчетов, появилась международная система унифицированных математических обозначений для стандартных финансовых и страховых схем. С другой стороны, бурное развитие индустрии ПК и их повсеместное внедрение привели к тому, что программы расчета основных финансовых показателей были реализованы на уровне, понятном широкому кругу пользователей (даже в финансовых калькуляторах!), в электронных таблицах, например в Excel.

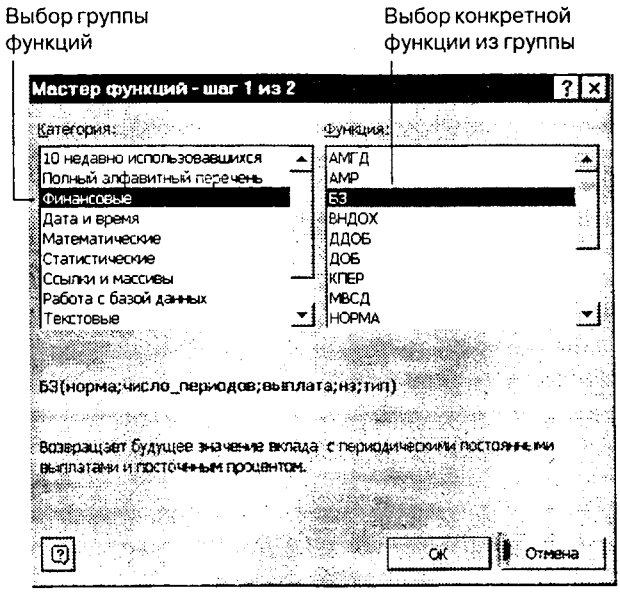


Рис. 6.21. Вызов финансовых функций

В Excel реализовано 15 встроенных и 37 дополнительных финансовых функций. В случае необходимости применения дополнительных финансовых функций необходимо установить надстройку Пакет анализа, см. 6.2.2.

Напомним, что вызов Мастера функций осуществляется либо из меню Вставка ▸ Функции — выбрать категорию Финансовые, либо с помощью одноименной кнопки на панели инструментов Стандартная. Далее в появившемся окне диалога необходимо выбрать категорию функций — и нужную функцию из категории (рис. 6.21).

По типу решаемых задач все финансовые функции Excel можно разделить на следующие условные группы:

- функции для анализа аннуитетов и инвестиционных проектов;
- функции для анализа ценных бумаг;
- функции для расчета амортизационных платежей;
- вспомогательные функции.

Функции каждой группы имеют практически одинаковый набор обязательных и дополнительных (необязательных) аргументов.

Дополнительную информацию по необходимой финансовой функции (расчетной формуле, реализованной в ней, списке аргументов и т. п.) пользователь может получить, вызвав контекстную справку (рис. 6.22).

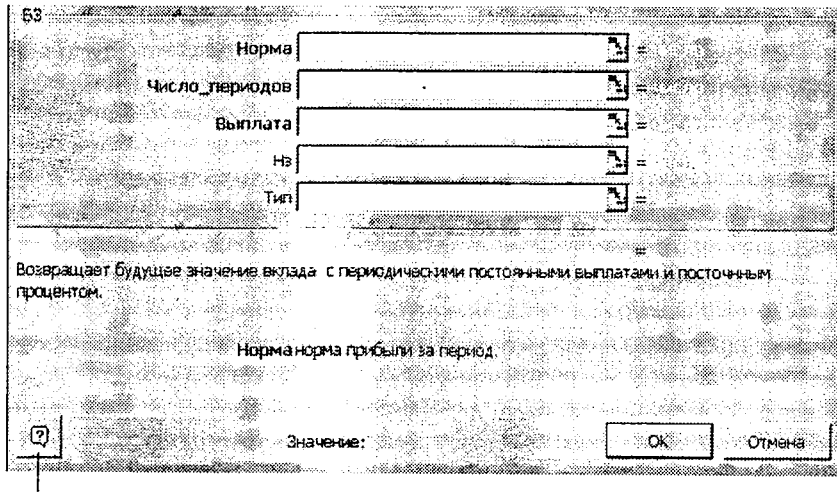


Рис. 6.22. Получение дополнительной справки по функции

6.2.4. Финансовые функции для расчетов по кредитам, займам и оценкам инвестиций

В финансовой практике часто встречаются операции, характеризующиеся возникновением потоков платежей, распределенных во времени. Потоки платежей, при которых выплаты (поступления) денежных средств осуществляются равными суммами через одинаковые интервалы времени, называются *обыкновенным*

аннуитетом. Такие потоки возникают при проведении кредитно-депозитных операций, формировании различных фондов, долгосрочной аренде и т. п.

Количественный анализ таких операций сводится к исчислению следующих основных характеристик:

- текущей величины потока платежей (Present value — Pv);
- будущей величины потока платежей (Future value — Fv);
- величины отдельного платежа (payment — R);
- нормы доходности в виде процентной ставки (interest rate – r);
- числа периодов проведения операции (число лет, месяцев).

К этой группе также относятся и функции, которые позволяют эффективно осуществить расчет наиболее широко применяемых критериев эффективности инвестиционных проектов, — NPV, IRR и т. п.

Функции этой группы используют сложные итерационные алгоримы для исчисления соответствующих показателей. При этом делаются некоторые допущения:

- потоки платежей на конец (начало) периода известны;
- для всего срока проведения операции определена оценка в виде процентной ставки, в соответствии с которой средства могут быть инвестированы.

В табл. 6.3 представлен список функций, которые могут быть использованы при решении задач анализа эффективности инвестиционных проектов. Здесь будет уместным напомнить читателю о содержании учебного примера, рассмотренного в главе 5 (о разработке документации по инвестиционному проекту). Очевидно, что приводимые ниже функции как раз и представляют собой один из инструментов проведения содержательных расчетов по подобным проектам.

Таблица 6.3. Функции для анализа аннуитетов и анализа эффективности инвестиционных проектов

Функция	Назначение функции и ее аргументы	Тип
БЗ	Позволяет определить будущую величину потока платежей при заданных исходных данных	Встроенная
	БЗ(норма; число периодов; выплата; нз; тип)	
ПЗ	Позволяет определить текущую (на момент начала операции) величину аннуитета	Встроенная
	ПЗ(норма; число периодов; выплата; бс; тип)	
КПЕР	Определяет общее число выплат (либо срок, через который начальная сумма займа достигнет заданного значения)	Встроенная
	КПЕР(норма; выплата; нз; бс; тип)	
БЗРАСПИС	Позволяет определить будущую ценность инвестиций (или единой суммы), если процентная ставка меняется во времени (по правилу сложного процента)	Дополнительная
	БЗРАСПИС(первичное; план)	

(продолжение)

Таблица 6.3 *(продолжение)*

Функция	Назначение функции и ее аргументы	Тип
НОРМА	Вычисляет процентную ставку (рентабельность операции)	Встроенная
	НОРМА(число периодов; выплата; нз; бс; тип)	
ПЛАТ	Вычисляет величину периодического платежа	Встроенная
	ПЛАТ(норма; число периодов; нз; бс; тип)	
ПЛПРОЦ	Вычисляет ту часть платежа, которая составляет его процентную часть	Встроенная
	ПЛПРОЦ(норма; период; число периодов; тс; бс)	
ОСНПЛАТ	Вычисляет ту часть платежа, которая составляет его основную часть	Встроенная
	ОСНПЛАТ(норма; период; число периодов; тс; бс)	
ОБЩПЛАТ	Вычисляет накопленные проценты (для расчетов плана погашения кредита)	Встроенная
	ОБЩПЛАТ(ставка; число периодов; нз; нач.период; кон.период)	
ОБЩДОХОД	Вычисляет накопленную сумму погашенного долга (для расчетов плана погашения кредита).	Встроенная
	ОБЩДОХОД(ставка; число периодов; нз; нач.период; кон.период)	
НПЗ	Определяет текущую (современную), приведенную к настоящему моменту времени[1] величину потока платежей	Встроенная
	НПЗ(норма; значения)	
ВНДОХ	Вычисляет внутреннюю норму рентабельности, то есть процентную ставку, при которой капитализация регулярного дохода даст сумму, равную первоначальным инвестициям. Ставку, при которой NPV=0	Встроенная
	ВНДОХ(значения; предположение)	
МВСД	Вычисляет модифицированную внутреннюю норму рентабельности (с учетом предположения о реинвестировании)	Встроенная
	МВСД(значения; финансовая норма; реинвест.норма)	
ЧИСТНЗ	Определяет текущую (современную), приведенную к настоящему моменту времени величину произвольного потока платежей, осуществляемых за любые промежутки времени, кроме этого эта функция уже учитывает величину первоначальных инвестиций	Дополнительная
	ЧИСТНЗ(ставка; значения; даты)	

Рассмотрим несколько примеров использования перечисленных функций для решения конкретных задач.

[1] Следует обратить особое внимание на то, что данная функция определяет текущую величину потока платежей по формуле $Pv = \sum_i \frac{P_i}{(1+r)^i}$, то есть не учитывает величину первоначальных инвестиций (инвестиций, сделанных в момент времени 0). Поэтому для определения значения NPV — чистого приведенного дохода (net present value) — из полученного результата следует вычесть величину первоначальных вложений.

Определение будущей стоимости

Пример 1. Фирма создает фонд для погашения долгосрочных обязательств, для чего перечисляет ежегодно в течение 4 лет платежи размером 100 тыс. р. в конце каждого года, на которые начисляются сложные проценты по ставке 18% годовых, начисляемых ежеквартально. **Определить величину фонда к концу срока выплат.**

Для решения данной задачи необходимо выполнить следующие действия:

1. Вызвать мастер функций.
2. Из общего списка выбрать функцию БЗ.
3. Заполнить диалоговое окно, то есть ввести необходимые аргументы функции (в поля ввода можно вводить как ссылки на ячейки, содержащие значения аргументов, так и сами значения), рис 6.23.
4. Завершить ввод аргументов и запуск расчета значений функции нажатием кнопки ОК.
5. При отказе работы с функцией нажать кнопку Отмена.

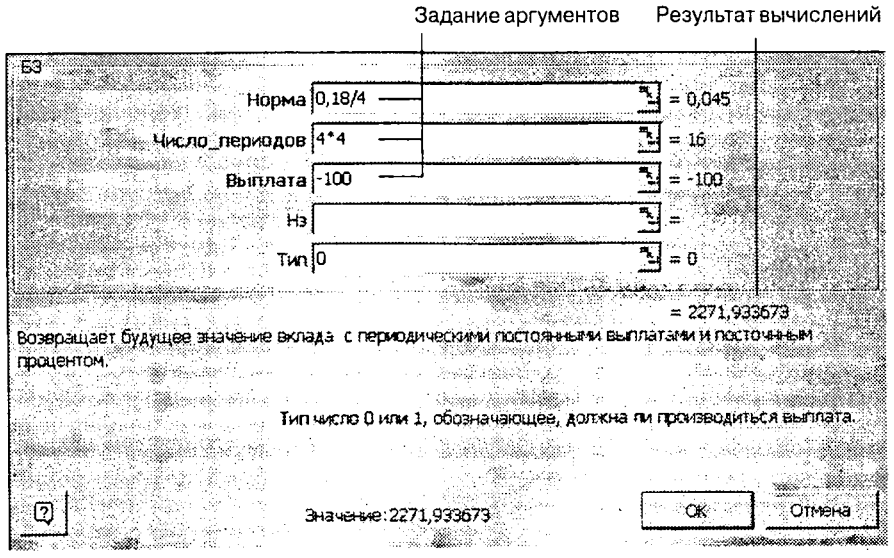


Рис. 6.23. Диалоговое окно ввода аргументов функции (БЗ)

Дадим краткие комментарии по значениям аргументов функции БЗ:

- величина периодических выплат (поле Выплата), согласно правилам заполнения, должна быть введена со знаком минус;
- по условию задачи мы соответствующим образом скорректировали поля Норма и Число периодов, так как предусматривается ежеквартальное начисление процентов;

- значение 0 у параметра Тип принимается по умолчанию и означает, что применяется *аннуитет постнумерандо* (взносы в конце года), значение –1 означает *аннуитета пренумерандо* (взносы в начале года).

Параллельно процессу заполнения полей аргументов происходит формирование результата расчетов, окончательно он равен **2271,93**. Для завершения ввода функции БЗ в ячейку рабочего листа Excel следует нажать кнопку ОК. Читателю в качестве упражнения предлагается проверить самостоятельно, что в случае аннуитета пренумерандо результат был бы равен **2374,17**.

В результате ячейка будет содержать формулу

```
=БЗ(0,18/4;4*4;-100;;1)
```

ПРИМЕЧАНИЕ

Необходимо отметить, что основные финансовые функции, такие как БЗ, ПЗ, НОРМА, КПЕР, используются не только для вычислений, имеющих дело с аннуитетом, но и для задач с разовыми вкладами на основе постоянной процентной ставки.

Пример 2. Банк выдает долгосрочный кредит в размере 186 тыс. р. по сложной ставке 19% годовых. **Определить сумму долга через 7 лет.**

Порядок действий при решении задачи тот же. Необходимо **правильно** заполнить поля ввода функции БЗ. Для решения этой задачи нужно заполнить не поле Выплата, а поле НЗ, так как в этом случае финансовая функция будет рассчитывать значение FV просто по формуле нахождения будущей стоимости для сложных процентов:

$FV = PV(1+i)^n$.

Заполним финансовую функцию БЗ следующим образом:

```
=БЗ(0,19; 7;; -186).
```

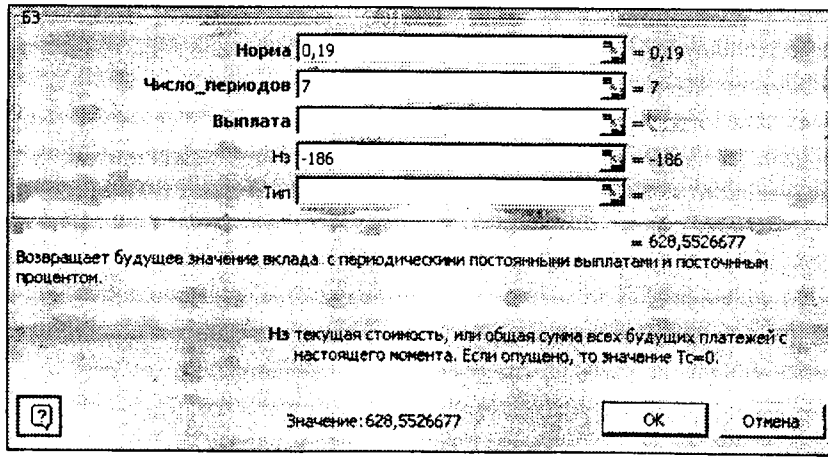


Рис. 6.24. Диалоговое окно функции БЗ

Определение текущей стоимости

Для решения задач расчета текущей стоимости используются следующие функции:

- ПЗ(норма; число периодов; ; бс) — расчет текущей стоимости для единой суммы вклада;
- ПЗ(норма; число периодов; выплата; ; тип) — расчет текущей стоимости фиксированных периодических выплат;
- НПЗ(норма; значения) — расчет текущей стоимости периодических поступлений переменной величины;
- ЧИСТНЗ(ставка; значения; даты) — расчет текущей стоимости нерегулярных поступлений переменной величины. Даты должны соответствовать суммам выплат (поступлений). Расчет производится на дату, осуществления первой операции, то есть первая сумма не дисконтируется. Если нужно произвести дисконтирование на другую дату, то необходимо первой дате в списке сопоставить выплату, равную 0.

Рассмотрим несколько простых примеров.

1. Какую сумму необходимо положить в банк, выплачивающий 13,7% годовых, чтобы через 3 года получить 250 тыс. р.?

   Для решения следует использовать функцию

   =ПЗ(норма: 0,137; число периодов: 3; ; бс: -250).

   Ответ: 170,08.

2. Платежи в фонд будут вноситься ежегодно по 200 тыс. р. в течение 4 лет с начислением на них сложных процентов по ставке 8% годовых. Определить современную сумму всех платежей с начисленными процентами.

   Для решения следует использовать функцию

   =ПЗ(норма: 0,08; число периодов: 4; выплата: -200; ; тип: 0).

   Ответ: 662,43.

3. Инвестиции в проект составляют 800 тыс. р. В последующие 5 лет ожидаются следующие годовые доходы по проекту: 250; 320; 210; 400; 150 тыс. р. Издержки привлечения капитала 7%. Рассчитать чистую текущую стоимость проекта.

   Для решения следует использовать функцию

   =НПЗ(норма: 0,07; значения: — диапазон ячеек со значениями доходов по проекту).

   Для нахождения непосредственного значения NPV необходимо вычесть начальную инвестицию.

   Ответ: 296,67 (рис. 6.25).

ПРИМЕЧАНИЕ

Положительное значение NPV является показателем того, что проект приносит чистую прибыль своим инвесторам после покрытия всех связанных с ним расходов.

D2 = =НПЗ(C2;B3:B7)-800

	A	B	C	D	E
1		Инвестиции	Издержки привлечения капитала		
2	Год 0	-800	0,07	296,67р.	
3	Год 1	250			
4	Год 2	320			
5	Год 3	210			
6	Год 4	400			
7	Год 5	150			
8					

Рис. 6.25. Нахождение NPV

4. Та же задача, но заданы конкретные даты: выплата — 1.01.99 г., поступления — 2.02.99 г., 15.03.99 г., 25.03.99 г., 10.04.99 г., 20.04.99 г. соответственно.

   Для решения следует использовать функцию ЧИСТНЗ. Порядок задания ее аргументов показан на рис. 6.26. Заметим, что значение начальной выплаты должно быть введено со знаком минус.

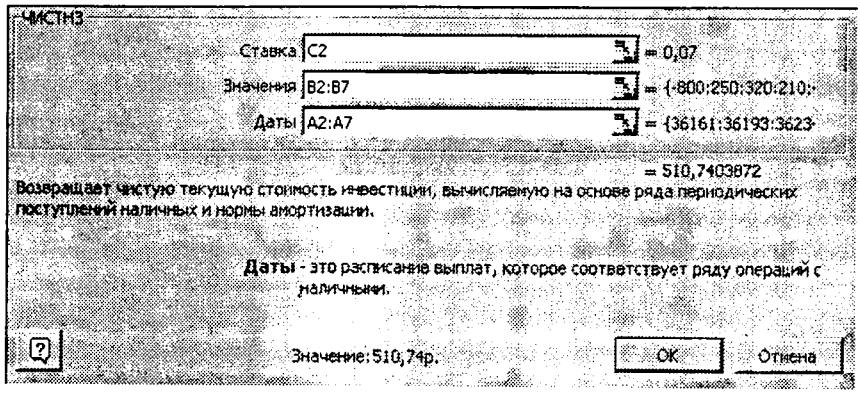


Рис. 6.26. Нахождение NPV на дату выплаты

Ответ: 510,74, результаты расчета представлены на рис. 6.27.

D2 = =ЧИСТНЗ(C2;B2:B7;A2:A7)

	A	B	C	D	E
1		Инвестиции	Издержки привлечения капитала		
2	01.01.99	-800	0,07	510,74р.	
3	02.02.99	250			
4	15.03.99	320			
5	25.03.99	210			
6	10.04.99	400			
7	20.04.99	150			
8					

Рис. 6.27. Нахождение NPV на дату первой выплаты

6.2.5. Финансовые функции для расчета амортизации

Существует несколько подходов к определению экономической категории «амортизация». Согласно одному из них, *амортизация* (depreciation) — это процесс перенесения стоимости средств труда по мере износа на производимый продукт. Как правило, процесс амортизации подразумевает дальнейшее использование перенесенной стоимости в новых циклах производства. Амортизационные фонды, формируемые предприятием (или иным экономическим субъектом), являются важнейшим источником для его развития.

Под *ускоренной амортизацией* понимается формирование амортизационных фондов в более высоком темпе, чем происходит реальный (физический) износ оборудования и других средств производства. Она может быть применена для снижения налога с прибыли, уплачиваемого предприятиями, и стимулирования, таким образом, инвестиций. На современном этапе развития российской экономики метод ускоренной амортизации становится важнейшим инструментом стимулирования малого и среднего предпринимательства.

Группа функций Excel, позволяющая автоматизировать вычисления, возникающие при составлении планов амортизации долгосрочных активов, представлена в табл. 6.4.

Таблица 6.4. Функции для расчета амортизации

Функция	Назначение функции и ее аргументы	Тип
АМР	Вычисляет амортизацию имущества за один период равномерным методом (величина амортизационных отчислений одинакова, совокупная величина отчислений к концу периода равна стоимости амортизируемого имущества)	Встроенная
	АМР(стоимость, ликвидная стоимость, время амортизации)	
АМГД	Рассчитывает амортизационные отчисления методом суммы лет. Величина начисляемой амортизации рассчитывается как отношение остатка срока эксплуатации актива к сумме лет, умноженная на разность начальная «стоимость – остаточная стоимость». Позволяет списать основную часть стоимости актива в начальные периоды их эксплуатации, когда они используются с максимальной отдачей.	Встроенная
	АМГД(стоимость, ликвидная стоимость, жизнь, период)	
ДОБ	Рассчитывает сумму амортизации для заданного периода методом уменьшающегося остатка ежегодный износ определяется с помощью заданной нормы амортизации, применяемой к чистой балансовой стоимости	Встроенная
	ДОБ(стоимость; остаточная стоимость; время эксплуатации; период; месяц)	

(продолжение)

Таблица 6.4 *(продолжение)*

Функция	Назначение функции и ее аргументы	Тип
ДДОБ	Рассчитывает сумму амортизации для заданного периода модифицированным методом уменьшающегося остатка — метод двойного списания (ускоренная норма амортизации, то есть обычная норма, используемая при равномерном списании, умноженная на некоторый коэффициент). При этом можно задать коэффициент ускоренной амортизации (по умолчанию он равен двум)	Встроенная
	ДДОБ(стоимость; остаточная стоимость; время эксплуатации; период; коэффициент)	
ПДОБ	Использует тот же алгоритм, что и функция ДДОБ, но позволяет рассчитать сумму износа за любой указанный период (границы периода и срок эксплуатации должны быть заданы в одних и тех же единицах)	Встроенная
	ПДОБ(ликвидная стоимость; остаточная стоимость; время полной амортизации; нач.период; кон.период; коэффициент; без переключения)	
	Если аргумент «без переключения» равен 0, используется метод двукратного учета амортизации с переходом на равномерный. Чтобы не было перехода на равномерный метод, установить значение1	
АМОРУВ	Предназначена для французской системы бухгалтерского учета. Использует метод равномерного списания. Если актив приобретается в середине учетного года, сумма амортизации распределяется пропорционально между всеми периодами	Дополнительная
	АМОРУВ(нач.стоимость; дата приобретения; ач.период; остаточная стоимость; период; норма; базис)	
АМОРУМ	Выполняет те же действия, что и функция АМОРУВ, но использует метод ускоренного списания. Величина коэффициента ускорения может быть 1,5; 2 или 2,5 в зависимости от срока полезной службы	Дополнительная
	АМОРУМ(нач.стоимость; дата приобретения; нач.период; остаточная стоимость; период; норма; базис)	

Рассмотрим следующий пример. Пусть необходимо рассчитать суммы амортизационных отчислений:

- равномерным методом;
- методом суммы лет;
- методом двойного списания;
- методом снижающегося остатка,

если начальная стоимость оборудования 15 000 р., остаточная стоимость 1800 р. и срок полезной службы 5 лет.

Для решения данной задачи используются функции АМП, АМГД, ДДОБ, ДОБ. Результат их применения представлен на рис. 6.28.

C6 = =АМГД(B1;B2;B3;A6)

	A	B	C	D	E
1	Начальная стоимость	15000			
2	Остаточная стоимость	1800			
3	Срок полезной службы	5			
4					
5	Период	Равномерное списание (АМП)	Метод суммы лет (АМГД)	Метод двойного списания (ДДОБ)	Метод снижающегося остатка (ДОБ)
6	1	2 640,00р.	4 400,00р.	6 000,00р.	5 190,00р.
7	2	2 640,00р.	3 520,00р.	3 600,00р.	3 394,26р.
8	3	2 640,00р.	2 640,00р.	2 160,00р.	2 219,85р.
9	4	2 640,00р.	1 760,00р.	1 296,00р.	1 451,78р.
10	5	2 640,00р.	880,00р.	144,00р.	949,46р.
11					
12	**Всего**	**13 200,00р.**	**13 200,00р.**	**13 200,00р.**	**13 205,35р.**

Рис. 6.28. Расчет амортизации

6.2.6. Финансовые функции для расчетов по ценным бумагам

Понятие *«ценная бумага»* в общем случае является многозначным термином. В узком смысле под ценной бумагой может пониматься документ, удостоверяющий некоторые права ее владельца по отношению к эмитенту (субъекту, выпустившему данную бумагу).

Среди основных видов ценных бумаг (активов), обращающихся на современном рынке, можно выделить:

- *облигации* — письменные долговые обязательства эмитента выплатить полученную сумму с процентами в определенный срок;
- *акции* — ценные бумаги, удостоверяющие долевое участие владельца в капитале деловой единицы бизнеса, имеющей статус юридического лица в форме корпорации;
- *деривативы* (производные ценные бумаги) — некоторые права, связанные с уже существующими реальными активами (чаще всего ценными бумагами). Основным видом деривативов являются срочные контракты — фьючерсы, форварды и опционы. Все они представляют собой соглашения двух сторон о будущей поставке некоторого актива, все условия которой оговариваются заранее, при заключении контракта. Также к деривативам относят различные инструменты хеджирования процентных ставок: соглашения о будущей процентной ставке (FRA), процентные фьючерсы, опционы и их комбинации, а также варранты. Производные ценные бумаги, как правило, являются еще более рискованными, чем обыкновенные акции.

MS Excel предоставляет пользователям широкие возможности по проведению расчетов, связанных со стандартными операциями, проводимыми с ценными бумагами. В табл. 6.5 приводится их список.

Таблица 6.5. Финансовые функции для расчетов по ценным бумагам

Функция	Назначение функции и ее аргументы	Тип
ДНЕЙКУПОН	Вычисляет число дней в период купона	Добавочная
	ДНЕЙКУПОН(дата соглашения; дата вступления в силу; периодичность; базис)[1]	
ЧИСЛКУПОН	Рассчитывает количество купонов, которые могут быть оплачены между датой покупки и датой погашения ценной бумаги, округленное до ближайшего целого числа	Добавочная
	ЧИСЛКУПОН(дата соглашения; дата вступления в силу; периодичность; базис)	
ДАТАКУПОНДО	Вычисляет последнюю дату выплаты купона, предшествующую покупке ценной бумаги	Добавочная
	ДАТАКУПОНДО(дата соглашения; дата вступления в силу; периодичность; базис)	
ДНЕЙКУПОНДО	Вычисляет количество дней, прошедших от момента купонного платежа до даты покупки ценной бумаги	Добавочная
	ДНЕЙКУПОНДО(дата соглашения; дата вступления в силу; периодичность; базис)	
ДАТАКУПОНПОСЛЕ	Определяет дату оплаты купона, следующую за датой приобретения ценной бумаги	Добавочная
	ДАТАКУПОНПОСЛЕ(дата соглашения; дата вступления в силу; периодичность; базис)	
ДНЕЙКУПОНПОСЛЕ	Определяет число дней от момента приобретения ценной бумаги до оплаты очередного купона	Добавочная
	ДНЕЙКУПОНПОСЛ(дата соглашения; дата вступления в силу; периодичность; базис)	
ДОХОД	Определяет годовую ставку помещения по операциям с ценными бумагами при заданной купонной ставке и разности курсов покупки и погашения за указанный период	Добавочная
	ДОХОД(дата соглашения; дата вступления в силу; ставка; цена; погашение; частота; базис)	
ЦЕНА	Определяет цену покупки (курс) ценной бумаги с периодическими выплатами купонных процентов	Добавочная
	ЦЕНА(дата соглашения; дата вступления в силу; ставка; доход; погашение; частота; базис)	
НАКОПДОХОД	Определяет накопленный на момент приобретения ценной бумаги купонный доход	Добавочная
	НАКОПДОХОД(дата выпуска; дата первой выплаты; дата соглашения; ставка; номинал; периодичность; базис)	
ДОХОДПЕРВНЕРЕГ	Определяет доход (ставку помещения) по ценным бумагам с нерегулярным первым периодом выплаты купона	Добавочная
	ДОХОДПЕРВНЕРЕГ(дата соглашения; дата вступления в силу; дата выпуска; первый купон; ставка; цена; погашение; частота; базис)	

[1] Поле **базис** в версии MS Excel 97 (и последующих) не используется.

Функция	Назначение функции и ее аргументы	Тип
ДОХОДПОСЛНЕРЕГ	Определяет доход по ценным бумагам с нерегулярным последним периодом выплаты	Добавочная
	ДОХОДПОСЛНЕРЕГ(дата соглашения; дата вступления в силу; последняя выплата; ставка; цена; погашение; частота; базис)	
ЦЕНАПЕРВНЕРЕГ	Определяет курс покупки ценных бумаг для нерегулярного первого периода	Добавочная
	ЦЕНАПЕРВНЕРЕГ(дата соглашения; дата вступления в силу; дата выпуска; первый купон; ставка; доход; погашение; частота; базис)	
ЦЕНАПОСЛНЕРЕГ	Определяет курс покупки ценных бумаг для нерегулярного последнего периода	Добавочная
	ЦЕНАПОСЛНЕРЕГ(дата соглашения; дата вступления в силу; последняя выплата; ставка; доход; погашение; частота; базис)	
ДОХОДПОГАШ	Определяет годовой доход (ставка помещения) по ценным бумагам с выплатой процентов и номинала в момент погашения (для облигаций)	Добавочная
	ДОХОДПОГАШ(дата соглашения; дата вступления в силу; дата выпуска; ставка; цена; базис)	
ЦЕНАПОГАШ	Определяет курс покупки ценных бумаг по которым купонный доход выплачивается в срок погашения одновременно с выкупом	Добавочная
	ЦЕНАПОГАШ(дата соглашения; дата вступления в силу; дата выпуска; ставка; доход; базис)	
НАКОПДОХОДПОГАШ	Определяет сумму накопленного купонного дохода по ценным бумагам за весь период их действия	Добавочная
	НАКОПДОХОДПОГАШ(дата выпуска; дата вступления в силу; ставка; номинал; базис)	
ИНОРМА	Определяет годовую ставку дополнительного дохода для ценных бумаг без периодической выплаты процентов согласно формулам расчета простой процентной ставки	Добавочная
	ИНОРМА(дата соглашения; дата вступления в силу; инвестиция; погашение; базис)	
ПОЛУЧЕНО	Определяет наращенную сумму, получаемую в срок вступления в силу ценных бумаг при использовании учетной (дисконтной ставки)	Добавочная
	ПОЛУЧЕНО(дата соглашения; дата вступления в силу; инвестиция; скидка; базис)	
ДОХОДСКИДКА	Определяет ставку годового дохода по ценным бумагам, периодические выплаты процентов по которым не предусмотрены и на которые сделана скидка	Добавочная
	ДОХОДСКИДКА(дата соглашения; дата вступления в силу; цена; погашение; базис)	

(продолжение)

Таблица 6.5. *(продолжение)*

Функция	Назначение функции и ее аргументы	Тип
СКИДКА	Определяет учетную ставку (скидку) для ценных бумаг, по которым не предусмотрены периодические выплаты	Добавочная
	СКИДКА(дата соглашения; дата вступления в силу; цена; выкуп; базис)	
ЦЕНАСКИДКА	Определяет цену покупки ценной бумаги, на которую сделана скида с цены погашения	Добавочная
	ЦЕНАСКИДКА(дата соглашения; дата вступления в силу; скидка; погашение; базис)	
ДОХОДКЧЕК	Определяет ставку годового дохода по краткосрочным ценным бумагам (срок действия не более года, периодических выплат нет) по простому проценту	Добавочная
	ДОХОДКЧЕК(дата соглашения; дата вступления в силу; цена)	
РАВНОКЧЕК	Определяет ставку годового дохода по краткосрочным ценным бумагам, эквивалентную ставке годового дохода (соотношение эквивалентности между ставкой наращения и учетной ставкой)	Добавочная
	РАВНОКЧЕК(дата соглашения; дата вступления в силу; скидка)	
ЦЕНАКЧЕК	Определяет цену покупки (курс) по краткосрочным ценным бумагам без периодических выплат с установленной скидкой к цене погашения	Добавочная
	ЦЕНАКЧЕК(дата соглашения; дата вступления в силу; скидка)	
ДЛИТ	Определяет срок действия ценных бумаг с периодическими выплатами процентов как среднее взвешенное текущих купонных выплат и номинала	Добавочная
	ДЛИТ(дата соглашения; дата вступления в силу; купон; доход; периодичность; базис)	
МДЛИТ	Определяет модифицированную длительность для ценных бумаг с предполагаемой нарицательной стоимостью в 100 единиц. Если купонных выплат нет — определяет срок действия облигации как длительность календарного периода от даты соглашения до даты погашения	Добавочная
	МДЛИТ(дата соглашения; дата вступления в силу; купон; доход; частота; базис)	

Рассмотрим следующий пример, а именно, определим учетную ставку для ценной бумаги, приобретенной 01.01.98 г. по курсу 82,14 со сроком погашения 15.11.98 г.

Для решения этой задачи следует воспользоваться функцией СКИДКА. На рис. 6.29 показан процесс заполнения ее полей.

Результат расчетов приведен на рис. 6.30.

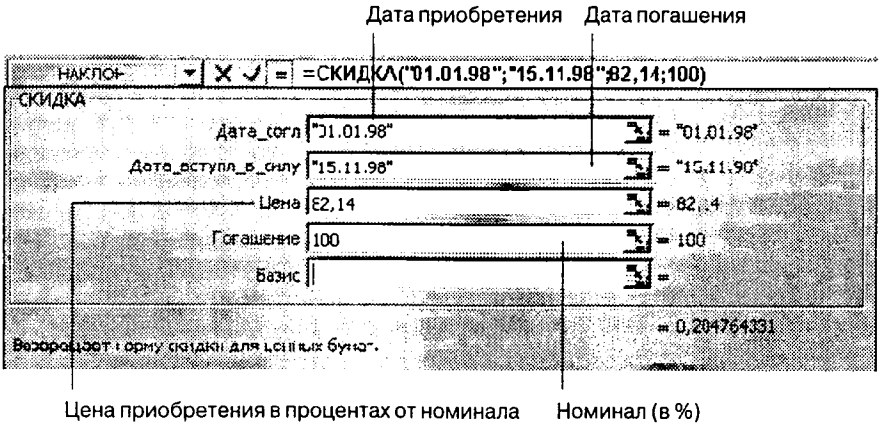


Рис. 6.29. Заполнение полей функции СКИДКА

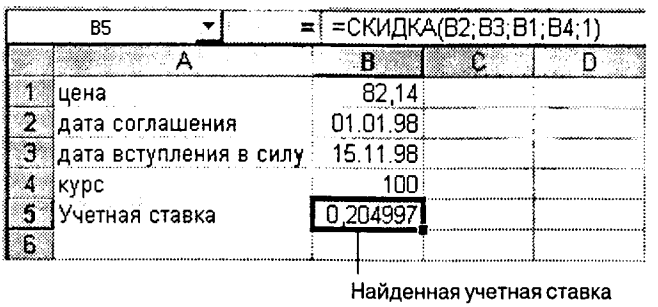


Рис. 6.30. Расчет учетной ставки

6.2.7. Решение уравнений и оптимизационных задач

Широкое применение на практике находят функции и режимы Excel, предназначенные для поиска решения уравнений и оптимизационных задач.

Подбор параметра

Удобным и простым для понимания инструментом решения уравнений является режим Подбор параметра. Он реализует алгоритм численного решения уравнения, зависящего от одной или нескольких переменных.

Процесс решения с помощью данного метода распадается на два этапа:

1. Задание на рабочем листе ячеек, содержащих переменные решаемого уравнения (так называемых *влияющих* ячеек), и ячейки, содержащей формулу уравнения (*зависимой* или *целевой* ячейки).
2. Ввод адресов влияющих и целевой ячеек в диалоговое окно Подбор параметра и получение ответа (или сообщения о его отсутствии/невозможности найти)[1].

[1] При работе с любыми программными средствами никогда не следует забывать и о математической стороне проблемы: во-первых, уравнение может просто не иметь решений, а во-вторых, численный алгоритм, лежащий в основе программной надстройки, может в тех или иных условиях оказаться расходящимся.

Рассмотрим теперь применение режима Подбор параметра на ряде конкретных примеров.

Пример 1. Найти решение уравнения $3x^2-2y^2=5$.

Результат создания влияющих и целевых ячеек показан на рис. 6.31, а ввод параметров в окно диалога, появляющегося при выполнении команды меню Сервис ▸ Подбор параметра..., — на рис. 6.32.



Рис. 6.31. Заполнение ячеек рабочего листа перед вызовом режима Подбор параметра

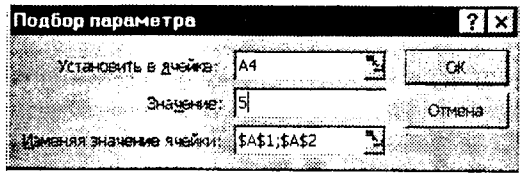


Рис. 6.32. Ввод значений в окно Подбор параметра

Наконец, на рис. 6.33 показан результат выполнения процедуры подбора параметра.

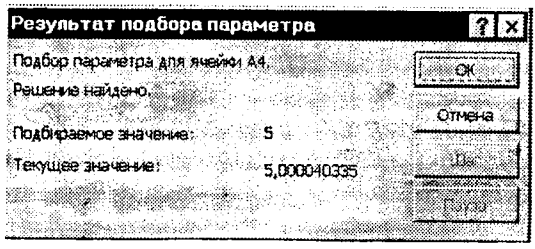


Рис. 6.33. Результирующее окно режима подбора параметра

По выводимому в результирующее окно (рис. 6.33) Текущему значению можно судить о степени точности найденного результата. Если нажать кнопу ОК, то

содержимое влияющих ячеек (в нашем случае это ячейки A1 и A2) будет заменено на решения уравнения. Дополнительно отметим, что поскольку рассматриваемое уравнение $3x^2-2y^2=5$ зависит от двух переменных и имеет бесконечное множество решений, то какие числа мы получим в ячейках A1 и A2, непосредственно зависит от их исходного содержания (так называемого начального приближения).

Пример 2. Определить, при какой ежемесячной процентной ставке можно за год накопить 5 тыс. р., внося каждый месяц платеж на 10% больше предыдущего, начав с первого платежа 100 р.

С помощью одной лишь финансовой функции эту задачу решить нельзя, в ней слишком много неизвестных. Для начала необходимо смоделировать реальный поток платежей, затем найти накопленную к концу года сумму, нарастив каждый платеж по предполагаемой ставке (так как в задаче дана ежемесячная ставка) на соответствующее число процентных периодов (первый платеж на 12 месяцев вперед, второй — на 11 и т. п., см. рис. 6. 35), и только после этого с помощью Подбора параметра найти истинное значение процентной ставки. Ответ: 14,72%, см. рис. 6.36.

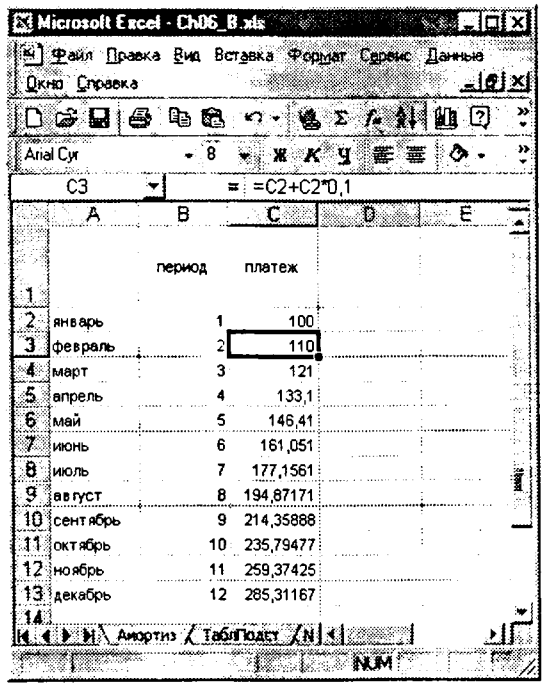
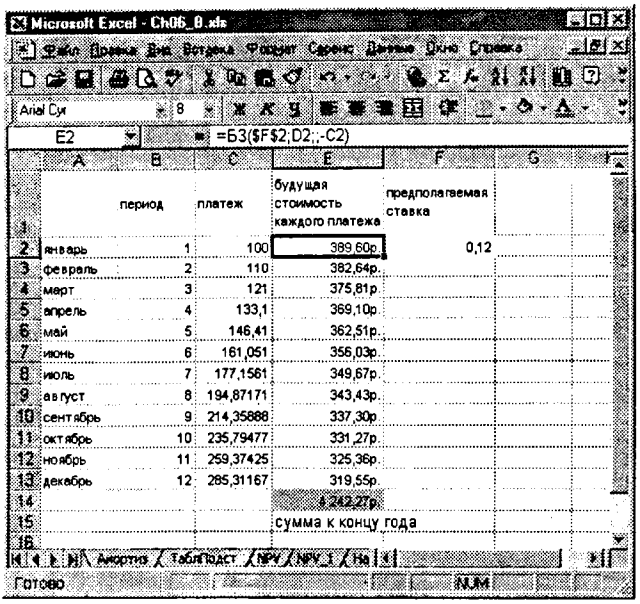


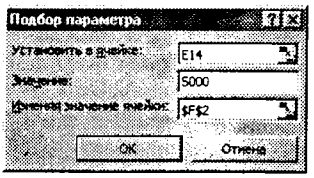
Рис. 6.34. Моделирование потока платежей

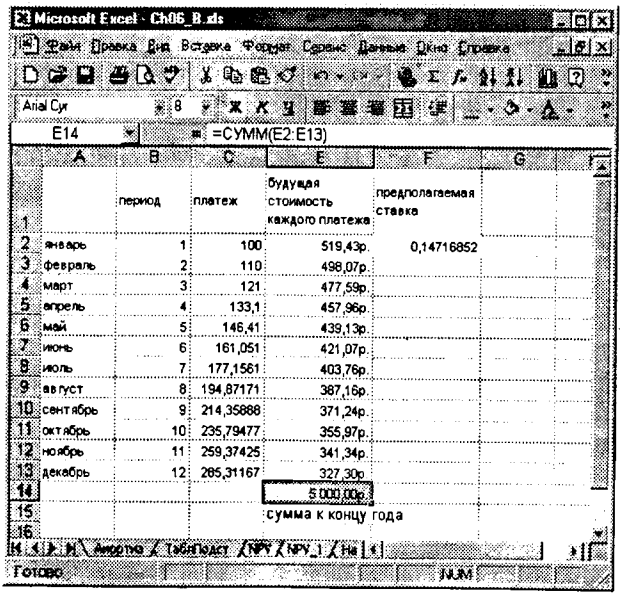
Очевидно, что «платой» за простоту такого инструмента, как Подбор параметра, является ограниченность его возможностей. Еще раз подчеркнем, что с его помощью могут быть решены только отдельно взятые уравнения.



	A	B	C	E	F	G
1		период	платеж	будущая стоимость каждого платежа	предполагаемая ставка	
2	январь	1	100	389,60р.	0,12	
3	февраль	2	110	382,64р.		
4	март	3	121	375,81р.		
5	апрель	4	133,1	369,10р.		
6	май	5	146,41	362,51р.		
7	июнь	6	161,051	356,03р.		
8	июль	7	177,1561	349,67р.		
9	август	8	194,87171	343,43р.		
10	сентябрь	9	214,35888	337,30р.		
11	октябрь	10	235,79477	331,27р.		
12	ноябрь	11	259,37425	325,36р.		
13	декабрь	12	285,31167	319,55р.		
14				4 242,27р.		
15				сумма к концу года		
16						

Рис. 6.35. Расчет накопленной к концу года суммы по предполагаемой ставке





	A	B	C	E	F	G
1		период	платеж	будущая стоимость каждого платежа	предполагаемая ставка	
2	январь	1	100	519,43р.	0,14716852	
3	февраль	2	110	498,07р.		
4	март	3	121	477,59р.		
5	апрель	4	133,1	457,96р.		
6	май	5	146,41	439,13р.		
7	июнь	6	161,051	421,07р.		
8	июль	7	177,1561	403,76р.		
9	август	8	194,87171	387,16р.		
10	сентябрь	9	214,35888	371,24р.		
11	октябрь	10	235,79477	355,97р.		
12	ноябрь	11	259,37425	341,34р.		
13	декабрь	12	285,31167	327,30р.		
14				5 000,00р.		
15				сумма к концу года		
16						

Рис. 6.36. Подбор параметра

Применение надстройки Поиск решения

Значительно более мощным по сравнению с Подбором параметра средством решения уравнения, а также достаточно эффективным инструментом решения оптимизационных задач является программная надстройка Поиск решения. Напомним, что вопросы установки надстроек были рассмотрены ранее в 6.2.2 в связи с надстройкой Пакет анализа. В случае успешной установки программной надстройки Поиск решения в меню Сервис появляется дополнительный пункт Поиск решения.... Рассмотрим процесс ее использования на примере простейшей *задачи об управлении портфелем активов*[1].

Пусть перед некоторым инвестором стоит проблема принятия решения о вложении имеющегося у него капитала. Набор характеристик потенциальных объектов для инвестирования, имеющих условные имена от A до F, задается в табл. 6.6.

Предположим, что при принятии решения о приобретении активов должны быть соблюдены условия:

1. Суммарный объем капитала, который должен быть вложен, составляет $100 000.
2. Доля средств, вложенная в один объект, не может превышать четверти от всего объема.

Таблица 6.6. Параметры условных активов (для учебного примера)

Название	Доходность, в %	Срок выкупа, год	Надежность, баллы
A	5,5	2001	5
B	6,0	2005	4
C	8,0	2010	2
D	7,5	2002	3
E	5,5	2000	5
F	7,0	2003	4

3. Более половины всех средств должны быть вложены в долгосрочные активы (допустим, на рассматриваемый момент к таковым относятся активы со сроком погашения после 2004 года).
4. Доля активов, имеющих надежность менее чем 4 балла, не может превышать трети от суммарного объема.

Дадим теперь описание экономико-математической модели для данной ситуации. В рассматриваемом примере в качестве управляемых переменных выступают объемы средств, вложенных в активы той или иной фирмы. Обозначим их как x_A, x_B, x_C, x_D, x_E, x_F. Тогда суммарная прибыль от размещенных активов, которую получит инвестор, может быть представлена в виде

[1] Пример заимствован из книги: Конюховский П. В. Математические методы исследования операций в экономике. СПб.: Питер, 2000.

$P=0{,}055x_A+0{,}06x_B+0{,}08x_C+0{,}075x_D+0{,}055x_E+0{,}07x_F$.

На следующем этапе моделирования мы должны формально описать перечисленные выше ограничения 1 — 4 на структуру портфеля.

1. Ограничение на суммарный объем активов:

$x_A+x_B+x_C+x_D+x_E+x_F \le 100\,000$.

2. Ограничение на размер доли каждого актива:

$x_A \le 25\,000$, $x_B \le 25\,000$, $x_C \le 25\,000$
$x_D \le 25\,000$, $x_E \le 25\,000$, $x_F \le 25\,000$.

3. Ограничение, связанное с необходимостью вкладывать половину средств в долгосрочные активы:

$x_B+x_C \ge 50\,000$.

4. Ограничение на долю ненадежных активов:

$x_C+x_D \ge 50\,000$.

Наконец, система ограничений в соответствии с экономическим смыслом задачи должна быть дополнена условиями неотрицательности для искомых переменных:

$x_A \ge 0$, $x_B \ge 0$, $x_C \ge 0$, $x_D \ge 0$, $x_E \ge 0$, $x_F \ge 0$,

Перечисленные условия образуют математическую модель поведения инвестора. В рамках этой модели может быть поставлена задача поиска таких значений переменных x_A, x_B, x_C, x_D, x_E, x_F, при которых достигается наибольшее значение прибыли и одновременно выполняются ограничения на структуру портфеля активов.

Перейдем к решению сформулированной задачи с помощью инструментов, предоставляемых программным обеспечением MS Excel. Оно распадается на следующие шаги:

1. На выбранном рабочем листе задать ячейки, которые будут предназначены для сохранения переменных решаемой задачи (x_A, x_B, x_C, x_D, x_E, x_F), как это сделано на рис. 6.37, где переменная x_A содержится в ячейке A2, x_B — в B2 и т. д. Заметим, что хорошим стилем работы является использование смежных ячеек для хранения имен переменных (на рис. 6.37 для этого служат ячейки с A1 по F1). Очевидно, что ни в коем случае не следует путать ячейки с переменными с ячейками со смысловыми именами. Увы, у начинающих пользователей это весьма распространенная ошибка.
2. Задать ячейку, содержащую формулу целевой функции решаемой задачи. В нашем примере ячейка H2 содержит формулу

```
=0,055*A2+0,06*B2+0,08*C2+0,075*D2+0,055*E2+0,07*F2.
```

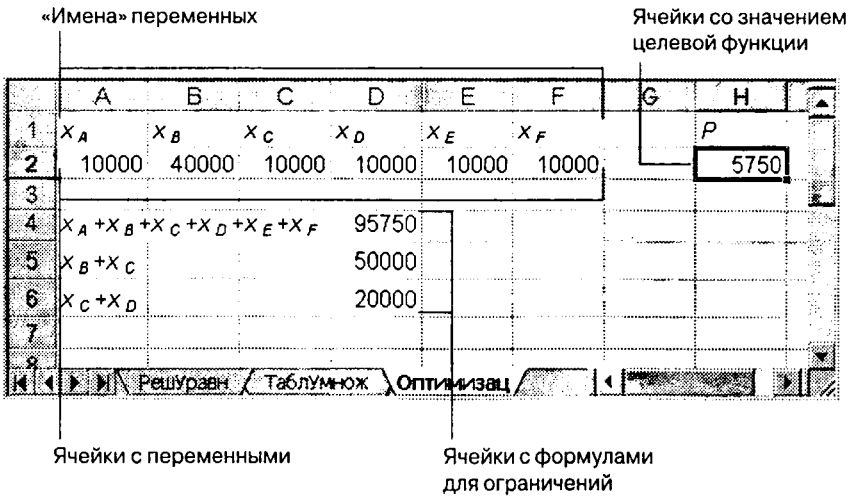


Рис. 6.37. Заполнение листа Excel перед вызовом Поиска решения

3. Заполнить ячейки для формул «сложных» ограничений (типа $x_B+x_C \geq 50\,000$ и т. п.). На рис. 6.37 для этого использованы ячейки D4, D5, D6, а ячейки A4, A5, A6 содержат соответствующие подписи.

ПРИМЕЧАНИЕ

Интерфейс надстройки Поиск решения устроен таким образом, что непосредственно в режиме диалога могут вводиться только ограничения типа A2 <= 25 000 или A2 >= 0. Поэтому для учета условия вида $x_B+x_C \geq 50000$ нужно в некоторую вспомогательную ячейку ввести формулу с левой частью неравенства и уже для нее задать ограничение >= 50 000.

4. Выполнить команду меню Сервис ▸ Поиск решения....
5. Заполнить параметры диалогового окна Поиск решения: адрес ячейки целевой функции (целевой ячейки), тип оптимизации (искать максимум или минимум), адреса ячеек с переменными (рис. 6.38).
6. Задать систему ограничений, для чего используется кнопка Добавить (группа Ограничения). По ее нажатию вызывается вспомогательное диалоговое окно, показанное на рис. 6.39, в поля которого вводятся адреса или значения, образующие выражение для условия, накладываемого на переменные решаемой задачи. Как видно из рис. 6.38, сформированную систему ограничений в дальнейшем можно редактировать. Для этого служат кнопки Изменить и Удалить.
7. Нажать кнопку Выполнить, после чего будет осуществлена процедура поиска решения, по результатам которой выводится сообщение о найденном решении (или о невозможности его обнаружить), см. рис. 6.40. Как видно из данного рисунка, полученные результаты можно сохранить (кнопка OK), изменив,

таким образом, содержимое ячеек с переменными, можно от них отказаться (не сохранять), наконец, можно сформировать отчет с более подробной информацией о том, как проходил процесс поиска решения[1].

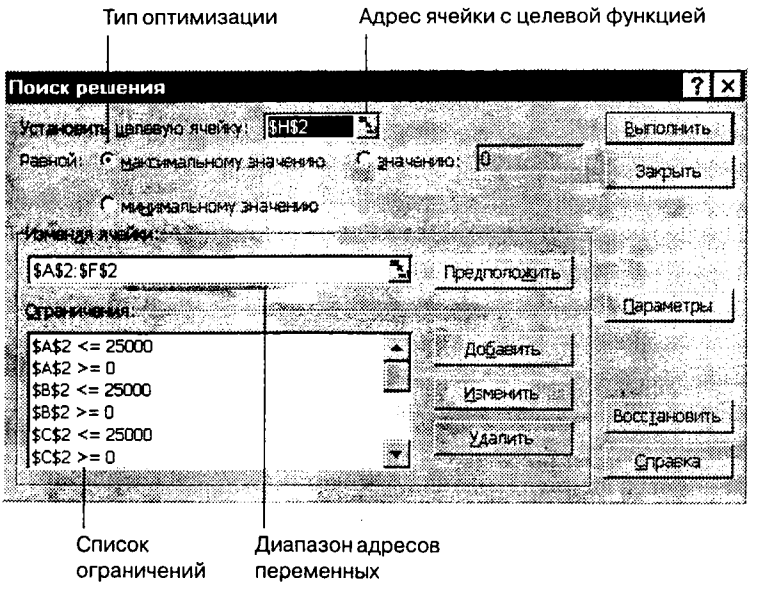


Рис. 6.38. Диалоговое окно надстройки Поиск решения

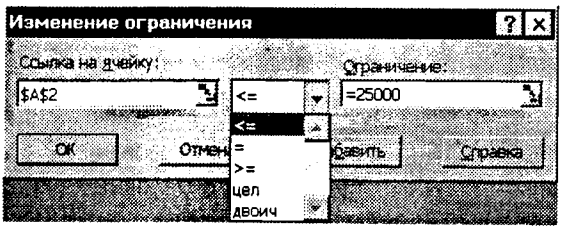


Рис. 6.39. Окно задания ограничений

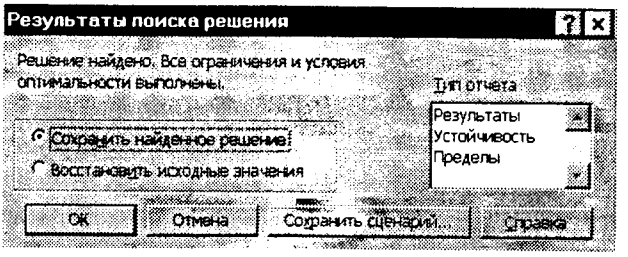


Рис. 6.40. Окно с результатами поиска решения

[1] О возможности использования команды Сохранить сценарий заинтересованный читатель может прочесть в подробных руководствах для пользователей MS Excel.

Результаты, полученные для рассматриваемой задачи, показаны на рис. 6.41. Таким образом, мы получили, что при оптимальном распределении прибыль инвестора составит $6374,4.

ПРИМЕЧАНИЕ

Как уже отмечалось ранее, сходимость (нахождение решения за конечное число шагов) в численных алгоритмах оптимизации не гарантируется. Однако в случае, если решение не найдено, пользователь, обладающий некоторой математической подготовкой, может оказать определенное влияние на процесс поиска, изменив значения его параметров, которые установлены по умолчанию (см. рис. 6.38, кнопка Параметры). В некоторых случаях такие вариации могут привести к положительному результату.

Другой полезный совет, который можно дать в случае, когда решение не удается найти сразу, касается изменения начальных значений переменных. Последнее оказывается полезным и с точки зрения анализа найденного решения на устойчивость (независимость от того, с какой точки начинается его поиск).

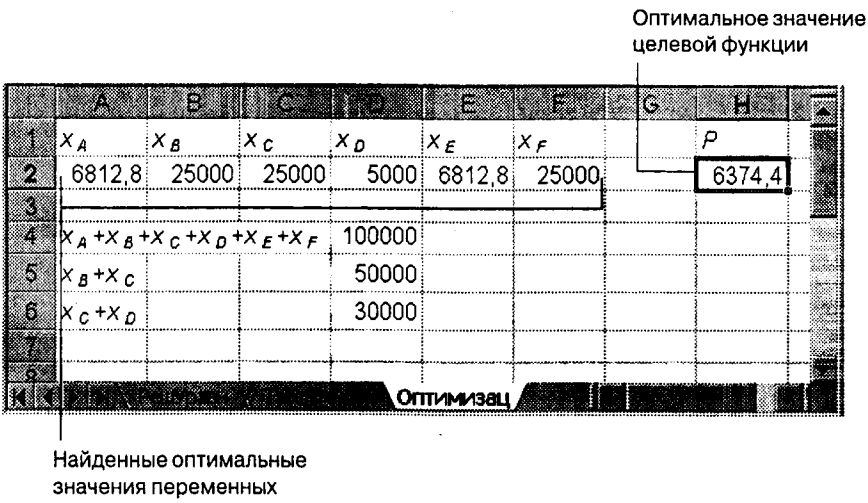


Рис. 6.41. Вид рабочего листа с найденным решением

Пример расчетов с использованием Таблиц подстановки

Зачастую при проведении финансово-экономических расчетов возникает необходимость провести вычисления по одним и тем же формулам, но для различных серий данных. Конечно, с данной проблемой можно справиться с помощью простого копирования формул, однако в Excel предусмотрен и более удобный способ ее решения с помощью так называемой *таблицы подстановки*. Ее идея состоит в связывании некоторой формулы с сериями значений, которые должны быть подставлены вместо некоторых переменных, входящих в данную формулу.

Таблицы подстановки в Excel могут содержать одну или две подстановочных переменных, или, другими словами, быть векторными или матричными.

Применение таблицы подстановки с двумя переменными продемонстрируем на простом, но наглядном примере — для построения таблицы умножения.



Рис. 6.42. Ввод расчетной формулы и выделение диапазона для таблицы подстановки

Необходимо проделать следующие операции:

1. Определить две ячейки, содержащие переменные. В нашем случае (рис. 6.42) это ячейки A1 и A2. Начальное содержимое данных ячеек может быть произвольным, так как они нужны для того, чтобы **определить переменные**, от которых будет зависеть целевая формула.
2. Задать в «матричной» форме, как это показано на рис. 6.42, целевую формулу, зависящую от ячеек, определенных на этапе 1 (для нашего примера — это формула =A1*A2), а также серии значений, предназначенных для подстановки вместо переменных. Серии значений должны располагаться в левой колонке и верхней строке. При определении таблицы необходимо соблюдать очевидное правило — ячейки с переменными не должны попасть в ее внутреннюю область.
3. Выделить область таблицы, как это показано на рис. 6.42.
4. Выполнить команду меню Данные ▸ Таблица подстановки....
5. Заполнить параметры в появившемся диалоговом окне (рис. 6.43). Первое значение — Подставлять значения по столбцам в — задает адрес ячейки с той переменной, вместо которой в целевую формулу будут подставляться значения из крайней верхней строки таблицы подстановки. В нашем случае вместо переменной из ячейки A1 последовательно будут подставлены в формулу значения из интервала C4:L4. Аналогично, второе значение — Подставлять значения по строкам в — задает адрес ячейки той переменной, вместо которой в целевую формулу будут подставляться значения из крайнего левого столбца таблицы

подстановки. В нашем примере вместо переменной из ячейки A2 в формулу будут подставлены значения из интервала B5:B14.

6. Нажать кнопку OK.

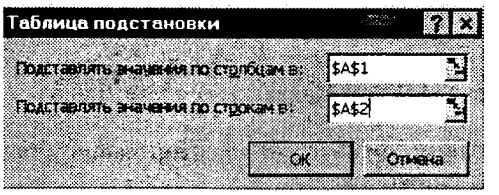
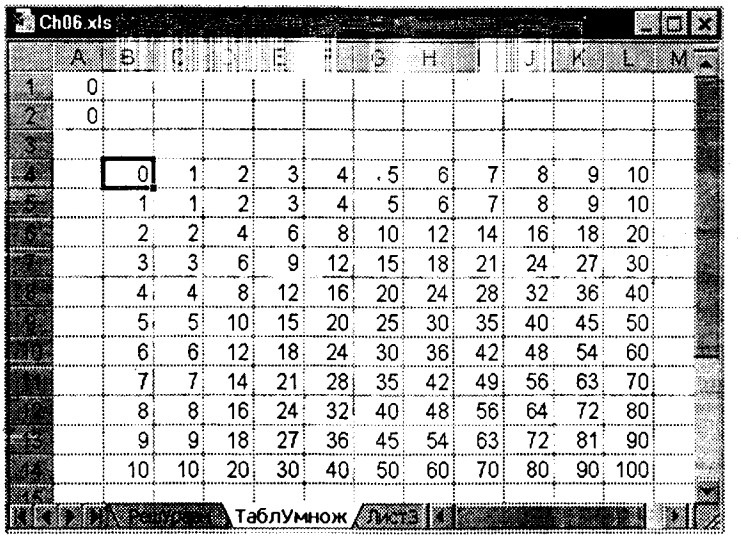


Рис. 6.43. Задание соответствия между переменными и сериями значений в столбцах и строках таблицы подстановки

Результат заполнения таблицы подстановки показан на рис. 6.44.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	0												
2	0												
3													
4		0	1	2	3	4	5	6	7	8	9	10	
5		1	1	2	3	4	5	6	7	8	9	10	
6		2	2	4	6	8	10	12	14	16	18	20	
7		3	3	6	9	12	15	18	21	24	27	30	
8		4	4	8	12	16	20	24	28	32	36	40	
9		5	5	10	15	20	25	30	35	40	45	50	
10		6	6	12	18	24	30	36	42	48	54	60	
11		7	7	14	21	28	35	42	49	56	63	70	
12		8	8	16	24	32	40	48	56	64	72	80	
13		9	9	18	27	36	45	54	63	72	81	90	
14		10	10	20	30	40	50	60	70	80	90	100	

Рис. 6.44. Результат заполнения таблицы подстановки

Ключевые понятия

- ячейка;
- рабочий лист;
- типы данных;
- формула;
- функция;
- диаграмма;
- надстройка;

- Пакет анализа;
- Подбор параметра;
- Поиск решения;
- таблица подстановки.

Контрольные вопросы

1. Перечислите известные вам способы редактирования и форматирования данных в ячейках рабочей книги.
2. Какие особенности имеет процесс ввода формул в ячейки Excel (по сравнению с вводом данных других типов)?
3. В каких случаях следует использовать абсолютную, а в каких — относительную адресацию?
4. Опишите процесс ввода встроенных функций.
5. Опишите процесс построения диаграммы в Excel.
6. Перечислите известные вам инструменты статистического анализа данных в Excel.
7. Что такое автофильтр?
8. Что такое сводная таблица?
9. Назовите как минимум два способа нахождения коэффициентов в уравнении регрессии.
10. Перечислите основные классы финансовых функции Excel и дайте краткую характеристику их назначения.
11. Какую функцию следует применить для расчета будущей стоимости?
12. Какая функций используется для расчета амортизационных исчислений методом суммы лет?
13. Для решения каких задач используется режим Подбор параметра?
14. Охарактеризуйте основные направления применения надстройки Поиск решения.
15. Для чего служит таблица подстановки?

Литература

1. Microsoft Office для Windows 95: 6 книг в одной/Под ред. В. Кошелева. М.: БИНОМ, 1997.
2. Белых Л. П. Основы финансового рынка. М.: ЮНИТИ, 1999.
3. Замков О. О., Толстопятенко А.В., Черемных Ю. Н. Математические методы в экономике. М.: ДИС, 1998.
4. Информатика/Под ред. Н. В. Макаровой. М.: Финансы и статистика, 1997.

5. Ковалев В. В., Уланов В. А. Финансовые и коммерческие вычисления в исторической ретроспективе// Вестн. Санкт-Петерб. ун. Сер. 5. Экономика. 1999. Вып. 4.
6. Кутуков И. Б. Основы финансовой и страховой математики. Методы расчета кредитных, инвестиционных, пенсионных и страховых схем. М.: Дело, 1998.
7. Новиков Ф., Яценко А. Microsof Office в целом. СПб.: BHV, 1995.
8. Овчарено Е. К., Ильина О. П., Балдыбердин Е. В. Финансово-экономические расчеты в Excel. М., 1999.
9. Роутледж Д. Р., Валнум К. Ваш персональный компьютер. М.: БИНОМ, 1995.
10. Уотшем Т. Дж., Паррамоу К. Количественные методы в финансах. М.: ЮНИТИ, 1999.

Глава 7. Системы управления базами данных

В самом широком смысле любая программа имеет дело с некоторой внешней по отношению к ее коду информацией, задающей какие-либо параметры или режим ее работы. Такую информацию также называют данными программы. Очевидно, что в зависимости от типа решаемых задач проблемы организации работы с данными будут качественно различными. В подавляющем большинстве случаев при решении хозяйственных, экономических и финансовых задач приходится иметь дело с обширными специфически структурированными и взаимозависимыми массивами данных. Такие сложные наборы данных традиционно принято называть базами данных.

7.1. Основные понятия теории баз данных

7.1.1. Понятие базы данных

Базу данных (БД) можно определить как унифицированную совокупность данных, совместно используемую различными задачами в рамках некоторой единой автоматизированной *информационной системы (ИС)*.

Теория управления базами данных как самостоятельная дисциплина начала развиваться приблизительно с начала 50-х годов двадцатого столетия. За это время в ней сложилась определенная система фундаментальных понятий. Приведем некоторые из них.

Предметной областью принято называть часть реального мира, подлежащую изучению с целью организации управления в этой сфере и последующей автоматизации процесса управления. В рамках данной книги для нас в первую очередь представляют интерес предметные области, так или иначе связанные со сферой экономики и финансов.

Объектом называется элемент информационной системы, сведения о котором хранятся в базе данных. Иногда объект также называют сущностью (от англ. *entity*). *Классом* объектов называют их совокупность, обладающую одинаковым набором свойств.

Атрибут — это информационное отображение свойств объекта. Каждый объект характеризуется некоторым набором атрибутов.

Ключевым элементом данных называются такой атрибут (или группа атрибутов), который позволяет определить значения других элементов данных.

Запись данных (англ. эквивалент *record*) — это совокупность значений связанных элементов данных.

Первичный ключ — это атрибут (или группа атрибутов), который уникальным образом идентифицируют каждый экземпляр объекта (запись). *Вторичным ключом* называется атрибут (или группа атрибутов), значение которого может повторяться для нескольких записей (экземпляров объекта). Прежде всего вторичные ключи используются в операциях поиска записей.

Процедуры хранения данных в базе должны подчиняться некоторым общим принципам, среди которых в первую очередь следует выделить:

- *целостность и непротиворечивость данных*, под которыми понимается как физическая сохранность данных, так и предотвращение неверного использования данных, поддержка допустимых сочетаний их значений, защита от структурных искажений и несанкционированного доступа;
- *минимальная избыточность данных* обозначает, что любой элемент данных должен храниться в базе в единственном виде, что позволяет избежать необходимости дублирования операций, производимых с ним.

Программное обеспечение, осуществляющее операции над базами данных, получило название СУБД — *система управления базами данных*. Очевидно, что его работа должна быть организована таким образом, чтобы выполнялись перечисленные принципы.

7.1.2. Модели организации данных

Набор принципов, определяющих организацию логической структуры хранения данных в базе, получил название *модели данных*. Модели баз данных определяются тремя компонентами:

- допустимой организацией данных;
- ограничениями целостности;
- множеством допустимых операций.

В теории систем управления базами данных выделяют модели трех основных типов: иерархическую, сетевую и реляционную.

Терминологической основой для иерархической и сетевой моделей являются понятия: атрибут, агрегат и запись. Под *атрибутом* (элементом данных) понимается наименьшая поименованная структурная единица данных. Поименованное множество атрибутов может образовывать *агрегат данных*. В некоторых случаях отдельно взятый агрегат может состоять из множества экземпляров однотипных данных, или, как еще говорят, являться *множественным элементом*. Наконец, *записью* называют составной агрегат, который не входит в состав других агрегатов.

В *иерархической модели* все записи, агрегаты и атрибуты базы данных образуют иерархически организованный набор, то есть такую структуру, в которой все эле-

менты связаны отношениями подчиненности, и при этом любой элемент может подчиняться только одному какому-нибудь другому элементу. Такую форму зависимости удобно изображать с помощью древовидного графа (схемы, состоящей из точек и стрелок, которая связна и не имеет циклов). Пример иерархической структуры базы данных приведен на рис. 7.1.

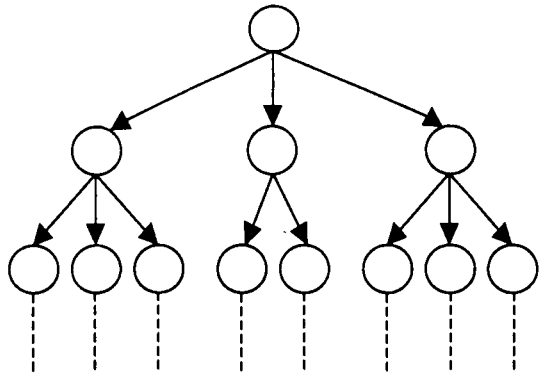


Рис. 7.1. Схема иерархической модели данных

Типичным представителем семейства баз данных, основанных на иерархической модели, является Information Management System (IMS) фирмы IBM, первая версия которой появилась в 1968 г.

Концепция *сетевой модели* данных связана с именем Ч. Бахмана. Сетевой подход к организации данных является расширением иерархического. В иерархических структурах запись-потомок должна иметь в точности одного предка; в сетевой структуре данных потомок может иметь любое число предков (рис. 7.2).

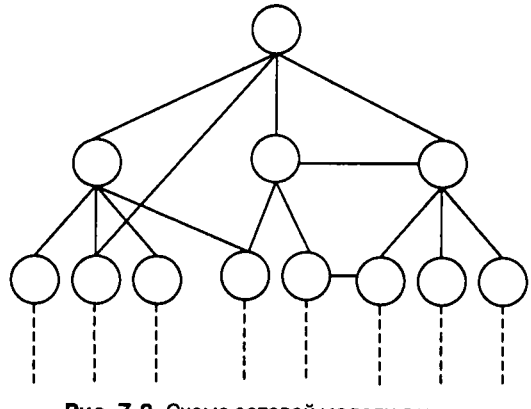


Рис. 7.2. Схема сетевой модели данных

Сетевая БД состоит из набора записей и набора связей между этими записями, точнее, из набора экземпляров записей заданных типов (из допустимого набора типов) и набора экземпляров из заданного набора типов связи.

Примером системы управления данными с сетевой организацией является Integrated Database Management System (IDMS) компании Cullinet Software Inc., разработанная в середине 70-х годов. Она предназначена для использования на «больших» вычислительных машинах. Архитектура системы основана на предложениях Data Base Task Group (DBTG), Conference on Data Systems Languages (CODASYL), организации, ответственной за определение стандартов языка программирования Кобол.

Среди достоинств систем управления данными, основанных на иерархической или сетевой моделях, могут быть названы их компактность и, как правило, высокое быстродействие, а среди недостатков — неуниверсальность, высокая степень зависимости от конкретных данных.

7.1.3. Реляционная модель данных

Концепции реляционной модели впервые были сформулированы в работах американского ученого Э. Ф. Кодда. Откуда происходит ее второе название — модель Кодда.

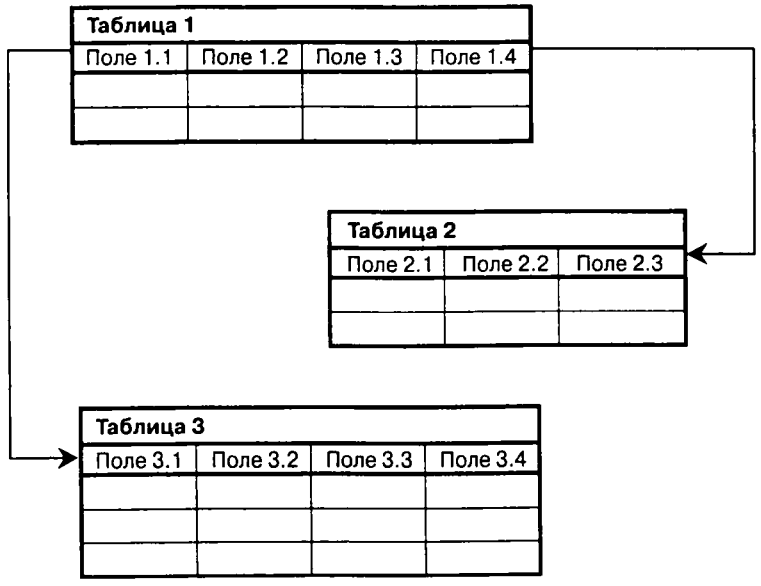


Рис. 7.3. Схема реляционной модели данных

В *реляционной модели* объекты и взаимосвязи между ними представляются с помощью таблиц (рис. 7.3). Для ее формального определения используется фундаментальное понятие *отношения*. Собственно говоря, термин «реляционная» происходит от английского *relation* — отношение.

Если заданы произвольные конечные множества $D_1, D_2, ..., D_n$, то *декартовым произведением* этих множеств $D_1 \times D_2 \times ... \times D_n$ называют множество всевозможных наборов вида $(d_1, d_2, ..., d_n)$, где $d_1 \in D_1, d_2 \in D_2, ..., d_n \in D_n$. *Отношением* R,

определенным на множествах $D_1, D_2, ..., D_n$, называется подмножество декартова произведения $D_1 \times D_2 \times ... \times D_n$. При этом множества $D_1, D_2, ..., D_n$ называются *доменами* отношения, а элементы декартова произведения — *кортежами* отношения. Число n определяет *степень* отношения, а количество кортежей — его *мощность*. Наряду с понятиями домена и кортежа при работе с реляционными таблицами используются альтернативные им понятия *поля* и *записи*.

В реляционной базе данных каждая таблица должна иметь первичный ключ (ключевой элемент) — поле или комбинацию полей, которые единственным образом идентифицируют каждую строку в таблице.

Важным преимуществом реляционной модели является то, что в ее рамках действия над данными могут быть сведены к операциям реляционной алгебры, которые выполняются над отношениями. Это такие операции, как объединение, пересечение, вычитание, декартово произведение, выборка, проекция, соединение, деление.

Важнейшей проблемой, решаемой при проектировании баз данных, является создание такой их структуры, которая бы обеспечивала минимальное дублирование информации и упрощала процедуры обработки и обновления данных. Коддом был предложен некоторый набор формальных требований универсального характера к организации данных, которые позволяют эффективно решать перечисленные задачи. Эти требования к состоянию таблиц данных получили название *нормальных форм*. Первоначально были сформулированы три нормальные формы. В дальнейшем появилась нормальная форма Бойса—Кодда и нормальные формы более высоких порядков. Однако они не получили широкого распространения на практике.

- Говорят, что отношение находится в *первой нормальной форме*, если все его атрибуты являются простыми.
- Говорят, что отношение находится во *второй нормальной форме*, если оно удовлетворяет требованиям первой нормальной формы и каждый неключевой атрибут функционально полно зависит от ключа (однозначно определяется им).
- Говорят, что отношение находится в *третьей нормальной форме*, если оно удовлетворяет требованиям второй нормальной формы и при этом любой неключевой атрибут зависит от ключа *нетранзитивно*. Заметим, что транзитивной называется такая зависимость, при которой какой-либо неключевой атрибут зависит от другого неключевого атрибута, а тот, в свою очередь, уже зависит от ключа.

Принципиальным моментом является то, что для приведения таблиц к состоянию, удовлетворяющему требованиям нормальных форм, или, как еще говорят, для *нормализации данных* над ними, должны быть осуществлены перечисленные выше операции реляционной алгебры.

Основным достоинством реляционной модели является ее простота. Именно благодаря ей она положена в основу подавляющего большинства реально работающих СУБД.

7.1.4. Язык SQL

В разработанной Коддом реляционной модели были определены как требования к организации таблиц, содержащих данные, так и язык, позволяющий работать с ними. Впоследствии этот язык получил название *SQL* (*Structured Query Language* – структурированный язык запросов). SQL был впервые реализован фирмой IBM в начале 70-х годов двадцатого века под названием Structures English Query Language (SEQUEL). Он был ориентирован на управление прототипом реляционной базы данных IBM–System R. В дальнейшем SQL стал стандартом *de facto* языка работы с реляционными базами данных. Этот его статус был впервые зафиксирован в 1986 году Американским национальным институтом стандартов (ANSI). Другими достаточно известными стандартами SQL стали стандарты ANSI SQL-92, ISO SQL-92, X/Open[1].

В составе SQL могут быть выделены следующие группы инструкций:

- язык описания данных – DDL (Data Definition Language);
- язык манипулирования данными – DML (Data Manipulation Language);
- язык управления транзакциями.

Инструкции DDL предназначены для создания, изменения и удаления объектов базы данных. Их описание приведено в табл. 7.1.

Таблица 7.1. Инструкции языка определения данных (DDL)

Инструкция	Назначение
CREATE	Создание новых объектов (таблиц, полей, индексов и т. д.)
DROP	Удаление объектов
ALTER	Изменение объектов

Например, нам необходимо создать таблицу, содержащую данные по каталогу фирм, каждая фирма в котором характеризуется кодом, наименованием, местом расположения штаб-квартиры, размером уставного фонда. Данной операции соответствует SQL-выражение

```
CREATE TABLE Фирмы
     (   КодФирмы     TEXT (5),
         НазвФирмы    TEXT (30),
         АдресФирмы   TEXT (40),
         УстФонд      DOUBLE       );
```

Отметим, что допустимые имена полей создаваемой таблицы и типы содержащихся в них данных могут варьироваться для различных версий и диалектов SQL. Если нам понадобится изменить структуру таблицы Фирмы – допустим, добавить к ней еще одну колонку с фамилией директора, то сделать это можно с помощью SQL-инструкции:

```
ALTER TABLE Фирмы ADD COLUMN Директор TEXT (30);
```

[1] ISO – Национальная организация стандартов, X/Open – группа стандартов для UNIX.

а выражение, дающее команду на уничтожение таблицы, будет выглядеть так:

```
DROP TABLE Фирмы;
```

Инструкции DML (табл. 7.2) позволяют выбирать данные из таблиц, а также добавлять, удалять и изменять их.

Таблица 7.2. Инструкции языка манипулирования данными (DML)

Инструкция	Назначение
SELECT	Выполнение запроса к базе данных с целью отбора записей, удовлетворяющих заданным критериям
INSERT	Добавление записей в таблицы базы данных
UPDATE	Изменение значений отдельных записей и полей
DELETE	Удаление записей из базы данных

SELECT — команда на выборку записей из базы данных — является наиболее часто используемой SQL-инструкцией. Сфера данных, которыми она манипулирует, определяется с помощью специальных предложений. Перечень основных предложений языка SQL приведен в табл. 7.3.

Таблица 7.3. Основные предложения языка SQL

Инструкция	Назначение
FROM	Указывает имя таблицы, из которой должны быть отобраны данные
WHERE	Специфицирует условия, которым должны удовлетворять выбираемые данные
GROUP BY	Определяет, что выбираемые записи должны быть сгруппированы
HAVING	Задает условие, которому должна удовлетворять каждая группа отобранных записей
ORDER BY	Специфицирует порядок сортировки записей

Примером простейшего применения инструкции SELECT может служить команда на выборку всех данных из таблицы Фирмы:

```
SELECT * FROM Фирмы;
```

Однако, вообще говоря, данная инструкция представляет собой весьма мощный инструмент манипуляции с содержимым баз данных. Так, выражение

```
SELECT Int([УстФонд]/500)*500 AS Диапазон,
     Count(КодФирмы) AS ЧислоФирм
     FROM Фирмы
     GROUP BY Int([УстФонд]/500)*500;
```

задает команду на вывод данных о распределении значений уставных фондов фирм по интервалам длиной 500 денежных единиц (д. е.), то есть сколько фирм имеют уставный фонд менее 500 д. е., от 500 до 1000 д. е. и т. д.

Третьей составной частью SQL является язык управления транзакциями. *Транзакция* — это логически завершенная единица работы, содержащая одну или более элементарных операций обработки данных. Все действия, составляющие транзакцию, должны либо выполниться полностью, либо полностью не выполниться. Инструкции языка управления транзакциями приведены в табл. 7.4.

Таблица 7.4. Инструкции языка управления транзакциями

Инструкция	Назначение
COMMIT	Фиксация в базе данных всех изменений, сделанных текущей транзакцией
SAVEPOINT	Установка точки сохранения (начала транзакции)
ROLLBACK	Откат изменений, сделанных с момента начала транзакции

В большинстве СУБД элементарные команды, составляющие тело транзакции, выполняются над некоторой буферной копией данных, и только если их удается успешно довести до конца, происходит окончательное обновление основной базы. Транзакция начинается от точки сохранения, задаваемой инструкцией SAVEPOINT, и может быть завершена по команде COMMIT или прервана по команде ROLLBACK (откат). Также в современных системах управления данными предусмотрены средства автоматического отката транзакций при возникновении системных сбоев. Таким образом, *механизм управления транзакциями является важнейшим инструментом поддержания целостности данных.*

7.1.5. Программные системы управления базами данных

Кратко остановимся на конкретных программных продуктах, относящихся к классу СУБД. На самом общем уровне все СУБД можно разделить:

- на профессиональные, или промышленные;
- персональные (настольные).

Профессиональные (промышленные) СУБД представляют собой программную основу для разработки автоматизированных систем управления крупными экономическими объектами. На их базе создаются комплексы управления и обработки информации крупных предприятий, банков или даже целых отраслей. Первостепенными условиями, которым должны удовлетворять профессиональные СУБД, являются:

- возможность организации совместной параллельной работы большого количества пользователей;
- масштабируемость, то есть возможность роста системы пропорционально расширению управляемого объекта;
- переносимость на различные аппаратные и программные платформы;

- устойчивость по отношению к сбоям различного рода, в том числе наличие многоуровневой системы резервирования хранимой информации;
- обеспечение безопасности хранимых данных и развитой структурированной системы доступа к ним.

Промышленные СУБД к настоящему моменту имеют уже достаточно богатую историю развития. В частности, можно отметить, что в конце 70-х — начале 80-х годов в автоматизированных системах, построенных на базе больших вычислительных машин, активно использовалась СУБД Adabas. В настоящее время характерными представителями профессиональных СУБД являются такие программные продукты, как Oracle, DB2, Sybase, Informix, Ingres, Progress.

Основоположниками СУБД Oracle стала группа американских разработчиков (Ларри Эллисон, Роберт Майнер и Эдвард Оутс), которые более двадцати лет тому назад создали фирму Relational Software Inc. и поставили перед собой задачу создать систему, на практике реализующую идеи, изложенные в работах Э. Ф. Кодда и К. Дж. Дейта. Результатом их деятельности стала реализация переносимой реляционной системы управления базами данных с базовым языком обработки SQL. В 1979 г. заказчикам была представлена версия Oracle для мини-компьютеров PDP-11 фирмы Digital Equipment Corporation сразу для нескольких операционных систем: RSX-11, IAS, RSTS и UNIX. Чуть позже Oracle был перенесен на компьютеры VAX под управлением VAX VMS. Значительная часть кода была написана на ассемблере, и поэтому процесс переноса системы на новую платформу требовал значительных усилий. Основным отличием Oracle очередной, третьей версии было то, что она была полностью написана на языке С. Такое решение обеспечивало переносимость системы на многие новые платформы, в частности, на различные клоны UNIX. Второй важной особенностью новой (1983 г.) версии была поддержка концепции транзакции. Примерно в это же время фирма получила новое имя — Oracle Corporation — и заняла лидирующее место на рынке производителей СУБД. Четвертая версия Oracle характеризовалась расширением перечня поддерживаемых платформ и операционных систем. Oracle был перенесен как на большие ЭВМ фирмы IBM (мэйнфреймы), так и на персональные компьютеры, работающие под управлением MS DOS. Именно в четвертой версии был сделан важный шаг в развитии технологии поддержки целостности баз данных. Для многопользовательских систем было предложено оригинальное решение Oracle поддержки «непротиворечивости чтения». В пятой версии была впервые реализована СУБД с архитектурой «клиент—сервер». Последующие версии СУБД Oracle были ориентированы на построение крупномасштабных систем обработки транзакций, изменение методов реализации систем ввода/вывода, буферизации, подсистем управления параллельным доступом, резервирования и восстановления. Также была реализована поддержка симметричных мультипроцессорных архитектур.

Проект и экспериментальный вариант СУБД Ingres были разработаны в университете Беркли под руководством одного из наиболее известных в мире ученых и специалистов в области баз данных Майкла Стоунбрейкера. С самого начала СУБД Ingres разрабатывалась как мобильная система, функционирующая в среде ОС UNIX. Первая версия Ingres была рассчитана на 16-разрядные компьютеры

и работала главным образом на машинах серии PDP. Это была первая СУБД, распространяемая бесплатно для использования в университетах. Впоследствии группа Стоунбрейкера перенесла Ingres в среду ОС UNIX BSD, которая также была разработана в университете Беркли. Семейство СУБД Ingres из университета Беркли принято называть университетской Ingres. В начале 80-х была образована компания RTI (Relational Technology Inc.), которая разработала и стала продвигать коммерческую версию СУБД Ingres. В настоящее время коммерческая Ingres поддерживается, развивается и продается компанией Computer Associates. Сейчас это одна из наиболее развитых коммерческих реляционных СУБД. В то же время, по поводу университетской Ingres имеется много высококачественных публикаций. Более того, университетскую Ingres можно опробовать на практике и даже посмотреть ее исходные тексты.

Перечисленные выше (для СУБД Oracle) тенденции носят универсальный характер и определяют пути развития других программных продуктов, что вполне объясняется жесткой конкурентной ситуацией, сложившейся на данном рынке.

Персональные системы управления данными — это программное обеспечение, ориентированное на решение задач локального пользователя или компактной группы пользователей и предназначенное для использования на микроЭВМ (персональном компьютере). Это объясняет и их второе название — настольные. Определяющими характеристиками настольных систем являются:

- относительная простота эксплуатации, позволяющая создавать на их основе работоспособные приложения как «продвинутым» пользователям, так и тем, чья квалификация невысока;
- относительно ограниченные требования к аппаратным ресурсам.

Исторически первой среди персональных СУБД, получивших массовое распространение, стала Dbase фирмы Ashton-Tate (впоследствии права на нее перешли к фирме Borland, а с 1999 г. данная программа поддерживается фирмой dBASE Inc.). В дальнейшем серия реляционных персональных СУБД пополнилась такими продуктами, как FoxBase/FoxPRO (Fox Software, в дальнейшем — Microsoft), Clipper (Nantucket, затем — Computer Associates), R:base (Microrim), Paradox (Borland, на настоящий момент правами владеет фирма Corel), Access (Microsoft), Approach (Lotus).

Завоевавшие широкую популярность в России системы Dbase, FoxPRO и Clipper работали с таблицами данных, размещавшихся в файлах, имевших расширение *.dbf (термин dbf-формат стал общепринятым). Впоследствии семейство этих баз данных получило интегрированное наименование Xbase.

Несмотря на неизбежные различия, обусловливавшиеся замыслами разработчиков, все перечисленные системы в ходе своей эволюции приобрели ряд общих конструктивных черт, среди которых, прежде всего, могут быть названы:

- наличие визуального интерфейса, автоматизирующего процесс создания средств манипуляции данными, — экранных форм, шаблонов отчетов, запросов и т. п.;
- наличие инструментов создания объектов базы данных в режиме диалога: Experts в Paradox, Wizards в Access, Assistants в Approach;

- наличие развитого инструментария создания программных расширений в рамках единой среды СУБД: язык разработки приложений PAL в Paradox, VBA (Visual Basic for Applications) в Access, Lotus Script в Approach;
- встроенная поддержка универсальных языков управления данными, например SQL или QBE (Query By Example).

Среди СУБД, которые, условно говоря, занимают промежуточное положение между настольными и промышленными системами, могут быть названы SQLWindows/SQLBase фирмы Centura (до 1996 г. Gupta), Interbase (Borland), наконец, Microsoft SQL Server.

В завершении раздела необходимо отметить, что в последние годы наметилась устойчивая тенденция к стиранию четких граней между настольными и профессиональными системами. Последнее, в первую очередь, объясняется тем, что разработчики в стремлении максимально расширить потенциальный рынок для своих продуктов постоянно расширяют набор их функциональных характеристик.

7.1.6. Применение СУБД в экономике

Очевидно, что экономические задачи, для решения которых необходимо применять программное обеспечение СУБД, весьма обширны и разнообразны. На его основе строятся автоматизированные системы управления предприятий различных уровней (от малых до крупных). Оно лежит в основе практически всех прикладных бухгалтерских программ (например, «1С: Бухгалтерия», «Парус» и др.). Одновременно СУБД применяются для автоматизации систем управления, мониторинга и прогнозирования развития отраслей и экономики страны в целом.

В качестве примера мы более подробно остановимся на вопросах использования СУБД при создании прикладного программного обеспечения, решающего задачи управления работой банков и финансовых компаний, или *автоматизированных банковских систем (АБС)*.

В настоящее время среди ведущих российских разработчиков программных продуктов в классе АБС могут быть названы фирмы «ПрограмБанк», «Диасофт», «Инверсия», «Асофт». В частности, фирмой «ПрограмБанк» разработаны такие известные банковские системы, как «Центавр», «Афина», «Гефест».

В середине 1998 г. компании «ПрограмБанк» и «Диасофт» объединили усилия в области разработки систем автоматизации, рассчитанных на крупные и крупнейшие банки. Учрежденная ими дочерняя компания «Диасофт+ПрограмБанк» сосредоточилась исключительно на развитии и продвижении информационной системы управления банковской деятельностью (ИСУБД) «Новая Афина», в основу которой легли ИБС «Афина» компании «ПрограмБанк» и разработка «Диасофта» – DiasoftBANK 5NT. Данная банковская система разработана на основе программной платформы Oracle.

ИСУБД «Новая Афина» обеспечивает комплексную автоматизацию всех направлений деятельности банка, финансовые методы управления им, поддержку текущего законодательства и правил ведения бухгалтерского учета, ведение планов

счетов произвольной структуры, поддержку различных форм платежного документооборота и маршрутизацию прохождения платежей с использованием различных вариантов верификации документов. Также в рамках ИСУБД решаются задачи управления многофилиальной структурой банка в едином информационном пространстве в режиме реального времени, автоматизации мультивалютного расчетно-кассового обслуживания, управления ЛОРО- и НОСТРО-счетами, обработки сообщений S.W.I.F.T., ведения договоров, контрактов и их приложений, формирования бухгалтерской и аналитической отчетности, связи с внешними информационными системами, администрирования и аудита, получения отчетов произвольной формы. Согласно информации, опубликованной на сайте фирмы «ПрограмБанк», ИСУБД «Новая Афина» применяют такие ведущие банки, как Сбербанк РФ, Внешторгбанк РФ, МДМ-банк, а также ряд представительств зарубежных банков.

7.2. СУБД MS Access и ее основные возможности

7.2.1. Общая характеристика СУБД MS Access

Microsoft Access в настоящее время является одной из самых популярных среди настольных (персональных) программных систем управления базами данных. Среди причин такой популярности следует отметить:

- высокую степень универсальности и продуманности интерфейса, который рассчитан на работу с пользователями самой различной квалификации. В частности, реализована система управления объектами базы данных, позволяющая гибко и оперативно переходить из режима конструирования в режим их непосредственной эксплуатации;
- глубоко развитые возможности интеграции с другими программными продуктами, входящими в состав Microsoft Office, а также с любыми программными продуктами, поддерживающими технологию OLE;
- богатый набор визуальных средств разработки.

Нельзя не отметить, что существенной причиной такого широкого распространения MS Access является и мощная рекламная поддержка, осуществляемая фирмой Microsoft. В процессе разработки данного продукта на рынок представлялись его различные версии. Наиболее известными (в некотором смысле этапными) стали Access 2.0, Access 7.0 (он впервые был включен в состав программного комплекса MS Office 95). Позже появились версии Access 97 (в составе MS Office 97) и Access 2000 (в составе MS Office 2000).

Очевидно, что отправной точкой в процессе работы с любой СУБД является создание файла (или группы файлов) базы данных. На рис. 7.4 показано окно, которое появляется после создания новой базы.

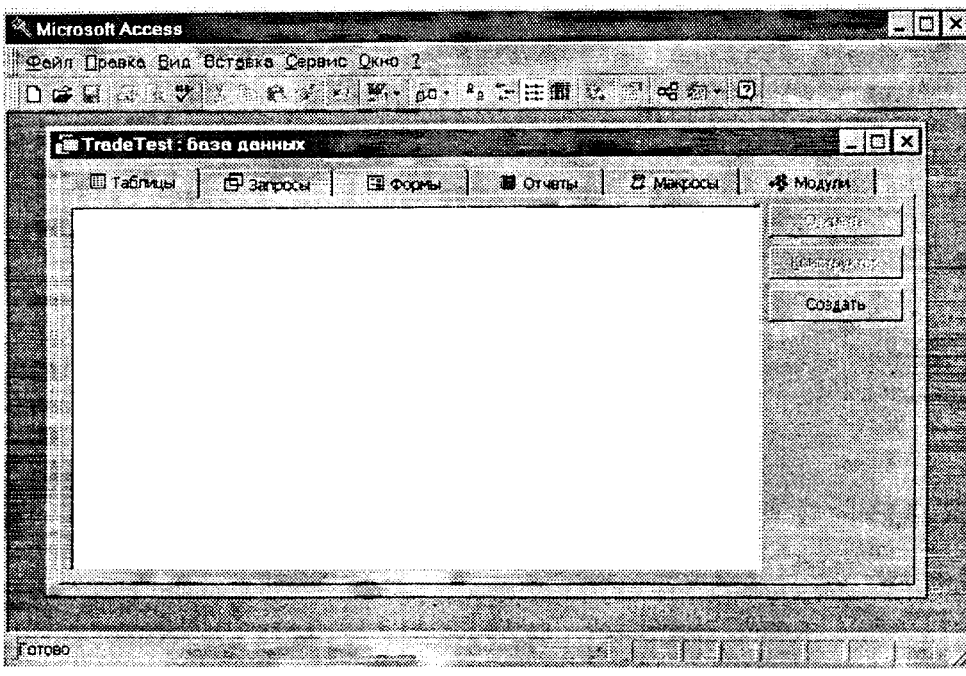


Рис. 7.4. Главное окно базы данных в Access

Основные разделы главного окна соответствуют типам объектов, которые может содержать база данных Access. Это Таблицы, Запросы, Отчеты, Макросы и Модули. Заголовок окна содержит имя файла базы данных. В данном случае он называется TradeTest.

Интерфейс работы с объектами базы данных унифицирован. По каждому из них предусмотрены стандартные режимы работы:

- Создать — предназначен для создания структуры объектов;
- Конструктор — предназначен для изменения структуры объектов;
- Открыть (Просмотр, Запуск) — предназначен для работы с объектами базы данных.

Важным средством, облегчающим работу с Access для начинающих пользователей, являются *мастера* — специальные программные надстройки, предназначенные для создания объектов базы данных в режиме последовательного диалога. Для опытных и продвинутых пользователей существуют возможности более гибкого управления ресурсами и возможностями объектов СУБД в режиме конструктора.

Специфической особенностью СУБД Access является то, что вся информация, относящаяся к одной базе данных, хранится в едином файле. Такой файл имеет расширение *.mdb. Данное решение, как правило, удобно для непрофессиональных пользователей, поскольку обеспечивает простоту при переносе данных с

одного рабочего места на другое. Внутренняя организация данных в рамках mdb-формата менялась от версии к версии, но фирма Microsoft поддерживала их совместимость снизу вверх, то есть базы данных из файлов в формате ранних версий Access могут быть конвертированы в формат, используемый в версиях более поздних.

7.2.2. Основные этапы разработки базы данных в среде MS Access

Как нетрудно догадаться, процесс разработки конкретного программного приложения в среде Access в первую очередь определяется спецификой автоматизируемой предметной области. Однако для большинства из них можно выделить ряд типичных этапов. Это:

- разработка и описание структур таблиц данных;
- разработка схемы данных и задание системы взаимосвязей между таблицами;
- разработка системы запросов к таблицам базы данных и (при необходимости) их интеграция в схему данных;
- разработка экранных форм ввода/вывода данных;
- разработка системы отчетов по данным;
- разработка программных расширений для базы данных, решающих специфические задачи по обработке содержащейся в ней информации, с помощью инструментария макросов и модулей;
- разработка системы защиты данных, прав и ограничений по доступу.

Очевидно, что между перечисленными этапами существует большое количество обратных связей, подразумевающих возврат к более ранним шагам, исходя из вновь открывшихся обстоятельств, которые невозможно было заранее учесть или предвидеть.

Еще раз подчеркнем, что описанная последовательность этапов разработки базы данных в MS Access не является безусловным эталоном. Однако очень часто отклонения от нее свидетельствуют не столько об оригинальности хода мысли разработчика, сколько о погрешностях, допущенных им при планировании процесса разработки, или вообще об отсутствии у него какого-либо плана.

7.2.3. Экономические приложения СУБД MS Access

Продуманность пользовательского интерфейса Access делает его особенно привлекательным в качестве средства решения задач организации и обработки данных для специалистов в области экономики и финансов, одновременно не имеющих квалификации или опыта в профессиональном программировании. Оговоримся, что здесь речь идет о приложениях, создаваемых таким специалистом для собственного пользования. В то же время, как только возникает необходимость в разработке средств для других пользователей, без программирования, как правило, обойтись не удается.

Можно перечислить более чем обширный список возможных приложений Access для решения финансово-экономических задач. Мы остановимся на достаточно условном примере, с помощью которого, однако, можно наглядно проиллюстрировать большинство наиболее важных функциональных возможностей этого программного продукта.

Предположим, что перед нами стоит задача автоматизации процесса управления торгами набором финансовых активов (ценных бумаг) на некотором ограниченном секторе рынка. Для ее решения (*еще раз подчеркнем, при условии относительной ограниченности объемов информации*) хорошо подходит СУБД MS Access.

Представим рассматриваемую ситуацию на содержательном уровне. Пусть на рынке (в некоторой торговой системе) циркулирует определенный набор ценных бумаг (акций), каждая из которых характеризуется наименованием, номинальной ценой, суммарным объемом пакета (то есть сколько всего единиц данной бумаги был эмитировано), датой эмиссии. Одновременно на рынке действуют его субъекты (агенты), которые могут продавать и покупать бумаги. Очевидно, что каждый агент характеризуется по меньшей мере наименованием и величиной средств, которыми он обладает. Таким образом, достаточно естественно выкристаллизовываются четыре массива информации: данные по бумагам, данные по агентам (рынка), данные по принадлежности бумаг агентам (по портфелям) и, наконец, данные по заявкам агентов на покупку или продажу тех или иных бумаг.

Теперь попытаемся описать структуры потоков информации, которые фигурируют в автоматизируемой предметной области, на более логически строгом уровне.

Массив (таблица) данных по существующим активам (присвоим ей имя Бумаги) будет содержать колонки (поля):

- Код бумаги;
- Наименование бумаги;
- Номинальная цена;
- Суммарный объем пакета;
- Дата эмиссии;
- Тип бумаги (например, акция или облигация).

Соответственно, таблица Агенты будет состоять из колонок:

- Код агента;
- Наименование агента;
- Объем денежных средств, которыми обладает агент;
- Комментарий по агенту.

Заметим, что поля Код бумаги и Код агента являются ключами, обеспечивающими уникальную идентификацию записей в соответствующих таблицах.

Для хранения информации о содержание портфелей ценных бумаг, которыми владеют агенты, создадим таблицу Портфели со структурой:

- Код бумаги;
- Код агента;
- Количество бумаг данного наименования в портфеле, которым обладает данный агент.

В таблице Портфели мы сталкиваемся с составным ключом, который образует комбинация полей Код бумаги и Код агента. Наконец, информацию намерениях тех или иных агентов продать те или иные бумаги поместим в таблицу Заявки:

- Код заявки;
- Код бумаги;
- Код агента;
- Объем заявки (в единицах измерения, соответствующих бумагам данного наименования);
- Цена заявки.

Отметим, что экономическое содержание, вкладываемое в величину, содержащуюся в поле Объем заявки, может иметь различные интерпретации. Например, можно считать, что если это значение положительно, то это заявка на покупку, а если отрицательно, то — на продажу. Очевидно, что возможны и альтернативные решения по организации данной таблицы. Например, можно было бы создать два отдельных поля: Объем заявки на покупку и Объем заявки на продажу. Дополнительно хочется обратить внимание на те резоны, в соответствии с которыми в качестве ключа использовано отдельное поле Код заявки. Это позволяет одновременно хранить в таблице разные предложения по одной и той же бумаге, поступающие от одного и того же агента.

Простота описанной системы таблиц не должна вводить читателя в заблуждение. Она определяется исключительно условностью рассматриваемого примера, в котором мы из соображений наглядности изложения абстрагировались от многих черт реального процесса торгов ценными бумагами.

7.2.4. Создание таблиц и схем данных

Как уже отмечалось ранее, процесс разработки базы данных в СУБД MS Access начинается с задания описания структур таблиц. Рассмотрим этот процесс более подробно для таблиц примера, описанного в 7.2.3.

Итак, для начала нам необходимо создать описание таблицы Бумаги. Нажав кнопку Создать и выбрав в появившемся вслед диалоговом окне режим Конструктор, мы попадаем в окно, предназначенное для ввода описания структуры создаваемой таблицы. Оно изображено на рис. 7.5.

При создании баз данных, предназначенных для решения финансовых и экономических задач, процесс описания атрибутов полей в создаваемой таблице приоб-

ретает особое значение. Как видно из рис. 7.5, процесс описания атрибутов поля начинается с присвоения ему имени (идентификатора). Желательно, чтобы это имя было, с одной стороны, информативным, а с другой — кратким, что обеспечивает несомненные удобства при дальнейших манипуляциях с ним. Далее необходимо определить тип поля, что, очевидно, должно делаться, исходя из содержания тех данных, которые будут в нем храниться.

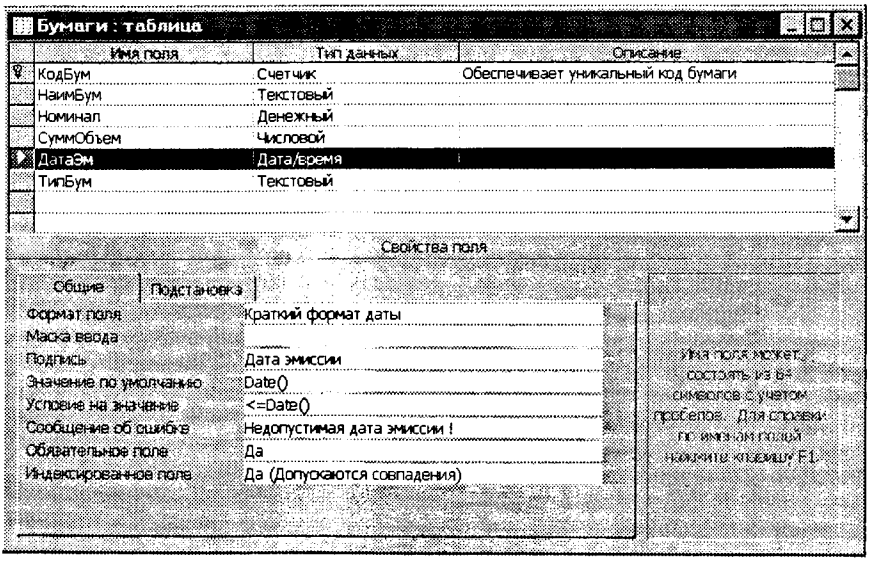


Рис. 7.5. Создание описания структуры таблицы Бумаги

Обратим внимание на тип Счетчик, присвоенный полю КодБум (код бумаги). Он означает, что СУБД будет самостоятельно помещать в это поле некоторое числовое значение для каждой вновь создаваемой записи таблицы, обеспечивая таким образом его уникальность.

Выбор типа данных в Access одновременно определяет набор дополнительных атрибутов соответствующего поля. В частности, поле ДатаЭм (дата эмиссии) имеет тип Дата и, как это показано на рис. 7.5, может иметь дополнительные атрибуты:

- **формат поля**, определяющий условия вывода данных из этого поля (по умолчанию);
- **маска ввода**, определяющая условия ввода данных в поле;
- **подпись** — содержит расширенный заголовок;
- **значение по умолчанию** — позволяет указать значение, автоматически присваиваемое полю при создании новой записи. В нашем случае по умолчанию будет задаваться текущая дата, возвращаемая встроенной функцией Date();
- **условие на значение** — определяет требования к данным, вводимым в поле. Например, для выполнения требования, чтобы дата эмиссии предшествовала текущей, следует задать выражение <=Date();

- **сообщение об ошибке** — определяет текст сообщения, которое будет выводиться в случае нарушения заданного выше условия;
- **обязательное поле** — указывает, требует или нет поле обязательного ввода значения;
- **индексированное поле** — определяет индекс, создаваемый по данному полю. Индекс ускоряет выполнение запросов, в которых используются индексированные поля, и операции сортировки и группировки.

Основываясь на опыте проектирования различных баз, необходимо заметить, что не следует пренебрегать возможностями управления данными, которые открывают дополнительные атрибуты полей. Их грамотное и продуманное использование позволяет организовать централизованный и эффективный контроль за корректностью и целостностью данных.

На завершающем этапе процесса проектирования структуры таблицы происходит задание ключей и индексов. В первом случае достаточно выделить строки, которые должны составить ключевое выражение, и щелкнуть мышью по пиктограмме Ключ на панели инструментов (рис. 7.6). В таблице Бумаги роль уникального ключевого идентификатора выполняет поле КодБум.



Рис. 7.6. Панель инструментов конструктора таблиц

Также при создании таблицы имеет смысл заранее продумать возможные упорядочения, которые могут понадобиться при работе с содержащимися в ней данными. Задание индексов с соответствующими ключевыми выражениями может в дальнейшем существенно ускорить процесс работы (особенно с большими массивами данных). В частности, при работе с данными из таблицы Бумаги весьма вероятно, что нам придется выводить их в алфавитном порядке по названиям, а также отсортированными в порядке убывания дат. Процесс задания соответствующих индексов показан на рис. 7.7.

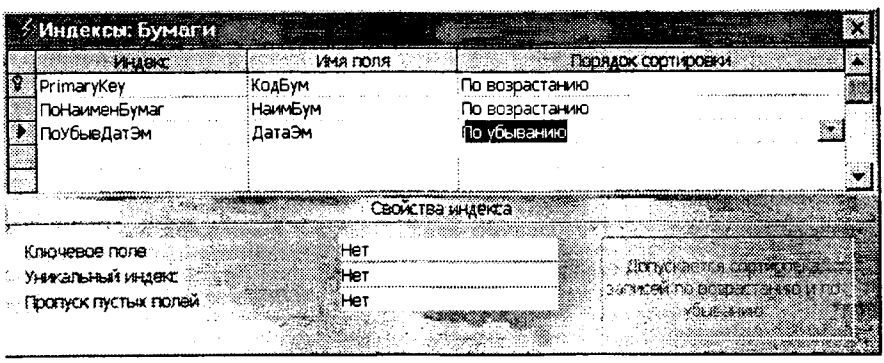


Рис. 7.7. Создание индексов для таблицы

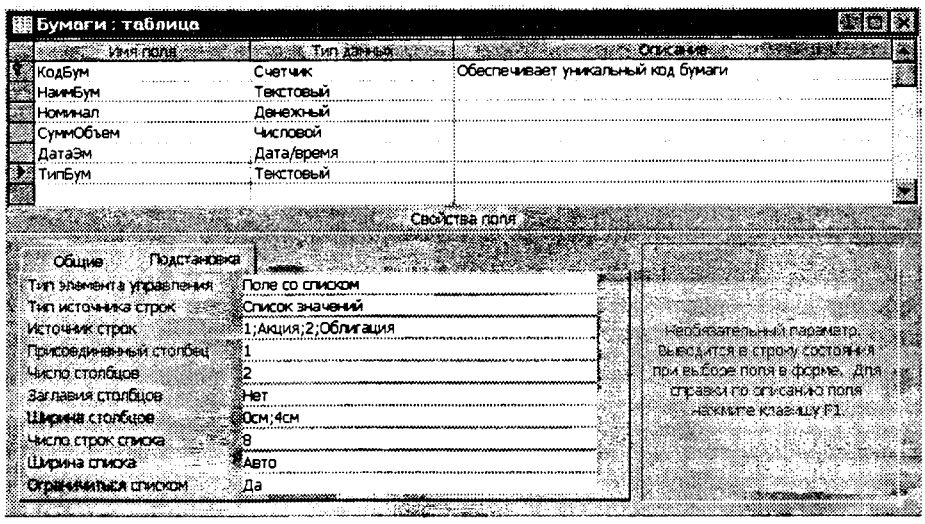


Рис. 7.8. Задание списка подстановки для поля

Эффективным методом решения задач контроля корректности входных данных является ограничение множества допустимых значений поля некоторым списком. Рассмотрим это более подробно на примере поля ТипБум (тип бумаги), которая, допустим, в рассматриваемой торговой системе может быть либо акцией, либо облигацией. Нетрудно заметить, что будет разумным присвоить типу Акция код 1, а типу Облигация — код 2. Это позволит существенно сэкономить место за счет уменьшения объема хранимой информации (особенно при большом количестве записей)[1]. Однако с точки зрения восприятия вводимой информации пользователем гораздо удобнее иметь дело с осмысленным текстом, чем запоминать, какие коды ему соответствуют.

Средством решения этой проблемы в Access является задание подстановочного списка значений для поля. Для этого следует выбрать вкладку Подстановка в окне Свойства поля, далее для свойства Тип элемента управления задать значение Список. На рис. 7.8 показано, как описать другие свойства элемента управления Список, чтобы организовать заполнение поля ТипБум требуемыми значениями.

После создания описания структуры таблицы можно перейти в режим непосредственного ввода в нее данных. Как уже говорилось, важным преимуществом интерфейса СУБД Access является продуманная гибкая система перехода от режима Конструктора к режиму ввода данных в таблицу (Режим таблицы). Такой переход можно осуществить, щелкнув мышью по пиктограмме Вид, расположенной на панели инструментов, либо выбрав функцию меню Вид ▸ Режим таблицы.

На рис. 7.9 показано окно режима непосредственного ввода данных в таблицу Бумаги.

---

[1] Здесь уместно привести простейший расчет: один лишний байт в поле при 10 тыс. записей в таблице увеличивает дополнительный объем, занимаемый базой, почти на 10 Кбайт.

Бумаги : таблица

Код бумаги	Наименование бумаги	Номинал	СуммОбъем	Дата эмисс	Тип бумаги
2	ОАО "Автоматика-Север"	1 000р.	2000	10.01.93	Акция
3	Красная шапочка	1 500р.	10000	01.10.95	Акция
4	Индиго-банк	900р.	5000	20.10.96	Облигация
(Счетчик)		0р.	0	08.03.00	Акция

Запись: 3 из 3

Рис. 7.9. Ввод данных в таблицу Бумаги

Хочется еще раз обратить внимание читателя на то, что выбор типа бумаги осуществляется из заранее предопределенного списка.

7.2.5. Создание схемы данных

Очевидно, что те действия, которые были подробно описаны для таблицы Бумаги, следует проделать и для остальных информационных массивов: Агенты, Портфели, Заявки. В результате мы получим систему таблиц базы данных TradeTest. Подчеркнем, именно систему, так как находящиеся в них данные тесно и содержательно связаны между собой. Действительно, данные, находящиеся в поле Код агента таблицы Портфели, должны быть согласованы по типу и размеру с данными, находящимися в одноименном поле таблицы Бумаги. Более того, логика рассматриваемой задачи требует, чтобы, работая с информацией, относящейся к портфелю, мы могли одновременно обратиться к данным, характеризующим текущего агента, и т. д.

Механизм описания логических связей между таблицами в Access реализован в виде объекта, называемого Схемой данных. Перейти к ее созданию можно из панели инструментов База данных, доступной из главного окна. Альтернативный вариант вызова данного режима доступен через меню Сервис ▸ Схема данных. Вид, который будет иметь схема данных для построенных на предыдущих шагах таблиц, представлен на рис. 7.10.

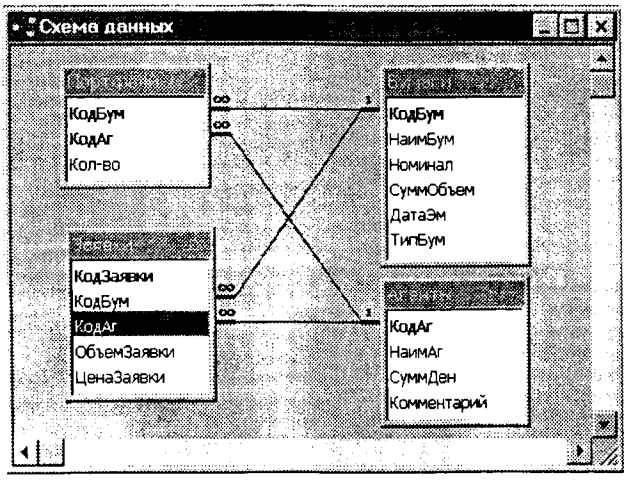


Рис. 7.10. Создание схемы данных

Интерфейс задания связей между полями в схеме основан на «перетаскивании» (перемещении при нажатой левой кнопки мыши) выбранного поля и «наложении» его на то поле, с которым должна быть установлена связь. Для связывания сразу нескольких полей их следует перемещать при нажатой клавише Ctrl.

Выделяют несколько типов связей между таблицами в схеме. **«Один к одному» (1:1)** — одному значению поля в одной таблице соответствует только одно значение поля в другой. **«Один ко многим» (1:∞)** — одному значению поля в одной таблице соответствует несколько (одно или более) значений в другой.

Важнейшей задачей, которую позволяет решать схема, является обеспечение *логической целостности данных* в базе. Так, в базе данных TradeTest нарушение целостности может возникнуть в случае удаления из таблицы Бумаги записей по тем бумагам, о которых существуют записи в таблицах Портфели или Заявки, в результате чего в их составе окажутся ссылки на «потерянные» коды. Очевидно, что это можно предотвратить, если каскадно удалить как записи из таблицы Бумаги, так и записи из связанных с ней таблиц. Такой эффект в Access может быть достигнут за счет задания определенных свойств для связи. Чтобы это сделать, необходимо щелкнуть кнопкой мыши, находясь на линии схемы, обозначающей связь. После этого появляется диалоговое окно, предназначенное для изменения свойств связи. Как видно из рис. 7.11, в рамках режима обеспечения целостности данных можно по выбранной связи задать как каскадное обновление значений для связанных полей, так и каскадное удаление связанных записей.

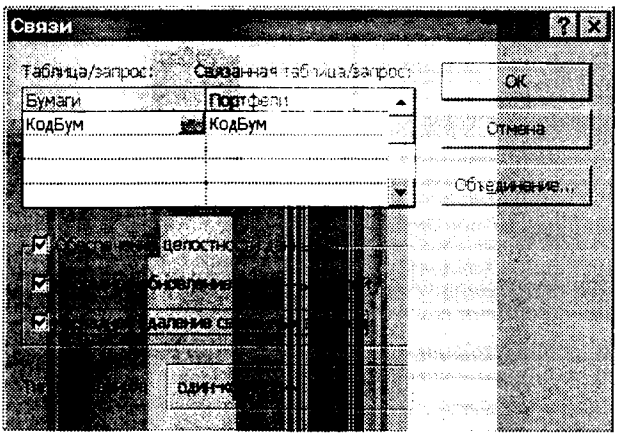


Рис. 7.11. Задание свойств связи

7.2.6. Разработка запросов к базе данных

Появление даже очень небольшой таблицы мгновенно приводит к возникновению целого комплекса проблем, связанных с необходимостью обработки содержащихся в ней данных. К простейшим задачам обработки могут быть отнесены:

- поиск записи по условию (см. функцию меню Правка ▸ Найти);
- сортировка записей в требуемом порядке (см. функцию меню Записи ▸ Сортировка);

- получение выборки записей таблицы, удовлетворяющей заданному условию, или, как еще говорят, задание фильтра для таблицы (Записи ▸ Фильтр).

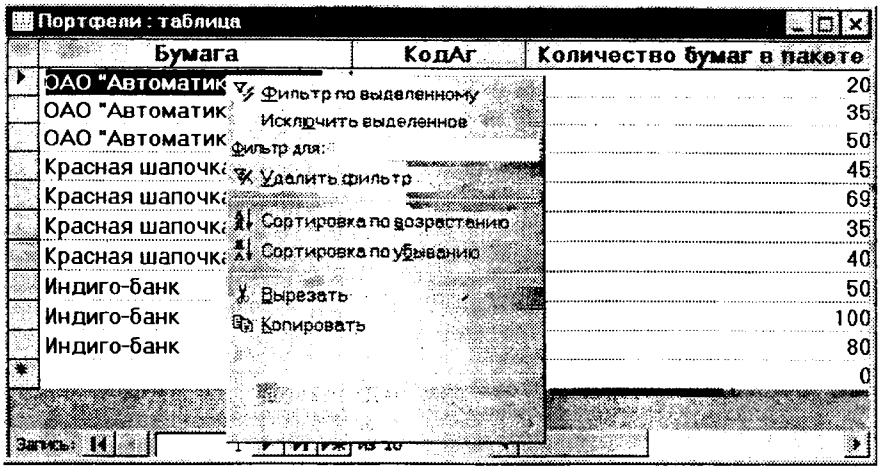


Рис. 7.12. Контекстное меню работы с данными в таблице

Перечисленные функции также доступны из контекстного меню, активизирующегося по нажатии правой клавиши мыши (рис. 7.12). Данный интерфейс представляется особенно удобным при практической работе с таблицами Access. Однако этих возможностей явно недостаточно для задач обработки данных, которые возникают в реальных экономических приложениях. Для их решения в СУБД Access служит развитой инструментарий запросов к базе данных.

Понятие *запроса* в Access употребляется в расширительном плане. Его следует трактовать как некоторую команду на выбор, просмотр, изменение, создание или удаление данных. Также нельзя не отметить значение запросов для решения задач анализа данных.

Наиболее распространенным и, если так можно выразиться, естественным типом запросов является *запрос на выборку*. Данный тип, собственно говоря, и устанавливается по умолчанию для вновь создаваемого запроса.

При работе с системой данных очень часто возникает задача соединения данных из различных связанных таблиц в одну. Так, в рамках нашего примера естественной представляется проблема построения таблицы, содержащей информацию по содержанию портфелей и имеющей следующую структуру:

- **Наименование бумаги**;
- **Наименование агента**;
- **Тип бумаги**;
- **Номинальная стоимость пакета**, вычисляемая как произведение номинальной цены на количество бумаг данного вида, которым обладает текущий агент.

Для ее решения следует перейти к разделу Запросы главного окна базы данных, нажать на кнопку Создать и выбрать режим Конструктор. Процесс создания запроса

начинается с выбора таблиц (в том числе и других запросов), на основе которых строится запрос. В дальнейшем состав этого набора может быть изменен. Наш запрос будет построен на основе данных таблиц Портфели, Агенты и Бумаги. Заметим, что при добавлении таблиц к запросу по умолчанию добавляются и связи между ними, заданные в схеме.

В процессе формирования запроса можно выделить ряд принципиальных этапов:

- описание структуры запроса (то есть указание того, какая информация должна выводиться в колонках таблицы запроса);
- задание порядка, в котором данные должны выводиться при выполнении запроса;
- задание условий вывода записей в запросе.

На рис. 7.13 показано окно конструктора запроса.

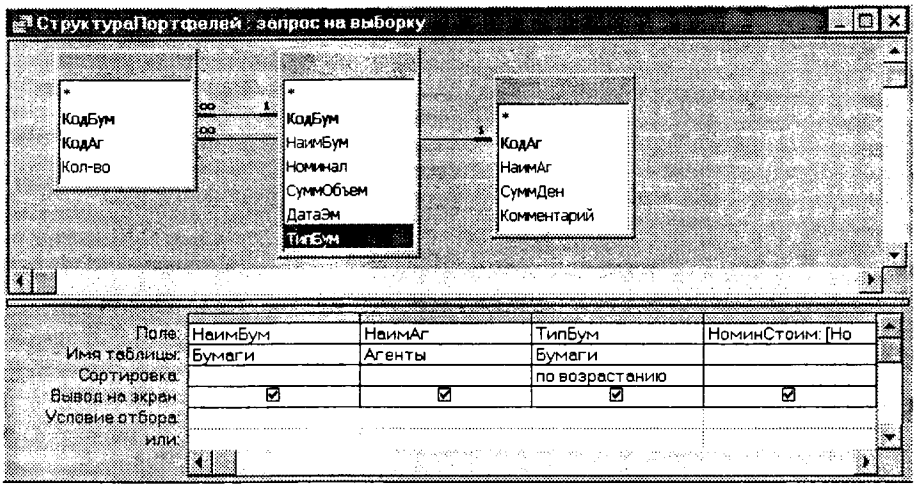


Рис. 7.13. Окно конструктора запроса

Отметим, что колонки таблицы запроса на рис. 7.13 содержат как поля таблиц, так и выражения, построенные на основе полей. В частности, последняя колонка (ей присвоено имя НоминСтоим) содержит выражение [Номинал]*[СуммОбъем], при этом записи будут выводиться отсортированными по типу бумаг.

По аналогии с принципами организации интерфейса работы с таблицами данных, при конструировании запросов также существует возможность оперативного перехода из режима Конструктор в Режим таблицы. При первом входе в Режим таблицы появляется приглашение сохранить вновь созданный запрос. В данном случае ему дано имя СтруктураПортфелей. На рис. 7.14 показано окно, в котором выводятся записи, соответствующие этому запросу.

Следует обратить внимание на исключительно важную роль механизма запросов в решении проблемы обеспечения минимальной избыточности сохраняемой в базе информации. Действительно, с их помощью мы можем получать произволь-

ное количество виртуальных таблиц, представляющих в самых различных видах и разрезах единственную реально хранимую совокупность данных.

СтруктураПортфелей : запрос на выборку

Наименование бумаги	Наименование агента	Тип бумаги	НоминСтоим
Красная шапочка	Пономарев	Акция	15 000 000р.
Красная шапочка	Жаворонкова	Акция	15 000 000р.
ОАО "Автоматика-Север"	Новиков	Акция	2 000 000р.
Красная шапочка	Мазур	Акция	15 000 000р.
ОАО "Автоматика-Север"	Коровкина	Акция	2 000 000р.
Красная шапочка	Свиридюк	Акция	15 000 000р.
ОАО "Автоматика-Север"	Корельский	Акция	2 000 000р.
Индиго-банк	Жаворонкова	Облигация	4 500 000р.
Индиго-банк	Кравчук	Облигация	4 500 000р.
Индиго-банк	Гурьева	Облигация	4 500 000р.

Запись: 1 из 10

Рис. 7.14. Вывод данных по запросу СтруктураПортфелей

Рассмотрим еще один случай применения запросов для решения задач обработки данных. Достаточно типичной (в том числе для приложений финансово-экономического характера) является проблема группировки данных по тому или иному признаку. Например, в рамках построенной нами базы данных может быть поставлена задача определения суммарного (или среднего) спроса и предложения по ценным бумагам, циркулирующим на рынке. Решить ее можно, построив запрос, содержащий групповые операции. Для активизации возможности их задания в окне Конструктора запросов необходимо включить функцию меню Вид ▸ Групповые операции.

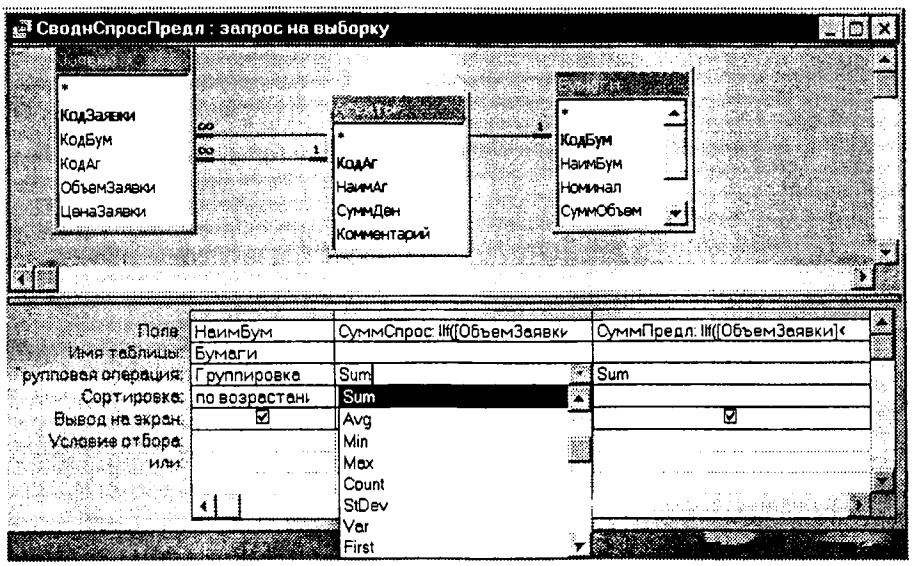


Рис. 7.15. Создание запроса с групповыми операциями

На рис. 7.15 показано окно конструктора в процессе создания запроса, выводящего информацию по суммарному спросу и предложению на ценные бумаги. Операция свертки нескольких записей из таблицы Заявки в одну результирующую запись, осуществляемая для каждого наименования бумаги, определяется командой Группировка, расположенной в строке Групповая операция. Для двух последующих колонок запроса (СуммСпрос и СуммПредл) определены операции суммирования по группе (Sum), расположенные в той же строке, а в строке Поле находятся производные выражения, суммы которых мы хотим получить в запросе. В соответствии с ранее принятыми соглашениями объем суммарного спроса определяется совокупностью всех записей по данной бумаге, имеющих положительное значение в поле ОбъемЗаявки, а объем суммарного предложения — записями, содержащими в данном поле отрицательную величину. Таким образом, для вычисления СуммСпрос необходимо просуммировать If([ОбъемЗаявки]>=0; [ЦенаЗаявки]*[ОбъемЗаявки];0), а для вычисления СуммПредл — If([ОбъемЗаявки]<=0;-1*[ЦенаЗаявки]*[ОбъемЗаявки];0).

ПРИМЕЧАНИЕ

Встроенная функция If(bArg; Arg1; Arg2) возвращает значение аргумента Arg1, если значение аргумента bArg, который может содержать только логическую величину, является истинным (bArg=ИСТИНА), и значение Arg2, если bArg=ЛОЖЬ.

Результат выполнения описанного запроса (ему дано имя СводнСпросПредл) приведен на рис. 7.16.

СводнСпросПредл : запрос на выборку

Наименование бумаги	СуммСпрос	СуммПредл
Индиго-банк	26 900р.	20 000р.
Красная шапочка	56 000р.	0р.
ОАО "Автоматика-Север"	20 000р.	30 000р.

Запись: 1 из 3

Рис. 7.16. Вывод данных по запросу СводнСпросПредл

Также следует обратить внимание читателя на такие важные возможности конструктора запросов, как:

- задание параметров, запрашиваемых при открытии запроса;
- встроенные статистические функции, доступные при задании групповых операций. Они делают запросы мощным инструментом анализа хранимой информации.

В завершение обзора средств построения запросов в СУБД Access следует указать также и на то, что в нее помимо мощного и эффективного визуального конструктора встроен также и режим непосредственного ввода SQL-выражений, определяющего запрос. Данный режим существует параллельно и доступен из меню Вид ▸ Режим SQL (а также из пиктограммы Вид на панели инструментов). Перейдя

в него, в частности, можно просмотреть SQL-выражение, соответствующее ранее построенному запросу СводнСпросПредл. Оно выглядит так:

```
SELECT Бумаги.НаимБум,
Sum(IIf([ОбъемЗаявки]>=0,[ЦенаЗаявки]*[ОбъемЗаявки],0))
        AS СуммСпрос,
   Sum(IIf([ОбъемЗаявки]<=0,-[ЦенаЗаявки]*[ОбъемЗаявки],0))
        AS СуммПредл
FROM Бумаги INNER JOIN
   (Агенты INNER JOIN Заявки ON Агенты.КодАг = Заявки.КодАг)
        ON Бумаги.КодБум = Заявки.КодБум
GROUP BY Бумаги.НаимБум
ORDER BY Бумаги.НаимБум;
```

Пользователь, владеющий синтаксисом языка SQL, может модифицировать данное выражение в ручном режиме. Очевидно, что такая техника работы требует существенно большей квалификации, но одновременно она дает в руки разработчика мощный и универсальный аппарат управления данными.

Говоря о связи между режимом визуального конструктора запросов и режимом построения SQL-выражений, необходимо отметить, что существует естественная и логичная связь между типами запросов и реализующими их SQL-операторами. В частности, запросу на выборку соответствует оператор SELECT, запросу на создание — CREATE, запросу на обновление — UPDATE, запросу на удаление — DELETE и т. д.

7.2.7. Конструирование экранных форм для работы с данными

В 7.2.4 был рассмотрен режим непосредственного ввода данных в таблицу. Очевидно, что он имеет весьма ограниченное применение. Это обусловливается как тем, что длина записи может оказаться достаточно большой и вводить информацию в нее в табличной форме будет технически неудобно, так и соображениями более принципиального характера:

- во-первых, структура таблицы должна строиться на основе логики задач хранения информации, которая, вообще говоря, может существенно отличаться от логики ее накопления и ввода;
- во-вторых, важным показателем качества автоматизированной системы является организация ее системы ввода/вывода в виде, максимально приближенном к традиционным формам представления информации на немашинных носителях. Такие формы, как правило, делают программное обеспечение привлекательным для конечного пользователя, уменьшают период его адаптации ко вновь внедряемой системе и позволяют быстро сосредоточиться на решении основных профессиональных задач;
- в-третьих, в сложной и развитой автоматизированной информационной системе должно обеспечиваться разделение доступа к различным группам полей

и записей для различных категорий пользователей в зависимости от выполняемых ими функций. Также в определенных ситуациях требуется представить одну и ту же информацию либо в различных видах и разрезах, либо в различных сочетаниях с другой информацией.

Для решения как этих, так и многих других проблем организации интерфейса ввода/вывода данных в Access служит механизм электронных форм. Выберем вкладку Формы главного окна базы данных и нажмем кнопку Создать. Появляющееся диалоговое окно позволяет выбрать как таблицу или запрос, для работы с данными которых составляется форма, так и режим ее создания. В зависимости от квалификации пользователя и, естественно, сложности разрабатываемой формы можно либо воспользоваться встроенными программными надстройками-мастерами, либо сразу начать ее создание с нуля в режиме Конструктора. Весьма плодотворным также оказывается комбинированный подход: сначала используется соответствующий мастер, а затем полученная форма дополнительно дорабатывается в «ручном режиме».

Проиллюстрируем сказанное на примере. Создадим форму для работы с таблицей Бумаги, воспользовавшись надстройкой Автоформа: в столбец. В результате получим окно следующего вида.

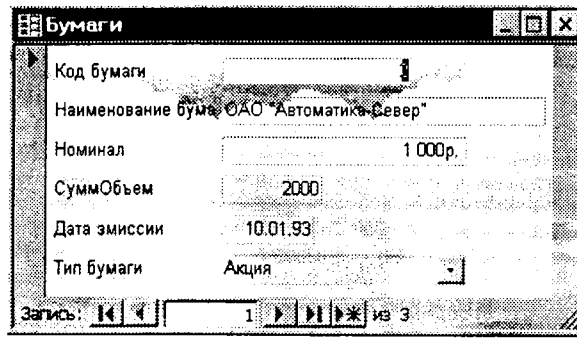


Рис. 7.17. Автоформа в столбец, созданная по таблице Бумаги

По умолчанию форме было предложено присвоить такое же имя, как и у таблицы, на основе которой она была создана, то есть Бумаги. Как видно из рис. 7.17, при создании подписей полей программная надстройка использовала их соответствующие атрибуты, заданные при конструировании таблицы. Последнее не всегда бывает удобным с точки зрения интерфейса пользователя. Для устранения этих и подобных недостатков нам придется вернуться в режим изменения макета формы (кнопка Конструктор либо пиктограмма Вид на панели инструментов).

На рис. 7.18 показана та же форма в режиме Конструктор. Технология процесса проектирования форм в среде Access сводится к добавлению управляющих элементов и изменению их свойств. В связи с этим при переходе в режим Конструктор на экране по умолчанию появляются два дополнительных окна:

- Окно Панель элементов, которое предназначено для выбора очередного добавляемого к проектируемой форме управляющего элемента. В конструктор форм Access встроены такие элементы управления, как надпись, поле, кнопка, флажок, пере-

ключатель, список, набор вкладок и др. Помимо этого к форме можно подключать специальные (дополнительные) элементы управления OLE, что значительно расширяет возможности развития интерфейса управления данными.

- Окно Свойств текущего элемента управления, предназначенное для изменения его атрибутов и настроек, например, цвета, шрифта, размера и т. п.

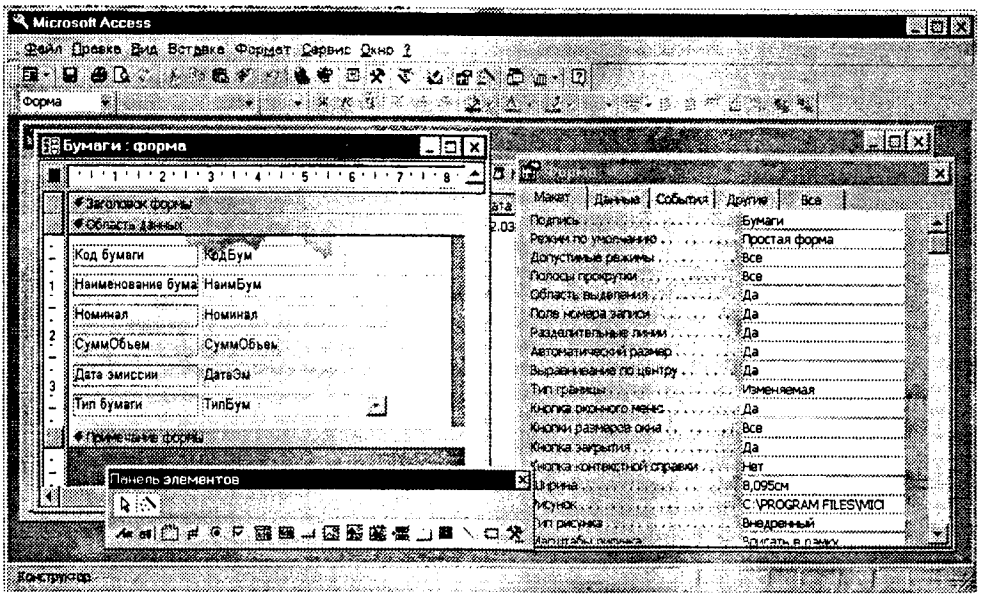


Рис. 7.18. Форма Бумаги в режиме конструктора

В режиме Конструктор явно видна структура формы. Она состоит из трех частей: Заголовок формы, Область данных и Примечание формы. Как нетрудно догадаться, такая структура в первую очередь ориентирована на возможности представления таблично организованных данных. Заметим, что как сама форма, так и ее разделы также рассматриваются как элементы управления, обладающие некоторыми настраиваемыми наборами свойств.

В качестве иллюстрации возможностей конструктора по изменению интерфейса ввода/вывода проведем следующие манипуляции над формой Бумаги:

1. *Удалим фоновый рисунок*: очистим свойство Рисунок, когда текущим выбранным элементом является вся форма.
2. *Изменим цвет фона*: выберем элемент ОбластьДанных и изменим у нее атрибут Цвет фона (рис. 7.19).
3. *Изменим внешний вид полей*: выделим группу полей (поля выбираются с помощью мыши при нажатой клавише Shift) и в окне свойств изменим значение атрибута Оформление на Утопленное.
4. *Отредактируем подписи полей и несколько изменим их расположение друг относительно друга:* для этого достаточно воспользоваться возможностями визуального редактирования элементов.

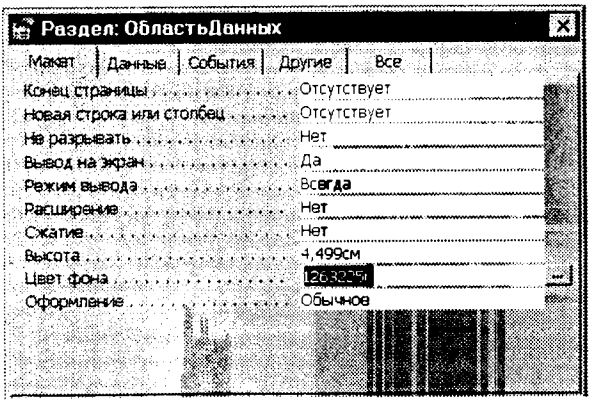


Рис. 7.19. Окно свойств элемента управления

5. *Добавим разделительную линию после поля* НаимБум (наименование бумаги): для этого следует воспользоваться элементом Линия.
6. *Добавим кнопку завершения работы с формой*: в большинстве ситуаций эту и подобные операции проще и удобнее делать в режиме мастера (нажата соответствующая кнопка на панели Элементы управления). В этом случае от пользователя требуется лишь ввести минимальное количество параметров для добавляемого программного компонента. Добавленную кнопку поместим в область Примечания формы.

В результате отредактированная форма Бумаги примет вид, показанный на рис 7.20.

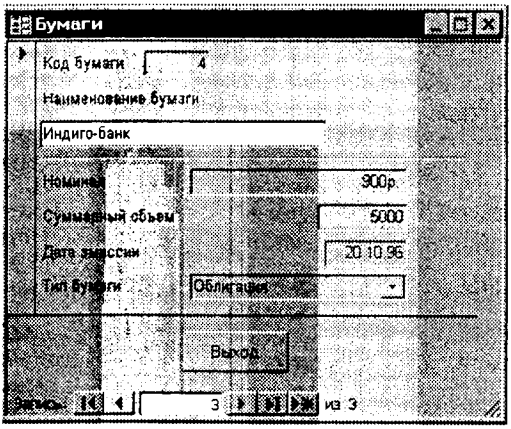


Рис. 7.20. Форма Бумаги после редактирования

Пример организации ввода/вывода данных в таблицу Бумаги с помощью одноименной формы носит в некотором смысле вырожденный характер: в нем структура полей в форме однозначно соответствует их структуре в таблице. Однако, как правило, при создании реальных приложений приходится решать задачу управления данными, находящимися в системе взаимосвязанных таблиц, из единой формы.

В качестве примера рассмотрим задачу построения формы, в которой для каждой ценной бумаги одновременно выводится информация по заявкам на ее покупку и продажу. Ее внешний вид приведен на рис. 7.21. Верхняя (заголовочная) часть формы соответствует текущей строке таблицы Бумаги и меняется при переходе от записи к записи, который может производиться с помощью стрелок, расположенных в нижней части окна. Одновременно должны меняться строки таблиц Заявки на продажу и Заявки на покупку, в которые выводится только информация, относящаяся к текущей бумаге.

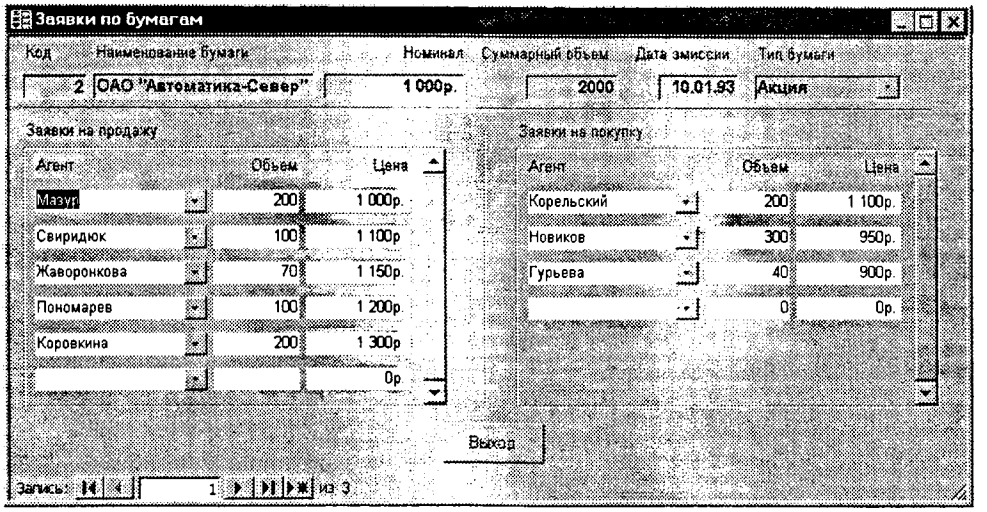


Рис. 7.21. Пример многотабличной формы

Рассмотрим более подробно те средства Access, с помощью которых может быть получен такой результат. На рис. 7.21 приведена так называемая сложная, или составная форма (Заявки по бумагам). Процесс ее создания состоит из двух принципиальных этапов:

- создание основной (главной) формы. Для этого осуществляются действия, аналогичные тем, которые выполнялись при создании формы Бумаги;
- создание подчиненных форм. Для этого в созданную главную форму добавляется элемент управления Подчиненная форма. При создании подчиненной формы в Access существует две принципиальные возможности:
  - создать новую форму на базе некоторой таблицы или запроса;
  - воспользоваться уже существующей формой, сделав ее подчиненной.

В данном случае созданы две новые подчиненные формы. Причем созданы они на базе специальных запросов. Такое решение позволяет выделить по отдельности из общей таблицы Заявки записи с заявками на продажу и на покупку. В частности, запрос ЗаявПрод, возвращающий выборку из заявок на продажу ценных бумаг, имеет структуру, показанную на рис. 7.22. В качестве преимуществ такого подхо-

да к организации источника данных для подчиненной формы следует отметить следующие моменты:

- во вспомогательном запросе достаточно просто обработать условие, идентифицирующее тип заявки (если объем заявки меньше нуля, то это заявка на продажу). Более того, для конечного пользователя в качестве объема заявки вместо отрицательных величин выводятся выглядящие более естественно положительные значения `-1*[ОбъемЗаявки]`;
- данные выводятся отсортированными по возрастанию предлагаемых цен, что несомненно упрощает процесс работы с ними в экранной форме.

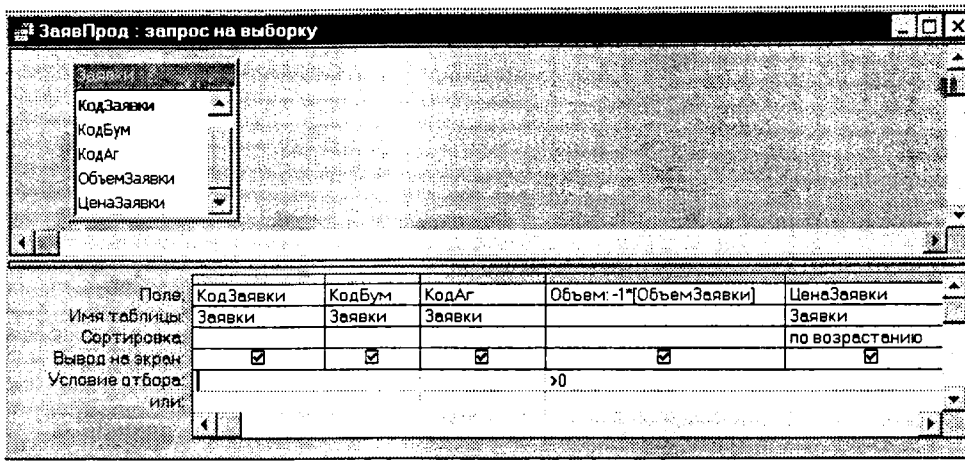


Рис. 7.22. Структура запроса ЗаявПрод (заявки на продажу)

Запрос, возвращающий записи с заявками на покупку, создается аналогично с учетом модификации условия отбор.

Наиболее существенным моментом в процессе внедрения подчиненной формы в главную является правильное задание условия связи между ними. Во многих случаях с этим корректно справляются программные надстройки мастеров. При этом они используют информацию из схемы данных и описаний структуры таблиц. В то же время, не следует забывать и о возможностях изменения условий связи между ведущей и подчиненной формами в ручном режиме. Для этого необходимо изменить атрибуты в элементе управления Подчиненная форма, находясь в режиме Конструктор (рис. 7.23).

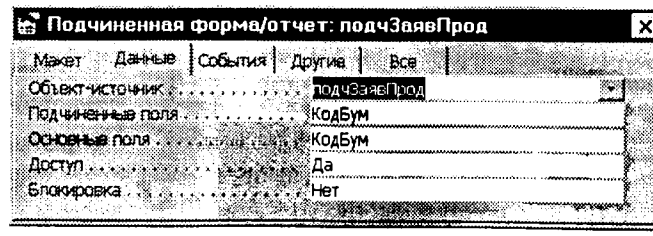


Рис. 7.23. Задание условий связи для подчиненной формы

7.2.8. Конструирование отчетов

Неотъемлемой функцией любых программных систем, так или иначе связанных с обработкой данных, является представление отчетов по хранимой информации. Под отчетом традиционно понимается специальным образом структурированное представление хранимых данных, выводимое (как правило) на бумажный носитель. Перечислим принципиальные отличия отчетов от экранных форм, обусловившие выделение их в отдельный программный объект СУБД Access:

- во-первых, отчеты являются исключительно средством вывода информации;
- во-вторых, организация данных в отчетах предполагает возможность их сложного, многоуровневого структурирования;
- в-третьих, структура информации, выводимой в отчете, должна быть согласована со структурой носителя. Например, разбиение отчета на страницы предполагает организацию вывода регулярных элементов в начале и конце каждого листа (колонтитулов), дублирование шапок таблиц и т. д. Также на внешний вид отчета значительное влияние оказывают параметры конкретного печатающего устройства, которое будет использовано для его вывода.

В то же время, к числу важных достоинств Access относится то, что идеология работы как с экранными формами, так и с отчетами максимально универсализирована. В частности, интерфейс режима конструирования макета отчета аналогичен режиму конструктора для экранных форм.

Рассмотрим способы решения задач разработки отчетов, которые могут возникать в рамках описываемой нами программной системы управления торгами ценными бумагами. Простейшие отчеты, которые, скорее всего, будут необходимы пользователям системы, — это распечатанные списки бумаг и агентов. Для их создания можно воспользоваться надстройками Автоотчет в столбец или Автотчет ленточный. На рис. 7.24 показан макет отчета по агентам, созданный в режиме Автоотчет ленточный.

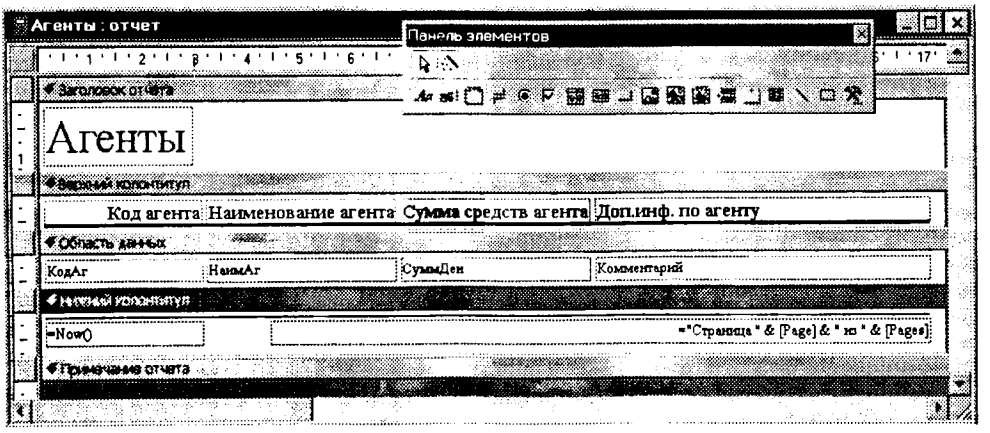


Рис. 7.24. Отчет по агентам в режиме конструктора

Из рис. 7.24 видно, что в процессе конструирования в макет отчета могут быть добавлены те же самые управляющие элементы, что и при конструировании макета экранной формы. В то же время следует отметить, что структура отчета как объекта базы данных имеет свою специфику. Во-первых, она определяется уровнями группировки данных, выводимых в отчет, а во-вторых, содержит секции, соответствующие регулярным элементам, помещаемым в начале и конце каждого листа, — верхнему и нижнему колонтитулам. Для задания уровней группировки данных используется функция меню Вид ▸ Сортировка и группировка или же одноименная пиктограмма на панели инструментов Конструктор отчетов.

При работе с отчетами активно используются (это видно из рис. 7.24) встроенные переменные [Page] и [Pages], возвращающие номер текущей страницы отчета и общее количество страниц в нем, а также функция Now(), определяющая текущую дату и время по системному календарю.

Остановимся теперь на более сложном примере. Поставим задачу построить отчет, выводящий сведения о спросе и предложении по ценным бумагам с учетом их типа, то есть записи должны быть структурированы по следующим уровням:

- все бумаги;
- тип бумаги;
- агент;
- предложения агента по данной бумаге.

Также по каждому из уровней желательно предусмотреть вывод промежуточных итогов (или же соответствующих средних значений).

Информация для данного отчета (назовем его РаспределЗаявок) должна браться из различных таблиц, поэтому в качестве источника данных для него целесообразно использовать специально построенный запрос. Для наглядности приведем SQL-выражение, соответствующее данному запросу:

```
SELECT
        IIf([ТипБум]="1","Акции","Облигации") AS Тип,
        Заявки.КодБум,
        Бумаги.НаимБум,
        Бумаги.Номинал,
        Агенты.НаимАг,
        IIf([ОбъемЗаявки]<0,-1*[ОбъемЗаявки],0) AS
              ОбъемПродажи,
        IIf([ОбъемЗаявки]<0,[ЦенаЗаявки],0) AS ЦенаПродажи,
        IIf([ОбъемЗаявки]>0,[ОбъемЗаявки],0) AS ОбъемПокупки,
        IIf([ОбъемЗаявки]>0,[ЦенаЗаявки],0) AS ЦенаПокупки
   FROM Бумаги
   INNER JOIN (Агенты INNER JOIN Заявки ON Агенты.КодАг = Заявки.КодАг) ON
Бумаги.КодБум = Заявки.КодБум
   ORDER BY IIf([ТипБум]="1","Акции","Облигации"),
        Бумаги.НаимБум;
```

На основе построенного запроса можно перейти к разработке отчета. На начальном этапе представляется рациональным воспользоваться услугами мастера отчетов.

Он в режиме диалога с пользователем позволяет создать походящую «заготовку», избавляя нас от многих рутинных операций, например таких, как добавление полей и подписей к ним.

Далее полученный макет вручную «доводится» до желаемого вида в режиме Конструктор. Процесс редактирования макета отчета РаспределЗаявок показан на рис. 7.25.

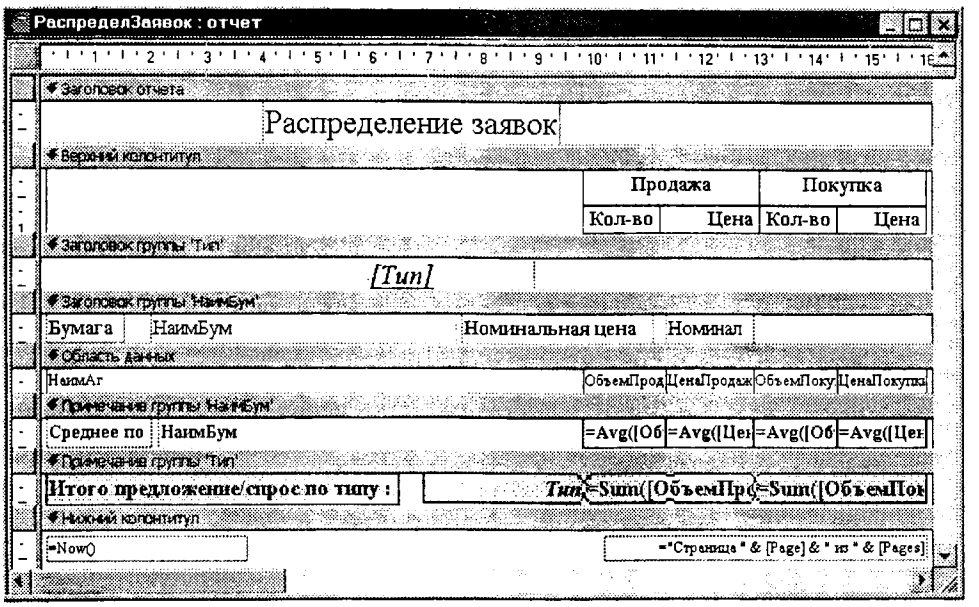


Рис. 7.25. Разработка макета отчета РаспределЗаявок

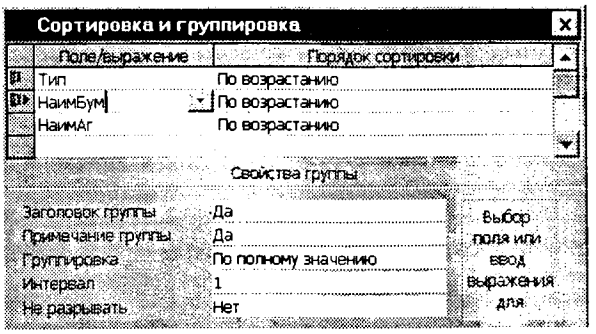


Рис. 7.26. Задание уровней группировки и сортировки

Важным этапом при создании многоуровневого отчета является задание уровней группировки выводимых данных. Это делается в окне, показанном на рис. 7.26, которое вызывается из меню Вид ▸ Сортировка и группировка. Для каждого из заданных уровней группировки данных могут быть определены раздел типа Заголовок, выводимый в начале каждой группы, и раздел типа Примечание, формируемый, когда группа заканчивается.

Задачи получения средних и итоговых значений по группам данных решаются с помощью встроенных функций Sum() и Avg(). Например, для получения среднего значения цены продажи бумаги в соответствующем элементе управления свойство Данные содержится строка =Avg([ОбъемПродажи]), а для определения итогового спроса используется формула =Sum([ОбъемПродажи]*[ЦенаПродажи]).

Распределение заявок

			Продажа		Покупка	
			Кол-во	Цена	Кол-во	Цена
Акции						
Бумага	Красная шапочка	Номинальная цена	1 500р.			
Сапегина					20	300
Сиротинин					100	500
Среднее по	Красная шапочка				60	400
Бумага	ОАО "Автоматика-Север"	Номинальная цена	1 000р.			
Гурьева					40	900
Жаворонкова			70	1150		
Корельский					200	1100
Коровкина			200	1300		
Мазур			200	1000		
Новиков					300	950
Пономарев			100	1200		
Сиротинин			100	1100		
Среднее по	ОАО "Автоматика-Север"		84	719	68	369
Итого предложение/спрос по типу:		*Акции*		770 500		597 000
Облигации						
Бумага	Индиго-банк	Номинальная цена	900р.			
Гурьева			100	200		
Жаворонкова					30	230
Коровкина					50	400
Новиков					50	500
Сапегина			80	150		
Среднее по	Индиго-банк		36	70	26	226
Итого предложение/спрос по типу:		*Облигации*		32 000		51 900

Рис. 7.27. Вид отчета РаспределЗаявок в режиме предварительного просмотра

На рис. 7.27 показан результат работы построенного отчета в режиме предварительного просмотра, то есть в виде, максимально приближенном к виду готового печатного документа.

7.2.9. Средства макропрограммирования в MS Access

Access, как и любая другая развитая программная система, обладает средствами разработки программных приложений, ориентированных на конечных пользователей. Эти средства базируются на инструментах двух типов: макросах и модулях.

Само понятие *макроса* подразумевает наличие набора некоторых стандартных команд системы, или макрокоманд (допустим, таких, как открытие формы, выполнение запроса, вывод отчета), из которых и конструируется сам макрос.

Макрос может быть как собственно макросом, состоящим из последовательности макрокоманд, так и *группой макросов*. Группой макросов называют их набор, со-

храняемый под общим именем[1]. В некоторых случаях для решения, должна ли в запущенном макросе выполняться определенная макрокоманда, может применяться условное выражение.

Особый интерес вызывает механизм вызова макросов в Access. Для этого существует две принципиальных возможности:

- вызов макроса по команде пользователя (либо непосредственно из раздела Макросы главного окна базы данных, либо с помощью меню или панели инструментов, с которыми он также может быть ассоциирован);
- вызов макроса по некоторому системному событию (открытие или закрытие формы, изменение управляющего элемента и т. п.).

Весьма полезной представляется возможность организовать автоматическое выполнение ряда действий при открытии базы данных. Для этого они должны быть описаны в специальном макросе с именем Autoexec.

Возможности применения макросов при работе в среде СУБД Access можно наглядно продемонстрировать на следующем примере. Предположим, что в ранее созданную форму Бумаги мы хотим добавить процедуру дополнительного контроля вводимых значений дат эмиссии ценных бумаг, которая должна будет выдавать предупреждающее сообщение, если вводится слишком «ранняя» дата. Допустим, что к таковым относятся даты, предшествующие 1 января 1991 года.

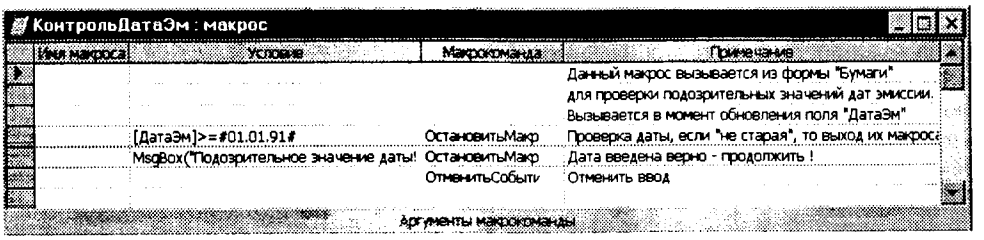


Рис. 7.28. Создание макроса

Технически решение представляется удобным реализовать в виде макроса, вызываемого по событию «*до обновления*», ассоциированному с полем ДатаЭм в форме Бумаги.

На рис. 7.28 показан процесс разработки данного макроса (ему дано имя КонтрольДатаЭм). Из него видно, что макрос содержит три макрокоманды:

- Первая, ОстановитьМакрос — прерывает работу, если введена дата более поздняя, чем 1 января 1991 года.
- Вторая, ОстановитьМакрос — выполняется, если пользователь считает, что несмотря на сделанное предупреждение введенная дата является верной. Для вывода предупреждения используется встроенная функция MsgBox.
- Третья — отменяет событие ввода данных, если они после предупреждения признаются ошибочными.

[1] Иногда группу макросов опять-таки сокращенно называют просто макросом, что вносит некоторую терминологическую путаницу.

Хорошим стилем разработки макросов является снабжение их комментариями, располагаемыми в соответствующей колонке.

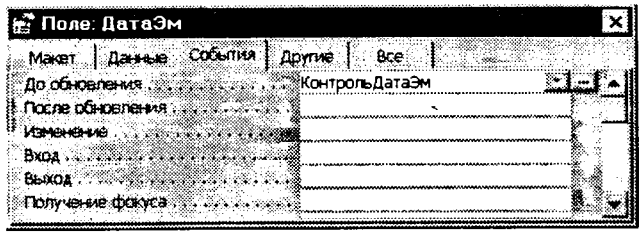


Рис. 7.29. «Привязка» макроса к событию

На рис. 7.29 показана привязка разработанного макроса к событию *«до обновления»* поля ДатаЭм формы Бумаги.

7.2.10. Разработка программных приложений для MS Access

Модули, в отличие от макросов, являются более тонким и мощным средством создания программных расширений в среде Access, максимально приближающимся по своим функциональным возможностям к таким профессиональным инструментам, как Delphi, Visual Basic или Power Builder. Одновременно применение модулей требует от пользователя навыков и квалификации программиста, а также знания основных принципов объектно-ориентированного программирования.

Для программирования в Access используется процедурный язык Visual Basic для приложений (VBA — Visual Basic for Applications) с добавлением объектных расширений и элементов SQL. Сам процесс создания программных расширений в среде Access предполагает активное использование технологии *объектно-ориентированного программирования (ООП)*. В основе ООП лежит идея «упакованной функциональности», в соответствии с которой программа строится из фундаментальных сущностей, называемых объектами. Каждый из объектов характеризуется набором свойств (англ. — *property*) и операций, которые он может выполнять (англ.— *method*). Реализация взаимодействий между объектами ложится на исполняющую среду того средства разработки, на котором пишется программа, и поэтому работа программиста в рамках технологии ООП сводится к созданию объектов, описанию их свойств и реакций на те или иные внешние события.

Фундаментальным понятием ООП является *класс*. Класс — это шаблон, на основе которого может быть создан конкретный программный объект. Созданный объект в таком случае становится *экземпляром класса*. К основополагающим принципам ООП относятся:

- *инкапсуляция* — объединение свойств и действий, присущих объекту, в едином пакете и сокрытие подробностей их реализации от окружающего мира. Это означает, что пользовательский доступ к объекту допускается только через его свойства и методы;

- *наследование* — предусматривает создание новых классов на базе существующих, что дает возможность классу-потомку иметь (наследовать) все свойства класса-родителя;
- *полиморфизм* — (от греч. «многоликость») означает, что порожденные объекты обладают информацией о том, какие методы они должны использовать в зависимости от того, где они находятся в цепочке наследования;
- *модульность* — объекты заключают в себе полное определение их характеристик, никакие определения методов и свойств объекта не должны располагаться вне его, что делает возможным свободное копирование и внедрение одного объекта в другие.

Многие программные объекты в Access совпадают с физическими объектами базы данных, такими как таблицы, формы, отчеты. Для названия составных объектов, которые включают в себя совокупность более простых объектов, используется термин *семейство*. Например, объект отчет входит в семейство отчеты. Помимо «видимых» объектов существует и большое количество «скрытых» объектов, управлять которыми можно только из программных приложений.

В Access существуют два типа модулей: стандартные и модули класса.

Стандартные модули содержат процедуры и функции, которые могут быть вызваны из любого окна базы данных. Как правило, такие модули содержат программный код универсального характера, предназначенный для применения в различных местах текущего приложения или даже в различных приложениях.

Модули класса используются для создания новых классов объектов. При создании конкретного объекта, являющегося экземпляром такого класса, любые процедуры, определенные в модуле, становятся свойствами и методами этого объекта.

Модули форм и модули отчетов являются модулями класса, связанными с определенной формой или отчетом. Заметим, что в ранних версиях Access они являлись единственно возможным инструментом объектно-ориентированного программирования. Эти модули содержат процедуры обработки событий, запускаемых в ответ на их возникновение в форме или отчете. Процедуры обработки событий используются для управления поведением формы или отчета и их откликом на события, например такие, как нажатие кнопки.

Важнейшей областью применения объектно-ориентированного программирования в Access является программирование доступа к данным. Для решения данной задачи фирмой Microsoft был разработан специальный интерфейс — DAO (Data Access Objects). DAO — это набор объектных классов, которые моделируют структуру реляционной базы данных. Они обеспечивают свойства и методы, которые позволяют выполнять такие операции, как создание базы данных, определение таблиц и индексов, задание связей между таблицами, формирование запросов и отчетов и т. п. Существенным достоинством объектной модели DAO является ее универсальный характер: она доступна для большинства средств разработки программного обеспечения, поддерживаемых Microsoft, в частности, для Visual Basic.

Классы объектов доступа к данным организованы по иерархической схеме. На ее вершине находится объект `DbEngine`, представляющий собой ядро базы данных. Далее следуют объекты, отвечающие за управление сеансами доступа пользовате-

ля к данным, — Workspace (от англ. «рабочая область»). Каждая рабочая область включает один или несколько объектов класса база данных — Database, а они, в свою очередь, содержат семейства объектов таблиц (TableDef), запросов (QueryDef), наборов записей (RecordSet) и т. д.

В заключение раздела, посвященного модулям, отметим, что мы сознательно не затрагиваем собственно вопросы теории и практики создания программ на VBA в среде Access, так как они являются весьма обширными. В случае необходимости читатели могут ознакомиться с ними в специальных профессиональных изданиях и руководствах.

7.3. Организация взаимодействия между системами управления данными

7.3.1. Проблема форматно независимого доступа к данным и технология ODBC

Процесс разработки и развития любой СУБД неизбежно приводит к необходимости решать проблему взаимодействия с данными, созданными и управляемыми в рамках других программных систем, или, как еще говорят, к проблеме доступа к **внешним источникам данных**. Это, в свою очередь, определяет принципиальное требование, которому должны удовлетворять прикладные СУБД: *программные процедуры обработки информации, создаваемые в рамках СУБД, должны быть максимально независимыми от формата хранимых данных.*

Выполнение этого принципа позволяет:

- во-первых, с наименьшими затратами осуществлять переход от одной СУБД к другой, потребность в чем, допустим, возникает при масштабировании ранее созданного программного обеспечения для предприятий и фирм качественно иного размера;
- во-вторых, успешно решать задачи интеграции двух и более независимых программных систем.

Важнейшим инструментом форматно независимого доступа к данным из программ стала технология ODBC (Open Data Base Connectivity[1]), созданная фирмой Microsoft. Ее принципиальная схема изображена на рис. 7.30.

Как следует из рис. 7.30, в рамках ODBC:

- **программное приложение** непосредственно взаимодействует с диспетчером драйверов, посылая ему ODBC-вызовы;
- **диспетчер драйверов** отвечает за динамическую загрузку нужного ODBC-драйвера, через который обращается к СУБД (серверу баз данных);

[1] Дословный перевод с англ. — открытый доступ к базам данных.

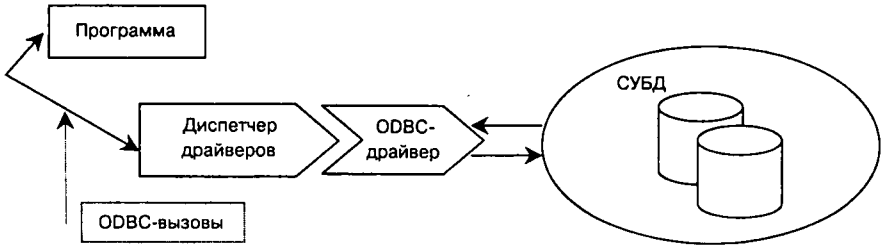


Рис. 7.30. Принципиальная схема технологии ODBC

- **ODBC-драйвер** выполняет все вызовы ODBC-функций, «переводит» их на язык источника данных;
- **СУБД** хранит и выводит данные в ответ на запросы со стороны ODBC-драйвера (или же возвращает код ошибки).

В настоящее время в состав подавляющего большинства систем управления данными входят соответствующие ODBC-драйверы.

Таким образом, при работе с базой данной через ODBC-драйвер она выступает как некоторый виртуальный источник данных, которым можно управлять с помощью SQL-подобных команд.

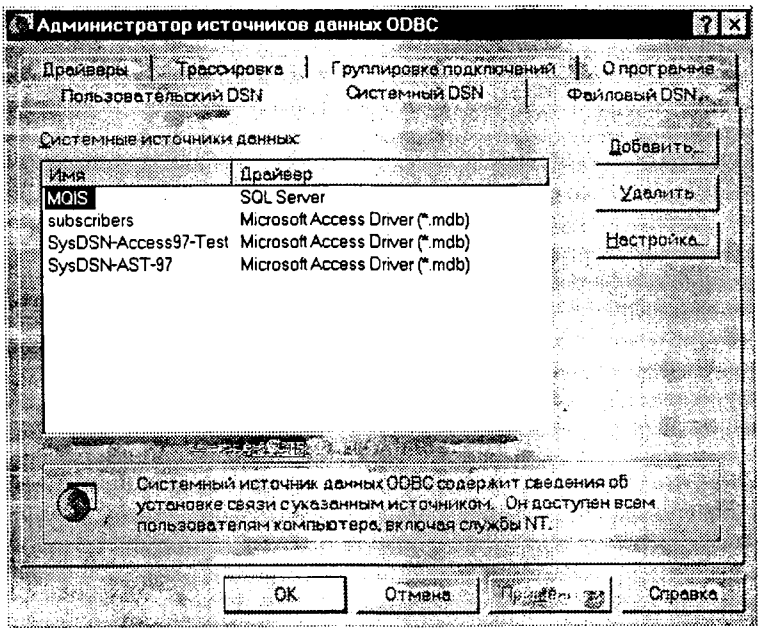


Рис. 7.31. Окно Администратора источников данных ODBC

Задание ODBC-источника данных (DSN — data source name) является действием, которое осуществляется средствами операционной системы, управляющей компьютером. В частности, в операционных средах Windows 95/Windows 98 для этого в

Панели управления предусмотрен пункт Источники данных ODBC (32 разр), из которого вызывается Администратор источников данных ODBC. С его помощью могут быть заданы:

- пользовательский DSN — источник данных, доступный только текущему пользователю на текущем компьютере;
- файловый DSN — источник данных, которые могут применять совместно различные пользователи, у которых установлены одинаковые ODBC-драйверы;
- системный DSN — источник данных, доступный всем пользователям и службам текущего компьютера.

Окно Администратор источников данных ODBC показано на рис. 7.31.

7.3.2. Доступ из MS Access к источникам данных в формате других программных приложений

В MS Access предусмотрены две принципиальные возможности работы с внешними данными. Это импорт данных и связь с внешними таблицами данных. Оба режима доступны из меню главного окна базы данных: Файл ▸ Внешние данные.

В случае *импорта* происходит создание дубликата внешних данных во вновь создаваемой таблице. Среди преимуществ такого решения могут быть названы:

- доступность всего арсенала средств СУБД Access при манипуляциях импортированными данными;
- высокое быстродействие при обращении к ним;
- независимость от исходного источника данных.

Однако, приобретая указанные преимущества, мы одновременно получаем и потенциальные проблемы, связанные с поддержанием актуальности и соответствия друг другу двух параллельных копий одной и той же информации. Очень часто подобные проблемы оказываются неразрешимыми.

Если актуальность данных является для нас критичным фактором, то необходимо использовать другой способ работы с внешними данными — *связь*. В этом случае в базе данных добавляется лишь ссылка на внешние источники данных и работа с ними происходит с помощью специальных драйверов. В поставку MS Access традиционно входят драйверы для работы с данными, созданными в форматах Paradox, Excel, dBase, FoxPRO, а также в текстовом (*.txt) и гипертекстовом (*.htm) форматах. Базы данных Paradox, Excel, dBase, FoxPRO и некоторых других форматов также называют базами данных с индексно-последовательной организацией (англ. — ISAM — Indexed Sequential Access Method). Специфические ISAM-драйверы, учитывающие конкретные особенности перечисленных форматов организации данных, как правило, обеспечивают высокую эффективность и быстродействие при работе с ними. Одновременно в Access существует возможность работы с обширным множеством универсальных источников данных, для которых установлены ODBC-драйверы. Для этого при указании типа файла, с которым устанавливается связь, необходимо выбрать Базы данных ODBC() (рис. 7.32).

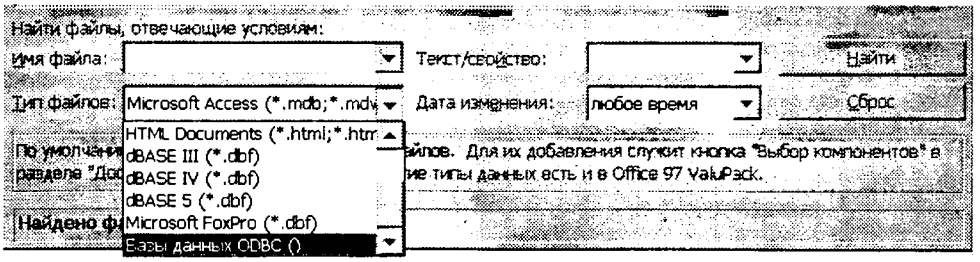


Рис. 7.32. Выбор типа внешнего источника данных

«Платой» за применение технологии связывания с внешними данными являются ограничения возможностей по управлению структурой добавляемых таблиц, а также зависимость от состояния самого внешнего источника, к которому мы подключаемся.

7.3.3. Технологические решения по организации доступа к данным

Рассмотрим чуть подробнее архитектуру доступа к данным в Access. Схематично она представлена на рис. 7.33. В представленной схеме блок *пользовательского интерфейса* олицетворяет видимую часть СУБД, то есть то, с чем пользователь взаимодействует непосредственно (формы, отчеты и другие объекты). Под *хранилищем данных* понимаются файл (файлы), содержащие таблицы данных (например, в Access это mdb-файлы). Хранилище — это некоторый пассивный элемент, в нем данные просто содержатся. Осуществлять манипуляции с ними — это задача *процессора базы данных* (или, как еще говорят, ядра базы данных). Он транслирует команды приложения в физические операции, непосредственно меняющие файл (файлы) хранилища данных. Основным достоинством описанной схемы является независимость приложения от типа базы данных, к которой она обращается: будут ли это данные во внутреннем формате Access или данные какой-то другой структуры — в приложении используются одни и те же объекты и методы доступа к ним.

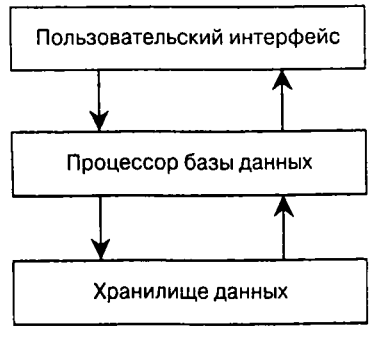


Рис. 7.33. Архитектура доступа к данным в Access

В СУБД MS Access используется процессор, получивший название Jet (Joint Engine Technology). Он реализован в виде набора файлов динамически компонуемых библиотек (DLL), которые связываются с прикладной программой Access в период ее выполнения. В состав процессора Jet входят процессор запросов SQL и процессор обработки результатов, возвращаемых этими запросами.

Рассмотренная ранее модель объектного интерфейса доступа к данным DAO представляет собой программную надстройку над процессором Jet. Jet также реализует описанные в 7.3.1 возможности по доступу к внешним данным в формате ISAM и источникам данных ODBC.

Для работы СУБД MS Access 97 был использован процессор Jet версии 3.5 для 32-разрядных приложений. Среди принципиальных преимуществ новой версии могут быть названы:

- ODBCDirect — альтернативный режим DAO, который предоставляет возможности прямого обращения к источникам данных ODBC в обход ядра Jet. Это позволяет в некоторых случаях оптимизировать процесс работы с данными за счет использования специфических характеристик удаленных ODBC-источников;
- для баз данных, управляемых процессором Jet, определены новые объекты, свойства и методы, позволяющие использовать новые возможности частичной репликации[1].

Также следует отметить, что в Jet реализована технология Rushmore — специальная методика управления запросами, которая позволяет очень эффективно отбирать наборы записей при использовании в их критериях определенных типов выражений.

7.4. Организация многопользовательского доступа к данным

7.4.1. Проблема многопользовательского доступа и параллельной обработки данных в автоматизированных информационных системах

Естественным следствием развития СУБД является проблема организации совместной работы нескольких пользователей с одной и той же совокупностью данных, или, кратко, *проблемы многопользовательского доступа к данным*.

Остановимся более подробно на основных аспектах этой проблемы. Прежде всего ситуация разделения одной и той же совокупности данных между несколькими пользователями может приводить к возникновению конфликтов (попытка единовременного изменения одной и той же записи, совпадение операций чтения и удаления информации и т. д.). Отдельное место при работе с СУБД занимают

[1] Репликация баз данных — создание особых копий (реплик) базы данных. Это позволяет пользователям, работающим за различными компьютерами, осуществлять одновременный доступ к собственным копиям и обмениваться или синхронизировать внесенные изменения.

вопросы предотвращения коллизий, которые могут возникнуть в случае несогласованных изменений структуры таблиц, форм или отчетов одним пользователем, когда с ними работают другие.

С точки зрения организации совместного доступа к данным со стороны нескольких пользователей режимы работы с ними делятся на *режим монопольного (эксклюзивного) доступа* и *режим общего (разделенного) доступа*.

Режим монопольного доступа к базе данных предусматривает, что только один из пользователей (программных процессов) может работать с ней, а возможность ее открытия другими пользователями (процессами) блокируется. Открытие базы данных в монопольном режиме, как правило, используется для выполнения операций по изменению структуры таблиц и связей между ними, экспорта большого количества информации, выполнения служебных операций с данными (сохранение, восстановление, сжатие) и т. п.

Соответственно, в *режиме разделенного доступа* сразу несколько пользователей могут работать с базой данных. Для предотвращения возможных конфликтов при попытках со стороны различных пользователей изменить одни и те же записи в СУБД используется механизм *блокировок*. Блокировка того или иного объекта в случае работы с ним какого-либо пользователя означает предотвращение любых других попыток изменить этот объект, но при этом сохраняется возможность его чтения. Таким образом, механизм блокировок предоставляет более гибкие возможности для манипуляций с данными по сравнению с режимом монопольного доступа.

Для различных СУБД конкретные технические решения по реализации аппарата блокировок существенно различаются. В MS Access, в частности, при изменении записи одним пользователем по умолчанию происходит ее автоматическая блокировка вплоть до момента завершения операции. При создании форм, отчетов или запросов в Access предусмотрены возможности задания параметров режима блокировки. На рис. 7.34 показан процесс изменения свойства Блокировка записей для формы.

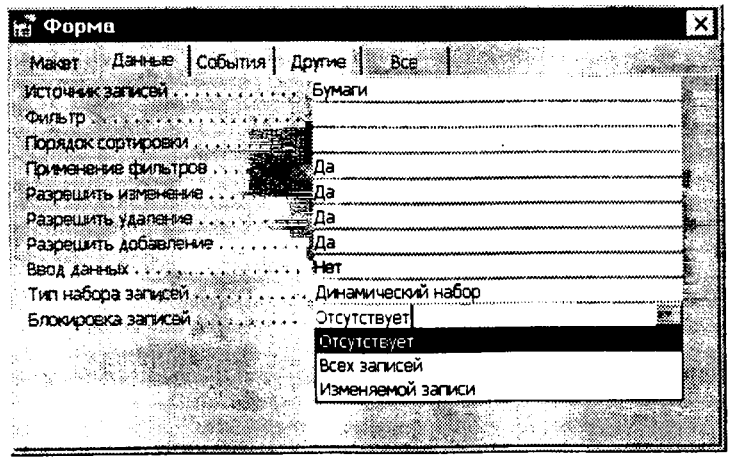


Рис. 7.34. Задание режима блокировки для данных, доступных из формы

Как видно из рисунка, свойство Блокировка записей может принимать значения:

- **Отсутствует** — допускается одновременное изменение записей со стороны нескольких пользователей. При этом если два пользователя пытаются сохранить произведенные изменения в одной и той же записи, то второму пользователю выводится предупреждающее сообщение, на основе которого он может либо отказаться от дальнейших действий, либо заместить изменения, сделанные первым пользователем, сохранив собственный вариант. Очевидно, что в таком режиме сохраняется максимальная свобода действий пользователей, «платой» за которую являются возможные конфликты ввиду несогласованности их действий.
- **Всех записей** — происходит блокировка всех записей в источнике данных при его открытии одним из пользователей, в результате чего он может беспрепятственно изменять его. Другие пользователи имеют доступ только на чтение (просмотр).
- **Изменяемой записи** — один из пользователей получает доступ на изменение нужной ему записи, а другие пользователи могут только читать содержащиеся в ней данные. Данный режим накладывает минимальные ограничения на совместную работу. Следует добавить, что технически в Access блокируются не записи как таковые, а так называемые страницы — блоки файла базы данных размером 2048 байт, содержащие нужные записи.

Отмена блокировки в Access происходит тогда, когда пользователь, ранее блокировавший запись, либо сохранит произведенные изменения, либо откажется от них. Для того чтобы изменения, производимые одним пользователем, становились видны другим, через определенные интервалы времени предусмотрено автоматическое обновление содержания таблиц, форм и отчетов. Значение периода обновления задается из меню Сервис ▸ Параметры, вкладка Другие, поле Период обновления.

Другим существенным вопросом, который должен быть решен для обеспечения нормального функционирования многопользовательских СУБД, является организация системы администрирования данных. Среди задач администрирования могут быть названы:

- создание системы пользователей и разделение прав доступа различных пользователей к объектам СУБД;
- организация и поддержание системы резервного хранения информации и ее восстановления в случае программных и аппаратных сбоев;
- мониторинг программных и аппаратных ресурсов, задействованных для обеспечения работы СУБД, и принятие на его основе решений по оптимизации их использования.

Некоторые вопросы, связанные с организацией системы пользователей СУБД, будут рассмотрены в 7.4.3.

Первые многопользовательские СУБД имели централизованную архитектуру и базировались на больших компьютерах или мини-ЭВМ. Рабочие места пользова-

телей располагались на терминалах, подключенных к центральному компьютеру, на котором выполнялись все процессы по манипуляции с данными. Однако с распространением персональных компьютеров особую актуальность приобрели СУБД, реализующие технологии распределенной обработки данных, то есть такие технологии, которые позволяют вести одновременную работу с нескольких относительно ограниченных по аппаратным возможностям машин, объединенных в сеть[1]. В этом случае одна часть функций СУБД выполняется на компьютере-клиенте, а другая — на компьютере-сервере, причем их взаимодействие осуществляется через некоторый согласованный протокол.

Исторически первая технология распределенной работы с данными получила название *файл-сервер (FS-модель)*. В ее рамках предполагается, что один из компьютеров в сети является файловым сервером и предоставляет свои ресурсы по обработке файлов другим компьютерам, на нем также располагается хранилище данных. На других компьютерах имеется прикладное программное обеспечение, реализующее функции пользовательского интерфейса доступа к данным, и копия процессора базы данных (СУБД). Всякий раз, когда прикладная программа обращается к базе, процессор данных обращается к файловому серверу. В ответ файловый сервер направляет по сети требуемый блок данных, получив который, СУБД осуществляет действия, декларированные в прикладной программе. Протокол обмена между серверным и клиентскими компьютерами представляет собой набор низкоуровневых вызовов, обеспечивающих интерфейсному приложению доступ к файловой системе сервера.

Технологические недостатки FS-модели вытекают из внутренне присущих ей ограничений. Среди них в первую очередь следует назвать:

- высокий сетевой трафик, обусловливаемый необходимостью передавать большое количество файловых блоков от сервера к приложениям;
- ограниченный набор допустимых действий по обработке данных на файловом сервере, который не способен «понимать» внутреннюю логическую структуру базы данных и воспринимает эту базу так же, как и любой другой файл;
- отсутствие надежных средств обеспечения безопасности работы с данными — допускается защита только на уровне функций сетевой операционной системы.

Перечисленные проблемы определяют то, что СУБД, основанные на технологии файл-сервер, могут применяться только в очень ограниченных масштабах. Например, при создании коллективных информационных систем, рассчитанных на небольшое количество пользователей и ограниченные информационные потоки. Одновременно следует отметить, что FS-модель положена в основу архитектур подавляющего большинства настольных СУБД, таких как FoxPRO, Clipper, Clarion, Paradox, Access, завоевавших широкую популярность среди отечественных разработчиков.

---

[1] Вопросы, связанные с сетями персональных компьютеров, подробно рассматриваются в главе 10.

7.4.2. Основные направления развития технологии клиент-сервер

Работа по преодолению недостатков, органически присущих системам файл-сервер, привела к появлению более прогрессивной технологии, получившей название *клиент—сервер*. Ее принципиальные отличия показаны на рис. 7.35.

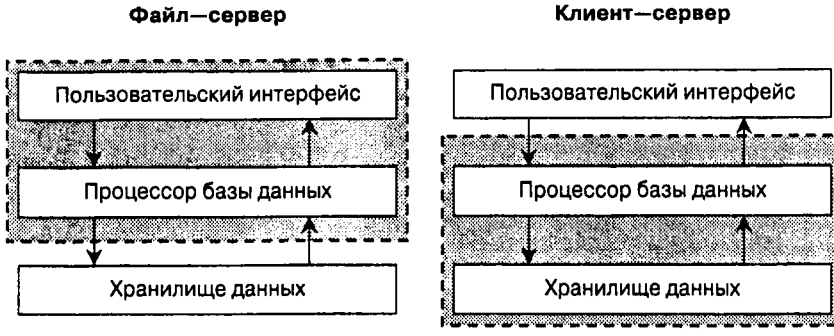


Рис. 7.35. Принципиальная схема технологий файл—сервер и клиент—сервер

В системе клиент-сервер процессор базы данных размещается на центральном сервере вместе с хранилищем данных. Он может обслуживать одновременно несколько клиентских приложений, управляя хранилищем и возвращая запрошенную информацию после обработки запросившему ее локальному приложению.

К настоящему моменту можно назвать ряд этапов, которые технология клиент—сервер прошла в своем развитии: RDA-модель, DBS-модель и AS-модель.

В *RDA-модели* клиентское приложение направляет запросы (как правило, на языке SQL) к информационным ресурсам сервера, на котором функционирует ядро СУБД. Ядро обрабатывает полученные запросы и возвращает клиенту результат, оформленный как блок данных. При такой схеме программы на компьютерах-клиентах являются инициаторами манипуляций с данными, а ядру СУБД отводится пассивная роль. Основное достоинство RDA-модели состоит в унификации интерфейса взаимодействия с сервером с помощью стандартного языка запросов. Унификация позволяет реализовывать дополнительные меры по защите хранимой информации на уровне задания системы прав по отношению к тем или иным командам. В рассматриваемой модели также происходит существенная разгрузка трафика сети за счет того, что между станциями сети теперь передаются не части файла базы данных, а команды и ответы на них.

Основу *DBS-модели* составляет механизм хранимых процедур. *Хранимые процедуры* — это средство программирования сервера баз данных. Они хранятся в словаре базы данных, могут разделяться между различными клиентами и выполняются на том же компьютере, где запущен программный процесс сервера баз данных. Как правило, языки, на которых создаются хранимые процедуры, представляют собой процедурные расширения языка SQL. На настоящий момент не существу-

ет единого стандарта для таких языковых средств, поэтому они являются специфичными для каждой конкретной СУБД. Среди достоинств DBS-модели могут быть названы возможность централизованного администрирования прикладных функций, дальнейшее снижение трафика (вместо SQL-запросов по сети передаются вызовы хранимых процедур), экономия ресурсов компьютера за счет использования однократно разработанного плана выполнения процедуры. К традиционным проблемам, связанным с DBS-моделью, относят трудности, сопутствующие процессам создания, отладки и тестирования хранимых процедур. О популярности и перспективности данной модели свидетельствует прежде всего то, что она реализована в таких широко используемых реляционных СУБД, как Oracle, Sybase, Informix, Ingres.

В *AS-модели* процесс, выполняющийся на компьютере-клиенте, называется *клиентом приложения* (*Application Client – AC*) и отвечает за интерфейс с пользователем. В случае необходимости выполнить те или иные прикладные операции он обращается к серверу приложения (*Application Server – AS*). Все операции над информационными ресурсами выполняются специальными программными компонентами, по отношению к которым *AS* играет роль клиента. В качестве примера прикладных компонентов могут быть названы ресурсы различных типов – базы данных, очереди, почтовые службы и др.

Само по себе ядро базы данных Jet, как уже упоминалось ранее, не является процессором, поддерживающим технологию клиент-сервер. Однако с помощью Access можно создавать соответствующие приложения, связываясь с клиент-серверными источниками данных через ODBC.

7.4.3. Организация защиты данных в СУБД MS Access

Непременной функцией любой развитой СУБД является обеспечение защиты данных от несанкционированного доступа. Очевидно, что полноценный с точки зрения надежности и устойчивости режим защиты может быть обеспечен только в рамках промышленных систем управления при условии комплексной реализации мер программного, аппаратного и административного характера. Перед настоящим параграфом поставлена более скромная задача – на примере MS Access описать на принципиальном уровне те подходы, которые применяются в СУБД для обеспечения программной защиты данных.

MS Access обеспечивает два традиционных способа защиты базы данных:

- установка пароля, требуемого при открытии базы данных;
- защита на уровне определения прав пользователей, которая позволяет ограничить возможность получения или изменения той или иной информации в базе данных для конкретного пользователя.

Кроме того, можно удалить изменяемую программу Visual Basic из базы данных, чтобы предотвратить изменения структуры форм, отчетов и модулей, сохранив базу данных как файл MDE.

Установка пароля на открытие базы данных представляет собой простейший способ защиты. После того как пароль установлен (функция меню Сервис ▸ Защита ▸ Задать пароль базы данных), при каждом открытии базы данных будет появляться диалоговое окно, в котором требуется ввести пароль. Открыть базу данных и получить доступ к ее ресурсам могут получить только те пользователи, которые введут правильный пароль. Этот способ достаточно надежен (MS Access шифрует пароль, так что к нему нет прямого доступа при чтении файла базы данных). Однако проверка проводится только при открытии базы данных, после чего все ее объекты становятся полностью доступными. В результате, установка пароля обычно оказывается достаточной мерой защиты для баз данных, которые совместно используется небольшой группой пользователей или установленны на автономном компьютере.

Гораздо более надежным и гибким способом организации защиты является *защита на уровне пользователей*. Он подобен способам, используемым в большинстве сетевых систем. Процесс задания защиты на уровне пользователей состоит из двух принципиальных этапов:

- создание системы пользователей, объединенных в группы (Сервис ▸ Защита ▸ Пользователи и группы);
- задание прав доступа различных пользователей и групп по отношению к объектам базы данных (Сервис ▸ Защита ▸ Разрешения).

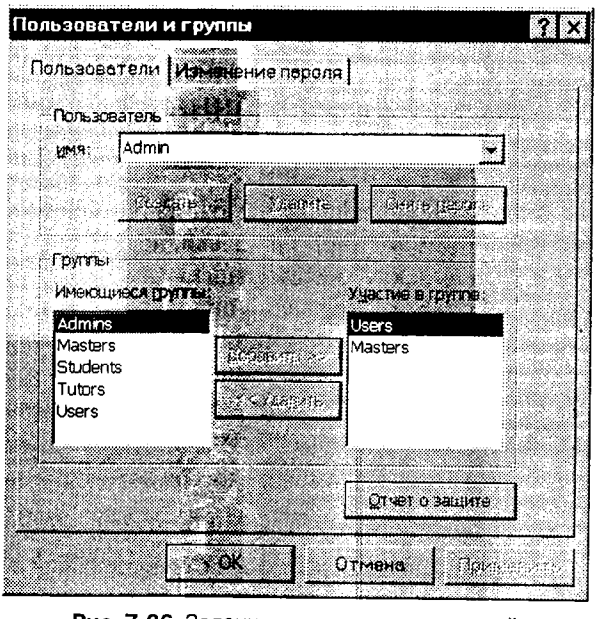


Рис. 7.36. Задание системы пользователей

Информация о системе пользователей сохраняется в специальном файле, называемом файлом рабочих групп. По умолчанию это файл System.mdw. Однако с помощью специальной программы, входящей в поставку Access, различные базы дан-

ных можно ассоциировать с различными файлами рабочих групп. При запуске Access от пользователей требуется идентифицировать себя и ввести пароль. Отдельные пользователи могут объединяться в группы, причем один и тот же пользователь может являться членом различных групп. Такая организация системы пользователей позволяет весьма гибко манипулировать набором их прав доступа, исходя из функциональной специфики предметной области. В файле рабочих групп Access по умолчанию создаются две группы: администраторы (группа Admins) и группа Users, в которую включаются все пользователи. Допускается также определение других групп. Процесс создания системы пользователей и определения их принадлежности к группам показан на рис. 7.36.

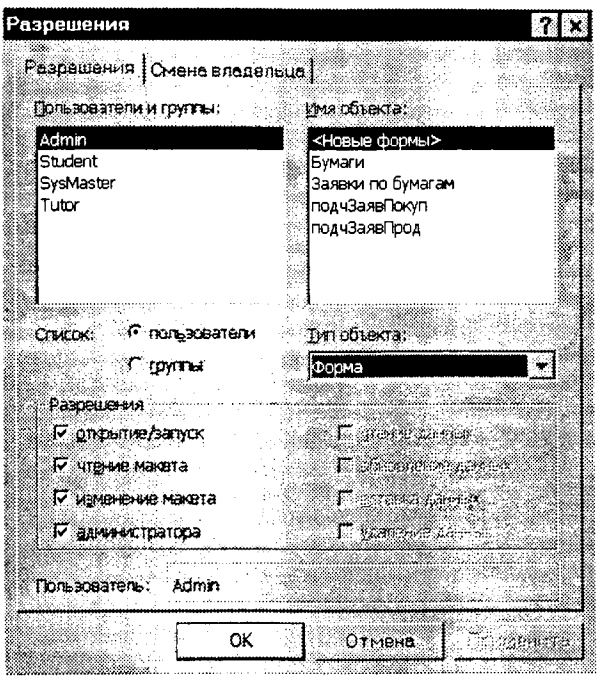


Рис. 7.37. Задание прав пользователей

Как группам, так и пользователям предоставляются разрешения на доступ, определяющие допустимые для них действия по отношению к каждому объекту базы данных. Набор возможных прав, очевидно, определяется спецификой объекта. Так, к примеру, список градаций разрешений на работу с экранной формой показан на рис. 7.37.

По умолчанию члены группы Admins имеют все разрешения на доступ ко всем объектам базы данных. Поскольку группа Users объединяет всех пользователей, то ей имеет смысл присваивать некоторый минимальный набор прав. Далее имеется возможность установить более разветвленную структуру управления, создавая собственные учетные записи групп, предоставляя этим группам соответствующие разрешения и добавляя в них пользователей. Процесс задания прав доступа пользователей по работе с формами базы данных TradeTest показан на рис. 7.37.

Заканчивая разговор о системах защиты, еще раз подчеркнем, что ее эффективная реализация возможна только на основе подробного изучения функциональной структуры автоматизируемого объекта и тщательного проектирования системы управления данными.

Ключевые понятия

- база данных;
- предметная область;
- ключ;
- запись, поле;
- модель данных;
- индекс (индексная таблица);
- модель данных;
- SQL;
- транзакция;
- DAO;
- ODBC;
- процессор баз данных Jet;
- клиент-сервер.

Контрольные вопросы

1. Что такое база данных и что такое СУБД? В чем различие этих понятий?
2. Дайте определения следующих понятий: объект, атрибут, запись, ключ.
3. Что такое модель данных? Какие модели вы знаете?
4. Основные свойства реляционной модели данных.
5. Что такое нормальные формы?
6. Назовите основные группы инструкций языка SQL.
7. Для чего служит инструкция SELECT?
8. Какие классы СУБД вы можете назвать? В чем их принципиальные различия?
9. Опишите основные этапы создания базы данных в среде MS Access.
10. Для чего служит схема данных MS Access?
11. Какие способы создания форм и отчетов в Access вы можете привести?
12. В чем основное различие функций макросов и модулей в Access?
13. Опишите основные принципы организации программирования доступа к данным в Access.

14. Какие основные методы доступа к внешним данным из СУБД Access вы можете назвать?
15. Для чего служит технология ODBC?
16. Опишите принципиальную схему организации доступа к данным в Access.
17. Какие принципиальные решения заложены в основу технологии клиент-сервер?
18. Перечислите основные этапы развития технологии клиент-сервер.
19. Какие основные методы защиты данных в Access вы можете назвать?

Литература

1. Бекаревич Ю. Б., Пушкина Н. В., Смирнова Е. Ю. Управление базами данных. СПб.: Изд. СПбГУ, 1999.
2. Горев А., Ахаян Р., Макашарипов С. Эффективная работа с СУБД. СПб.: Питер, 1997.
3. Корнелюк В. К., Веккер З. Е., Зиновьев Н. Б. Access 97. М.: СОЛОН, 1998.
4. Ramakrishnan R. Database management systems. McGraw-Hill, 1997.

Глава 8. Электронные презентации

В этой главе мы остановимся на программном обеспечении, которое решает задачи представления информации на качественно ином уровне, чем, например, в текстовых процессорах. Применительно к продукции компании Microsoft, лидирующей на программном рынке России, надо отметить следующее. Каждое приложение и утилита имеют свои специфические особенности — MS Word, в основном, ориентирован на автоматизацию традиционных функций подготовки оригинал-макетов комплексных документов, MS Graph позволяет создавать выразительные и информативные диаграммы, MS Clip Gallery содержит разнообразные забавные картинки и т. д. MS PowerPoint обеспечивает широкие возможности по координированию работы любых приложений, сведению данных воедино и представлению их для коллективного просмотра в виде так называемых презентаций.

8.1. Современные способы организации презентаций

На практике оказывается, что для грамотного экономиста мало только уметь работать с наборами данных, анализировать их и подготавливать сложные финансово-экономические документы. В конечном счете, нужные данные потребуется показать другим людям: коллегам, начальнику, деловым партнерам, — а делать это следует как можно нагляднее. Это и предполагает создание некой презентации.

Под *презентацией* (от лат. *praesento* — передаю, вручаю или анг. *present* — представлять) подразумевается передача и представление аудитории новых для нее идей, планов, разработок. Другими словами, презентация в нашем понимании — это демонстрационные материалы для практически любого более или менее публичного выступления, от доклада начальству до рекламной акции или лекции перед студенческой аудиторией. Компьютерная же презентация — это файл, в который собраны такие материалы. Подобно тому, как текстовый документ состоит из страниц, файл презентации состоит из последовательности кадров, или *слайдов*. Очевидны достоинства такой презентации по сравнению с привычными плакатами на ватмане:

- **последовательность изложения.** При помощи слайдов, сменяющих друг друга на экране, удержать внимание аудитории гораздо легче, чем двигаясь с указ-

кой между развешанных по всему залу плакатов. В отличие же от обычных слайдов, пропускаемых через диапроектор, экранные презентации позволяют быстро вернуться к любому из уже рассмотренных вопросов или вовсе изменить последовательность изложения;

- **возможность воспользоваться официальными шпаргалками.** Презентация — это не только то, что видит и слышит аудитория, но и заметки для выступающего: о чем не забыть, как расставить акценты. На соответствующих платформах и при наличии двух мониторов появилась возможность отображать слайды на одном мониторе, а различную служебную информацию (например, заметки) или же следующий слайд — на другом. Эти заметки видны только докладчику: они выводятся на экран управляющего компьютера;
- **мультимедийные эффекты.** Слайд презентации — не просто изображение. В нем, как и в любом компьютерном документе, могут быть элементы анимации, аудио- и видеофрагменты. Они способны не только существенно украсить презентацию, но и повысить ее информативность;
- **копируемость.** Копии электронной презентации создаются мгновенно и ничем не отличаются от оригинала. При желании каждый слушатель может получить на руки все показанные материалы;
- **транспортабельность.** Дискета с презентацией гораздо компактнее свертка плакатов и гораздо меньше пострадает от частых поездок по разным презентационным мероприятиям. Более того, файл презентации можно переслать по электронной почте, а при наличии доступа — опубликовать выступление в Интернете[1] и не тратить время на разъезды.

Для создания презентаций существует ряд программных продуктов, являющихся приложениями соответствующих комплексов — Office — разработок ведущих мировых производителей ПО для создания и представления комплексных документов.

Программы создания презентаций по принципам работы с ними находятся где-то посередине между текстовыми редакторами (Word, LyX и т. д.) и редакторами векторной графики (CorelDraw, xfig и т. д.). С помощью редактора презентаций можно создавать презентации различных типов: на экране, на слайдах (цветных и черно-белых) и на бумаге.

Презентация представляет собой серию независимых страниц, то есть, если текст и иллюстрации не помещаются на странице целиком, то избыток не переносится на новую страницу, а теряется. Распределением всех объектов по страницам и размещением их в пределах страниц управляет пользователь. При этом в его распоряжении обычно имеется обширный набор готовых объектов (геометрических фигур, пиктограмм, текстовых окон и т. д.). Самое важное в программе подготовке презентаций — это не число новых и необычных возможностей, а простота выполнения и степень автоматизированности тех операций, которые приходится выполнять чаще всего.

---

[1] Вопросы Интернета (Internet) и технологий глобальных компьютерных сетей подробно рассматриваются в главе 11.

Основными инструментами для подготовки и показа презентаций в мировой практике являются программы PowerPoint компании Microsoft, Corel Presentations фирмы Corel и пакет StarOffice компании StarDivision GMBH.

Создание презентаций в Corel's Office

В главе, посвященной обработке текстовой информации, описывался текстовый процессор WordPerfect из пакета Corel's Office, здесь же будет рассмотрено другое приложение этого Office — Corel Presentations.

В начале 2000 г. появился офисный пакет Corel's Office Suite for Linux. Стандартная комплектация включает в себя текстовый процессор WordPerfect 9, электронную таблицу Quattro Pro 9, программу создания презентаций Corel Presentations 9, CorelCentral 9, панель инструментов Desktop Application Director, Trellix 2 и средства интеграции с Microsoft Visual Basic for Applications. Встраиваемые модули позволяют публиковать файлы в форматах eXtensible Markup Language (XML), Adobe PDF или создавать гипертекстовые файлы в формате Trellix. Довольно скоро должны появиться пакет векторной графики CorelDRAW 9 и прочие усовершенствованные приложения Corel's Office. Главными задачами, которые решала Corel при разработке данной версии, были обеспечение совместимости и достижение максимальной эффективности. В данной главе нас, прежде всего, интересует программа создания презентаций Corel Presentations 9.

Эта программа дает возможность создавать высококачественные слайдовые презентации, прежде всего, благодаря усовершенствованным возможностям конвертации, импорта и экспорта файлов из других программных приложений для создания презентаций. Улучшенный альбом дает быстрый доступ к часто используемым изображениям и фотографиям, а графический редактор позволяет с легкостью модифицировать изображения. Усовершенствованный редактор для подготовки Интернет-публикаций дает пошаговые инструкции для публикации слайдовой презентации в Интернете.

В числе новых возможностей программы Presentations 9, как и других основных приложений Corel's Office, — функция предварительного просмотра реального времени, которая переформатирует документ на экране, пока пользователь экспериментирует со шрифтами, изменением параметров форматирования или с цветами, диаграммами и другими элементами графического оформления.

Что касается решения проблем совместимости, то надо отметить, что помимо того, что программа успешно оперирует с файлами приложений Corel's Office, Corel Presentations 9 также импортирует и экспортирует файлы PowerPoint и содержит новые инструменты управления яркостью, контрастностью и другими атрибутами растровых изображений.

Создание презентаций в StarOffice

В той же главе, посвященной обработке текстовой информации, мы уже рассматривали особенности офисного пакета, принадлежащего фирме StarDivision GmbH, — основная из них та, что, начиная с версии 4.0, все компоненты пакета интегриро-

ваны в единую большую программу. Поэтому довольно трудно выделить отдельно функции приложения, отвечающего за создание презентаций, хотя оно и существует. В «интегрированном пакете» StarOffice сам редактор презентаций называется Presentation.

Поскольку StarOffice — это, по существу, одна программа, все компоненты выполняются в одном окне. Фактически, программисты из StarDivision GmbH создали собственную полную оконную среду, которая работает внутри единственного окна, необходимого StarOffice. При таком подходе программу легче поддерживать; доказательством может служить то, что StarDivision уже создала версии StarOffice для нескольких платформ: OS/2, Windows, Linux, Sun и некоторых разновидностей Unix. По признанию разработчиков, от конкретной реализации зависит всего 20% кода.

Надо отметить, что разработчики StarOffice, как и Corel's Office, тоже учли, что интерфейс продуктов Microsoft оказывается безусловным лидером во всех областях и что в результате повсеместного распространения MS Office принятые в нем устройство меню и назначения клавиш сделались фактическим стандартом. Поэтому, чтобы обеспечить максимальную совместимость разных пакетов, они организовали и установку, и настройку, и работу с пакетом в полном соответствии с этим «стандартом».

Принципы работы по подготовке презентаций приблизительно такие же, как в других Office, например, той же компании Microsoft. Для создания презентаций в StarOffice предусмотрено два способа: либо за основу берется пустая презентация, либо используется ассистент — Auto Pilot. Во втором случае нужно последовательно пройти все окна ассистента. Результатом будет последовательность шаблонов слайдов, все происходит примерно так же, как в PowerPoint. Вкратце рассмотрим, какие средства для подготовки презентации содержит StarOffice Presentation, с тем, чтобы более подготовленными подойти к описанию новых и усовершенствованных возможностей PowerPoint.

Подавляющее большинство всех слайдов для презентаций очень похожи друг на друга: сначала идет заголовок, а за ним — чаще всего — текст, представляющий собой список пунктов, — это позволяет выбрать для слайда одну из предусмотренных стандартных раскладок (рис. 8.1). Структуру слайда можно в любой момент изменить.

Презентации, конечно, не должны состоять исключительно из текстовых слайдов. В хорошей презентации рекомендуется располагать текст так, чтобы размещение элементов наглядно представляло отношения между соответствующими понятиями. Возможность заключить текст в изображение — прямоугольник, овал или другую геометрическую фигуру — является непременной принадлежностью редактора презентаций. В Presentation имеются обычные прямоугольники, прямоугольники со скругленным краем, эллипсы и другие фигуры. Более сложные и реже используемые двухмерные фигуры, такие, какие есть в PowerPoint, здесь отсутствуют. Если они все же понадобятся, их придется создавать вручную, при помощи линий и многоугольников. Зато пакет содержит базовый набор трехмерных фигур, в которых можно менять множество параметров с весьма впечатляющим результатом. Любые атрибуты фигур можно менять, пользуясь как инструментальными панелями, так и контекстными меню, как и в приложениях MS Office.

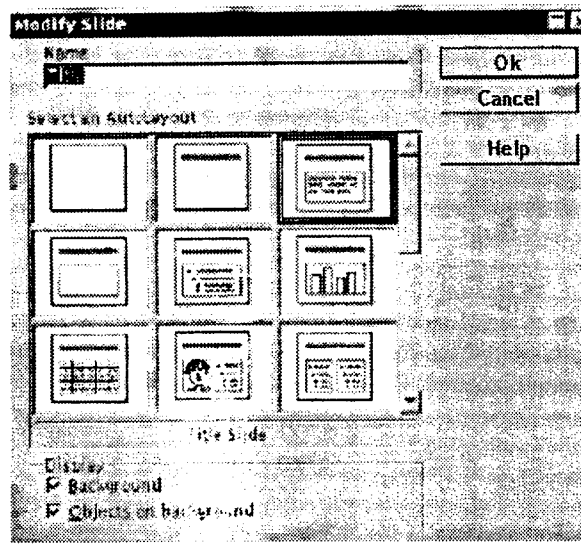


Рис. 8.1. Выбор структуры слайда в StarOffice

Особый интерес представляют линии определенного типа: соединительные линии — коннекторы. Если два объекта соединены коннектором, то при любых перемещениях они остаются соединенными. Линии могут оканчиваться квадратиком, кружочком, стрелкой, быть сплошными или пунктирными.

Также в слайд можно поместить рисунок — *растровый объект* форматов jpg, bmp, gif, png, tiff. С момента попадания в слайд изображение становится таким же внутренним объектом, как и любой другой: его можно перемещать, масштабировать и совершать с ним прочие действия. Здесь определим разницу между так называемыми *рисунками* и *изображениями*. Использование такой терминологии более корректно и позволяет избежать неточностей и искажения смысла при описании оперирования с графическими объектами, в большом количестве присутствующими в большинстве презентаций любого Office. Как было отмечено ранее, рисунок — это некий объект растровой природы, а изображение — это объект векторный. Они различаются по способу описания изображений. Векторные объекты задаются в виде набора векторов, определяющих перемещение пера графопостроителя по бумаге или электронного луча в трубке векторного дисплея. Объекты растровые задаются как комбинации точек в матрице заданного размера (в растре), что обусловливает ряд особенностей, по сравнению с векторными, главная из которых — неприспособленность к трансформациям, таким как масштабирование, поворот, наклон. Размеры растровых объектов могут иметь лишь ряд фиксированных значений, поскольку они определяются размерами матрицы (растра), на базе которой построены. Изменение размера или положения происходит путем замены одного растра на другой. Дело не в том, что объект совсем не может быть увеличен или уменьшен, но в том, что при выполнении этих операций существенно искажается его внешний вид. Поэтому рисунки в оформлении презентаций используются, как правило, как фиксировано вставляемые объекты, а в качестве фигур, подлежащих трансформации, берутся изображения.

Перейдем теперь к самым эффектным возможностям: анимационным эффектам. Если презентация проводится с помощью видеопроектора, то для такой демонстрации можно использовать саму программу Presentation. В этом случае докладчик меняет слайды, нажимая кнопку мыши, либо они сменяются автоматически по счетчику. Нужно лишь подсоединить видеовход проектора к выходу видеоплаты компьютера и нажать кнопку Slide Show. В этом случае можно сделать так, чтобы объекты двигались и взаимодействовали заранее заданным способом. В процессе создания презентации для каждого объекта может быть задано время появления и исчезновения. Кроме того, программа позволяет выбрать способ перехода от одного слайда к другому, предлагая большое количество стандартных типов перехода.

Наконец, самое привлекающее (а порой и отвлекающее) внимание средство — это средство озвучивания презентации. Здесь самое главное — помнить, что избыток звуковых эффектов, равно как и визуальных, может увести внимание аудитории от темы. Это же надо учитывать при создании и включении в презентации анимированных gif-файлов.

Наконец, последнее, что надо отметить, — так же, как родственные продукты компаний Corel и Microsoft, программа позволяет использовать в презентациях документы, созданные в других компонентах StarOffice, и точно так же встраивать в любые другие документы слайды презентаций.

8.2. Microsoft PowerPoint 2000 и его новые возможности

В PowerPoint 2000, по сравнению с предыдущей версией, появились новые возможности по созданию, средствам оформления и показу презентаций. Казалось бы, предыдущая версия этой программы создания презентаций полностью справлялась со своими задачами и никакой дальнейшей доработки не потребует. Но помимо того, что любую программу можно совершенствовать практически бесконечно, сильно меняются внешние условия, например развитие коммуникационных систем. Вследствие всего этого PowerPoint 2000 приобрела несколько новых черт.

Новые возможности, реализованные в PowerPoint 2000

- *Оптимизация графики Web-презентаций.* Обозреватель MS Internet Explorer 5.0 имеет встроенную функцию отображения и масштабирования векторной графики, что помогает повысить качество изображения и уменьшить размер файлов презентаций PowerPoint, создаваемых в формате HTML.
- *Двойной выходной формат HTML.* Пользователи могут сохранять презентации в формате для обозревателя Internet Explorer версий 3 и 4. Это позволяет оптимизировать файл для Internet Explorer 4.0 при сохранении совместимости с обозревателями версии 3. Презентации также могут публиковаться в формате, предназначенном только для Internet Explorer 3.

- *HTML как равноправный формат хранения файлов.* PowerPoint 2000 может с высокой степенью точности сохранять и читать файлы в формате HTML. Теперь HTML и формат двоичного файла (*.ppt) являются равноправными форматами.
- *Подгонка по ширине окна.* Автоматическая настройка слайдов в соответствии с разрешающей способностью дисплея и размером окна избавляет от необходимости вручную изменять их размер или разрешение экрана.
- *Сетевой показ презентации.* Возможность организовывать показ презентаций через интрасеть, отображая слайды в формате HTML вместе с комментариями в виде потокового видео и аудио (для использования потокового видео требуется сервер NetShow). Участникам сеанса необходим обозреватель Internet Explorer 4.0 (или более поздней версии) для просмотра презентаций.
- *Планирование сеансов вещания.* Программа обмена сообщениями и совместной работы Outlook позволяет планировать сеансы сетевого вещания, создавать напоминания и подключаться к сеансам вещания одним нажатием кнопки.
- *Web-страница событий.* С помощью этой страницы другие пользователи могут настраивать систему на просмотр презентации как во время вещания, так и после него, а также получать нужную информацию до начала вещания.
- *Показ по запросу.* Сеансы показа презентаций сохраняются на Web-сервере и доступны для просмотра в любое время.
- *Авторазмещение текста.* Размер текста автоматически изменяется так, чтобы он целиком размещался в рамке и не выходил из слайда.
- *Объединение трех областей просмотра.* Объединение режимов просмотра слайда, структуры и заметок. Такой режим просмотра упрощает выполнение многих действий, включая добавление новых слайдов, редактирование текста, внесение заметок во время создания презентации и переход между слайдами во время редактирования.
- *Таблицы.* PowerPoint позволяет создавать таблицы без использования MS Word или Excel. Таблицы составлены из фигур средств рисования Office, что обеспечивает простое редактирование. Таблицы привязаны к текстовым стилям и темам. Их можно свободно импортировать в PowerPoint 2000 и экспортировать из этого приложения.
- *Автоматическая нумерация маркеров.* PowerPoint позволяет использовать нумерованные маркеры, автоматически располагающиеся в правильной логической последовательности. При вводе числа маркеры форматируются в виде нумерованного списка без дополнительных команд пользователя.
- *Маркеры в виде картинок.* PowerPoint 2000 позволяет использовать графические изображения вместо маркеров. Приложение включает несколько образцов картинок, однако пользователи могут самостоятельно выбирать графику для маркеров.
- *Функции автозамены для нескольких языков.* Теперь PowerPoint 2000 поддерживает несколько списков автозамены, по одному для каждого языка.

Помимо предоставления совершенно новых возможностей, в PowerPoint 2000 были сохранены и частично усовершенствованы прежние, получившие широкое

признание и одобрение у пользователей. Для того чтобы кратко ознакомить с ними начинающего пользователя или освежить в памяти у более опытных пользователей основные функциональные возможности PowerPoint, рассмотрим некоторые из них.

Стандартные возможности PowerPoint

- *Сохранение возможностей приложения при работе с HTML.* Пользователи могут установить HTML как формат файла по умолчанию, сохраняя возможность использования ключевых функций PowerPoint.
- *Просмотр в обозревателе.* Работая с презентацией в формате PowerPoint HTML, вы можете воспользоваться специальной кнопкой для переключения в полноэкранный режим. Вместо программы просмотра для показа презентаций можно использовать обозреватель Internet Explorer 4.0 или более поздней версии.
- *Параметры вывода в формате HTML.* Несколько встроенных параметров форматирования позволяют выбирать формат, разрешение, стиль рамки, цвет, заметки докладчика, структуру, звуковые настройки, а также масштабирование слайдов и графики для презентаций в формате HTML.
- *Организация собраний в сети.* Интеграция с Microsoft NetMeeting позволяет организовывать конференции для совместного редактирования любых офисных документов. С помощью Outlook пользователи также могут планировать встречи по сети, устанавливать напоминания и подключаться к собраниям.
- *Автоформат при вводе.* Помимо нумерованных маркеров, PowerPoint также распознает порядковые числительные, дроби, длинные и короткие тире, элементы автозамены в формате RTF и парные кавычки, за которыми следует число.
- *Анимированные GIF-файлы.* Анимированные GIF-файлы можно вставлять в проект и отображать при показе слайдов и в обозревателе.

Усовершенствованные возможности PowerPoint 2000

- *Сохранение на Web-сервере.* Пользователи могут сохранять презентации PowerPoint в формате HTML непосредственно на Web-сервере. Эта возможность позволяет самостоятельно размещать информацию в интрасети, не обращаясь к Web-мастеру.
- *Рамка для заметок* Страницы заметок автоматически создаются в отдельной рамке для удобного просмотра.
- *Панель перемещения.* Презентации PowerPoint, сохраненные в формате HTML, включают панель перемещения в отдельной рамке для упрощения навигации.
- *Создание шаблонов.* Темы, доступные в некоторых приложениях пакета Office и средстве создания и управления Web-узлами FrontPage, позволяют создавать Web-страницы, оформленные в едином стиле. PowerPoint 2000 включает шаблоны оформления, согласованные с темами.
- *Удобный режим показа слайдов.* Возможность настройки автоматического скрытия указателя и значка показа слайдов после заданного периода неактивности

пользователя во время обычной полноэкранной демонстрации. Небольшие перемещения мыши не приведут к появлению указателя.

- *Речевое сопровождение.* Воспроизведение речевого сопровождения теперь синхронизировано с презентацией, в том числе со сменой слайдов и демонстрацией анимационных эффектов. Приложение также включает новую функцию записи сопровождения для отдельных слайдов.
- *Дополнительные автофигуры.* Более 50 новых автофигур, в том числе фигуры для сетевых и Web-диаграмм, заголовков концептуальных схем и планов офисов.
- *Пользовательский интерфейс средств анимации.* Интуитивный пользовательский интерфейс позволяет настраивать анимационные эффекты и группировать объекты для согласованной анимации.
- *Автоматическое определение языка.* PowerPoint 2000 определяет язык по раскладке клавиатуры и применяет соответствующие средства проверки.

8.3. Организация презентаций средствами MS PowerPoint

8.3.1. Экранный интерфейс и настройки PowerPoint

Запуск приложения PowerPoint 2000

Программное приложение PowerPoint 2000 входит в состав пакета Microsoft Office 2000 Professional (и не входит, допустим, в поставку Office для малого бизнеса). В случае корректной установки на компьютере оно может быть запущено любым из способов, предоставляемых операционной системой, например, из меню Пуск ▸ Программы ▸ Microsoft PowerPoint.

При запуске на экране появится окно PowerPoint с диалоговым окном, предлагающим разными способами создать презентацию (рис. 8.2). PowerPoint вообще кардинально отличается от других основных приложений MS Office, активно используемых экономистами, тем, что сам, помимо встроенного Помощника, является активным и разумным проводником по всем функциям создания, показа и публикации презентаций. В принципе, более опытные пользователи могут при очередном появлении данного диалогового окна воспользоваться предлагаемой им возможностью Больше не выводить это окно. Сложно сказать, имеет ли это смысл, скорее, как в случае любых индивидуальных настроек, это дело вкуса, поскольку в обоих случаях предоставляются аналогичные возможности работы.

Экранный интерфейс PowerPoint 2000

Несомненное удобство всех приложений MS Office заключается в единообразном представлении пользовательского интерфейса. Поскольку PowerPoint является Windows-приложением, его интерфейс стандартен для этой операционной системы. PowerPoint 2000 также, в отличие от предыдущих версий, имеет SDI-интерфейс — для каждой создаваемой презентации открывается отдельное окно с меню и пане-

лями инструментов (рис. 8.3). Конкретный вид окна существенно зависит от режима отображения документа, применяемого в данный момент.

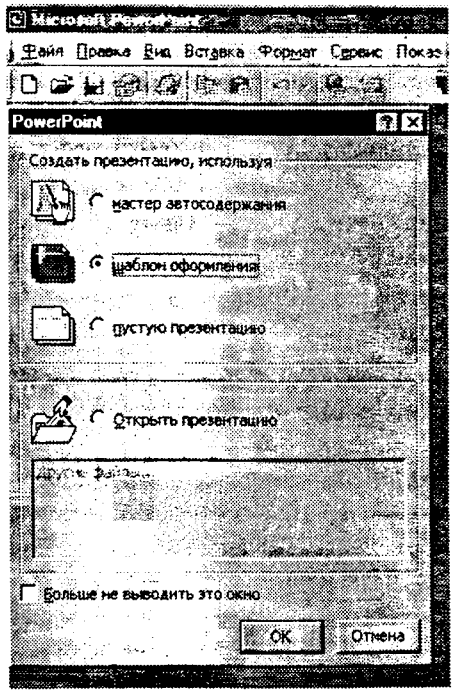


Рис. 8.2. Диалоговое окно, автоматически появляющееся при запуске PowerPoint

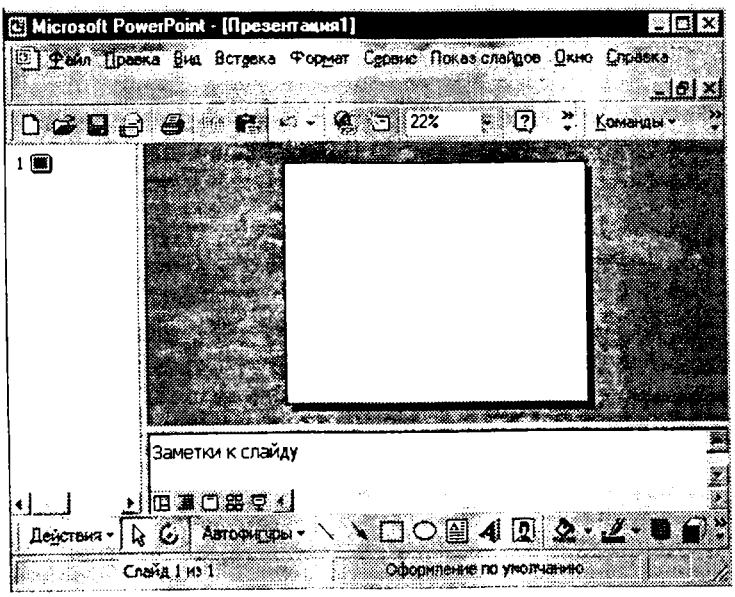


Рис. 8.3. Экранный интерфейс приложения PowerPoint

Управление приложением осуществляется с помощью клавиатуры и мыши. Многим командам соответствует заданное сочетание клавиш, так называемые «горячие» клавиши. В PowerPoint, в отличие от Word или Excel, «горячими» клавишами пользуются не так часто, хотя, теоретически, каждой команде можно поставить в соответствие подходящее сочетание клавиш. Элементы управления активизируются нажатием на них левой кнопки мыши. При нажатии правой кнопки мыши появляется контекстное меню с пунктами, соответствующими элементу, над которым нажата кнопка. Работа с контекстными меню, пожалуй, наиболее проста и удобна.

Когда курсор мыши задерживается над каким-либо управляющим элементом, то появляется окно подсказки с кратким описанием функций этого элемента.

При стандартных настройках окно MS PowerPoint состоит из следующих элементов:

- *заголовок окна* — стандартный для окна Windows. Слева вверху в строке заголовка находится значок PowerPoint, рядом — имя презентации, находящейся в работе, а затем — имя програмы: Microsoft PowerPoint. Если щелкнуть мышью на значке PowerPoint, то откроется системное меню, пункты которого позволяют установить подходящие размеры окна программы, свернуть или закрыть окно. То же самое можно сделать с помощью трех кнопок, расположенных на правом краю строки заголовка. Щелчок на левой кнопке обеспечивает сворачивание окна, щелчок на средней — увеличение или уменьшение его, а щелчок на правой кнопке — закрытие;
- *строка меню* расположена под строкой заголовка. Меню со списком открываются щелчком на них левой кнопки мыши. PowerPoint 2000 также имеет «интеллектуальное» меню, которое автоматически настраивается в зависимости от последних используемых команд и метода работы пользователя. С накоплением определенного опыта работы пользователь начинает приспосабливаться к интерфейсу PowerPoint и представлять, какие пункты меню ему требуются чаще всего. Тогда PowerPoint предоставляет прекрасную возможность создать пользовательскую панель инструментов. Это выполняется в меню Сервис ▸ Настройка, где в открывшемся диалоговом окне на панели Команды надо выбрать пункт Новое меню;
- *панели инструментов.* Ниже строки меню располагается область панелей инструментов. По умолчанию на экране отображаются панели Стандартная и Форматирование. Панель инструментов состоит из управляющих элементов — кнопок, областей ввода и выбора, что позволяет быстро выполнять команды и настройки, не обращаясь к меню. С помощью настроек можно создать или активизировать дополнительные панели инструментов, а также настроить находящиеся на них элементы управления;
- *рабочая область* — поле, в котором отображается и редактируется документ. Справа рабочая область ограничена полосой прокрутки. Существенная разница, по сравнению с Word или Excel, в том, что бегунок позволяет переходить от слайда к слайду, а не по тексту в пределах одного слайда. Конкретный вид рабочей области зависит от режима работы редактора. Улучшение интерфейса

PowerPoint в большой степени затронуло рабочую область окна, которая из двухпанельной стала трехпанельной;

- *строка состояния*. Внизу окно программы замыкает строка состояния. Это информационная строка, в которой указываются режимы просмотра и различная информация о презентации: число слайдов, текущее положение в презентации, язык, параметры оформления слайда и т. п.

Для того чтобы начинающий пользователь мог иметь общее представление о возможностях, предоставляемых PowerPoint при подготовке и показе презентации, опишем основные пункты *встроенных меню*:

- Файл. Это меню содержит команды создания, открытия, сохранения, переноса и публикации презентации;
- Правка. Здесь содержатся команды работы с буфером обмена, выделением части документа для последующего редактирования, поиска и замены слайдовых фрагментов, а также отмены изменений и возврата к ним;
- Вид. Этим меню переключаются режимы просмотра презентации, отображение панелей инструментов и линеек, возможности просмотра различных образцов, работа с колонтитулами, масштаб и, наконец, создание заметок, существенно облегчающих докладчику проведение презентации с экрана;
- Вставка. С помощью этого меню производится вставка в слайды различных объектов: рисунков, диаграмм, мультимедийных эффектов, стандартов оформления слайдов (даты, номера слайда), — а также осуществляется работа с элементами других приложений Office;
- Формат. Команды этого меню позволяют менять параметры форматирования слайдов: шрифтов, фона, цветового оформления и расположения объектов на слайде, — в том числе методом применения шаблонов оформления и *авторазметки*;
- Сервис. Содержит различные сервисные операции: проверку орфографии, настройку языка, параметров программы, автоматизацию ввода текста, совместную работу над одним документом, работу с макросами, в том числе — и установку безопасности от макросов;
- Показ слайдов. Это меню оперирует с функциями, отвечающими за проверку предварительного просмотра презентаций, а также обеспечивает показ презентаций в электронном виде. Оно дает возможность настроить параметры смены каждого слайда, анимационные эффекты и интересную функцию работы со скрытым слайдом;
- Окно. Это меню дает возможность управлять параметрами окна со слайдами;
- Справка. Через это меню становится доступной система помощи редактора презентаций, где самая интересная и полезная услуга — вызов Помощника.

После описания возможностей каждого меню рассмотрим имеющиеся в распоряжении пользователя встроенные панели инструментов. В программе PowerPoint 2000 в списке *панелей инструментов* их насчитывается тринадцать. Внутри панелей

инструментов собраны элементы управления, представляющие операции, выполнение которых необходимо при работе с презентациями. То есть панели инструментов обеспечивают простой структурированный доступ к важнейшим функциям PowerPoint 2000. Управление отображением панелей инструментов осуществляется в меню Вид ▸ Панели инструментов. По этой команде появляется полный список панелей инструментов. Если перед названием панели установлен флажок, то она отображается на экране. Выключение приводит к ее скрытию. Пользователь по своему усмотрению может вызывать на экран и убирать любую панель, а также создавать свои собственные пользовательские панели путем комбинирования различных, наиболее часто употребляемых элементов управления. Создание пользовательских панелей инструментов производится в меню Сервис ▸ Настройка (рис. 8.4). Так же, как и при создании нового меню, надо в появившемся диалоговом окне сделать активной панель Команды и затем компоновать свою панель инструментов необходимыми элементами управления.

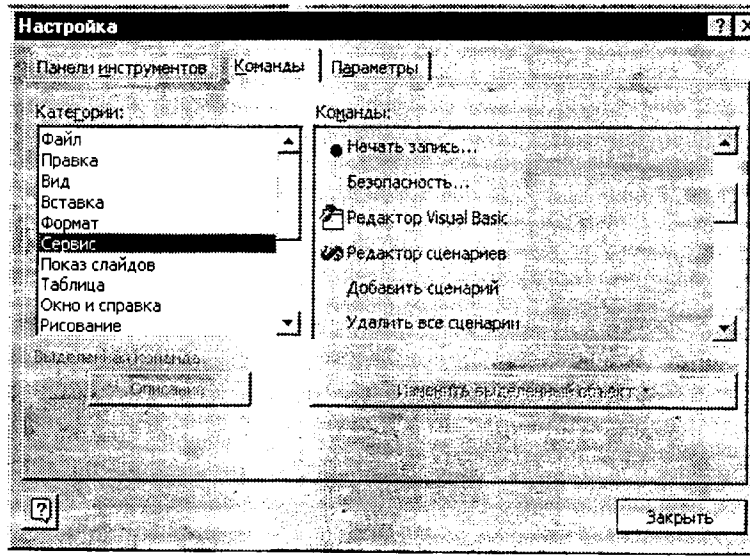


Рис. 8.4. Диалоговое окно настройки меню и панелей инструментов

Перечислим панели инструментов PowerPoint:

- Стандартная — содержит команды файловых операций, печати, редактирования и экранного отображения;
- Форматирование — команды по изменению стиля и формата объектов слайда;
- Visual Basic — команды доступа к созданию и отладке макросов на языке VBA (Visual Basic for Applications);
- Web — содержит команды для создания презентации, предназначенной для публикации в виде Web-страниц;
- Word-Art — доступ к стандартному набору художественных элементов оформления документа;

- Буфер обмена — набор команд, позволяющих копировать объекты с помощью буфера обмена;
- Настройка изображения — команды, управляющие параметрами растровых изображений;
- Рецензирование — доступ к функциям работы с различными версиями документа, внесение и удаление изменений без изменения исходного текста;
- Рисование — команды, позволяющие создать и изменить параметры векторных изображений, выполняющие простые графические операции;
- Структура — содержит набор команд, позволяющих оперировать со структурой презентации: перемещать и сортировать слайды, повышать и понижать уровень абзацев;
- Таблицы и границы — команды по установке параметров таблиц и границ страниц слайдов;
- Элементы управления — набор готовых компонентов ActiveX для создания Web-страниц и форм;
- Эффекты анимации — команды, позволяющие добавить в презентацию анимационные эффекты и изменять их параметры.

Еще один аспект, который необходимо осветить в этом разделе, — это режимы просмотра редактора PowerPoint. Приступая к созданию презентации, необходимо знать, чем отличается один способ ее просмотра от другого и какими из них лучше пользоваться при выполнении тех или иных операций. PowerPoint предлагает пять, или, если быть еще точнее, шесть режимов отображения документов. Переключение в тот или иной режим осуществляется в меню Вид, либо пятью кнопками в левой части горизонтальной полосы прокрутки.

Обычный режим. Когда мы говорили об улучшении пользовательского интерфейса применительно к рабочей области, то прежде всего имели в виду именно этот режим. Практически он является основным рабочим режимом в процессе создания презентации. Удобство его в том, что он трехпанельный: в одной из панелей (слева) редактируется текст слайдов и отображается их список, вторая панель служит для показа макета слайда (справа вверху), а третья панель позволяет вводить комментарии к нему или заметки (справа внизу). Причем размер и расположение панелей таковы, что позволяют организовать работу по созданию презентации максимально эффективно, без перемещения границ панелей, хотя такая возможность тоже заложена в PowerPoint.

Режим структуры. Этот режим является в некотором роде трансформацией обычного режима, с утрированным выделением структуры презентации. Он помогает докладчику переорганизовать порядок слайдов в презентации и выделить основные моменты, которые должны быть в ней представлены. Более подробно о полезности работы со структурами будет рассказано в следующем разделе.

Режим слайдов. Режим слайдов концентрирует внимание докладчика на внешнем виде самого слайда. Им имеет смысл воспользоваться для улучшения дизайна каждого конкретного слайда и обеспечения наибольшей его информативности за

счет корректировки размещения текста, графики, проверки правописания и наглядности.

Режим сортировщика слайдов. Этот режим, наоборот, позволяет оценить вид презентации в целом. В нем на экране в миниатюре отображаются все слайды, следующие один за другим в нужном порядке. Пользователь может увидеть, насколько единообразно построена вся презентация, а также, при желании, переставить один или несколько слайдов в другое место презентации.

Режим Показ слайдов. Этим режимом пользуются для предварительного просмотра, репетиции готовой презентации, а также для реального показа презентации.

Режим Страницы заметок. В режиме Страницы заметок утрированно выделена панель окна, отвечающая за внесение заметок докладчика. Этот режим отличается от прочих тем, что для него нет соответствующей кнопки в полосе прокрутки, а войти в него можно лишь из меню Вид ▸ Страницы заметок.

8.3.2. Создание новой презентации и оперирование структурой

Под созданием новой презентации мы будем понимать, аналогично «созданию нового документа» в Word 2000, инициирование создания презентации через меню Файл ▸ Создать или при помощи диалогового окна, автоматически появляющегося при запуске PowerPoint. В любом случае, создать новую презентацию можно тремя разными способами:

- при помощи мастера автосодержания;
- на основе предлагаемых шаблонов;
- используя пустую презентацию, или, иначе говоря, создавая документ «с нуля».

Рассмотрим каждую из предлагаемых возможностей.

Создание презентации при помощи мастера автосодержания

Создание презентации при помощи мастера автосодержания — это самый легкий, но зато самый стандартизированный способ. В этом случае мастер, как и большинство мастеров Office 2000, задает вам определенные вопросы, собирает тем самым необходимую информацию и предлагает на ее основе набор слайдов по указанной теме. Для начинающего пользователя этот способ видится самым простым и удобным, поскольку мастер грамотно ведет его через все стадии подготовки презентации. С другой стороны, он самостоятельно принимает ключевые решения, практически не оставляя возможности для свободного творчества. Щелчком мыши на значке Мастер автосодержания вызывается следующее диалоговое окно (рис. 8.5).

Слева в окне находится список, указывающий, из каких этапов мастер планирует построить работу по подготовке презентации.

На первом этапе он предлагает выбрать Вид презентации. При этом предлагается шесть блоков, раскрываемых нажатием соответствующих кнопок. Каждый из

блоков содержит несколько стандартов презентаций, из которых вы выбираете наиболее тематически подходящую. Единственное неудобство работы с мастером заключается в том, что невозможно предварительно посмотреть на экране, что же мы выбрали.

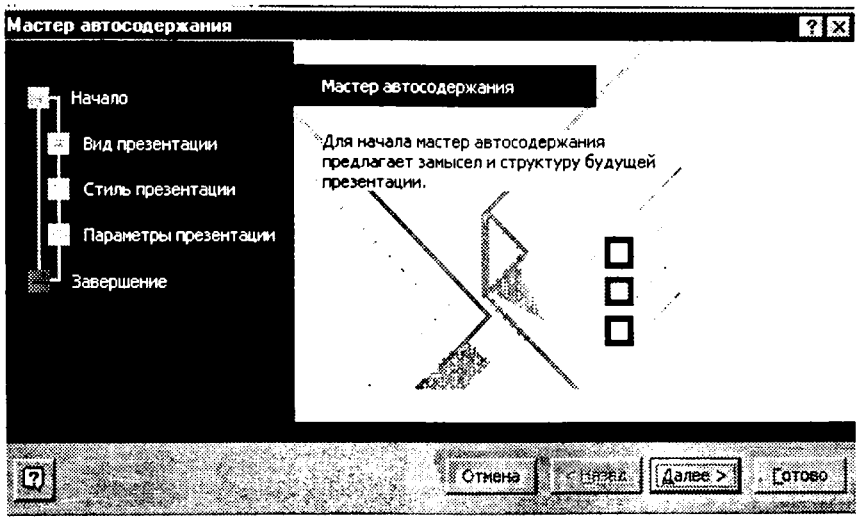


Рис. 8.5. Создание презентации при помощи мастера автосодержания

На втором этапе — Стиль презентации — вы указываете вид последующей публикации презентации: более традиционная печать на бумаге, слайдах или прозрачных пленках либо же публикация в Интернете, которая становится все более популярной.

Наконец, на третьем, последнем этапе производится оформление титульного слайда. Здесь дается название презентации, а также вводится текст для нижнего колонтитула всех слайдов. Щелчок мышью на кнопке Готово дает мастеру команду о завершении работы, и он на основе собранной информации создает базовый набор слайдов. На экран выводится первый слайд в Обычном режиме, и теперь можно заниматься окончательным оформлением слайдов, вводя необходимые данные: текст, изображения, рисунки, — а также, при необходимости, меняя параметры форматирования.

Создание презентации на основе шаблонов

Создание презентации на основе шаблонов имеет несколько другой оттенок, нежели использование шаблонов, например, в Word. PowerPoint 2000 предлагает два типа шаблонов:

- шаблоны презентаций;
- шаблоны оформления.

Шаблоны презентаций (в диалоговом окне PowerPoint они называются Презентации), более всего похожи на стандартные шаблоны в Word: они носят названия План продаж, Бизнес-план, Обзор финансового состояния и др. Эти шаблоны так же

содержат наиболее подходящие и принятые для каждой тематики стандарты и служат основой для создания чаще всего применяемых стандартных типов презентаций. Пользователь может принять к сведению предлагаемый шаблоном текст или полностью заменить его на авторский. Ниже, на рис. 8.6. приведен пример нескольких слайдов из набора, предлагаемого шаблоном презентации Вооодушевляем сотрудников.

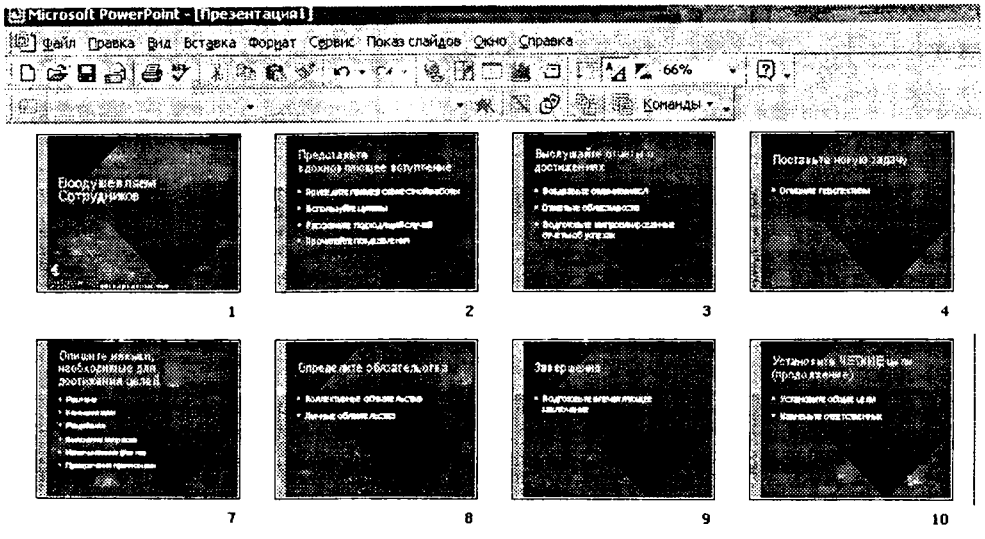


Рис. 8.6. Пример слайдов, предлагаемых шаблоном презентации

Шаблоны оформления, пожалуй, не имеют аналогов в других приложениях MS Office. Они созданы профессиональными дизайнерами и служат для придания всем слайдам презентации единообразного оформления. Каждый из шаблонов имеет определенную цветовую гамму, фон и свою стилистику, содержит разнообразные графические элементы, параметры размера и начертания шрифтов, некоторые специальные эффекты. Некоторые из них выдержаны в совершенно темных тонах, такие, например, как Ржавый замок, Мрамор или Синий обелиск. Иные — наоборот, практически белые, с минимальным набором оформительских изысков, например Галстук или Рисовая бумага. При работе с презентацией необходимо внимательно отнестись к выбору шаблона оформления, поскольку помимо того, что он может не сочетаться с темой презентации, на нем элементарно могут потеряться вводимый текст или вставляемые графические объекты.

При выборе способа создания презентации надо учитывать, что шаблоны презентаций и шаблоны оформления — это не взаимозаменяемые вещи, а скорее взаимодополняющие. На практике оказывается наиболее удобным сначала создать слайды презентации каким-либо из возможных способов, ввести текст и графику, при желании — воспользоваться анимацией, составить все надлежащим образом, и лишь затем приступать к окончательному оформлению в единой цветовой гамме при помощи, например, тех же шаблонов оформления либо заданием соб-

ственного сочетания цветов и фона. Единственное, что надо при этом помнить, — это то, что после задания авторских параметров форматирования PowerPoint уже не меняет их на шаблонные, даже если они вступают в противоречие со стилями выбранного шаблона оформления.

Для того чтобы задать новый шаблон оформления для готовых слайдов, надо выбрать в меню Формат ▸ Применить шаблон оформления, а затем в открывшемся диалоговом окне щелкнуть мышью на подходящем шаблоне и нажать кнопку Применить. Внешний вид слайдов с применением назначенного шаблона можно предварительно посмотреть в правой части диалогового окна.

Иной способ цветового оформления слайдов — это задание цветовой гаммы и фона иными средствами меню Формат.

При помощи команды меню Формат ▸ Цветовая схема слайда производится изменение цвета и интенсивности фона. В открывающемся диалоговом окне можно выбрать одну из предложенных цветовых схем на панели Стандартная либо настроить свою цветовую схему на панели Специальная и, при желании, сохранить ее, щелкнув мышью на кнопке Добавить как стандартную схему. Затем для того, чтобы завершить цветовое оформление, надо щелкнуть на кнопке Применить, если вы хотите оформить таким образом только текущий слайд, либо Применить ко всем, тогда в таких цветах будут раскрашены все слайды презентации. Лучше, как правило, единообразно оформлять всю презентацию, особенно в тех случаях, когда освещаются аспекты бизнес-планов, инвестиционных проектов, ноу-хау, глобальных экономических разработок.

Командой меню Формат ▸ Фон вызывается диалоговое окно, позволяющее разработать авторский дизайн фона для слайдов: как для одного текущего слайда, так и для всей презентации в целом. На создание фона имеет смысл обратить особое внимание, поскольку предоставляемые PowerPoint возможности здесь очень широки, а с помощью удачного задания фона можно сделать презентацию гораздо более выигрышной. Для того чтобы создать интересный фон, надо в этом диалоговом окне открыть список Заливка фона и выбрать команду Способы заливки. Тогда появляется новое диалоговое окно, содержащее четыре панели: Градиентная, Текстура, Узор, Рисунок. Наиболее эффектно и в то же время не раздражающе смотрятся градиентные заливки. К тому же их использование может иметь вполне функциональное предназначение. Если, например, использовать вертикальную градиентную заливку и верхнюю часть слайда сделать более светлой, а затем при движении вниз переходить ко все более и более темному тону, то можно добиться визуального эффекта устойчивости слайда. Если двигаться от заголовка, то можно создать эффект выпуклости или вогнутости слайда с визуальным выделением заголовка. Особый интерес представляет панель Рисунок. С ее помощью можно в качестве фона назначить, например, фотографию презентуемого объекта, хранящуюся в виде файла *.tif, или логотип компании, ведущей инвестиционный проект.

Команда меню Формат ▸ Разметка слайда открывает диалоговое окно, изображенное на рис. 8.7, в котором содержатся стандартные наборы компоновки слайдов. В правом нижнем углу окна выводится описание выбранного варианта слайда.

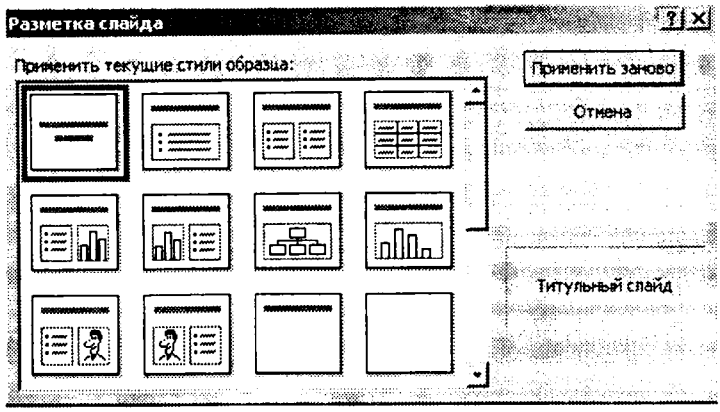


Рис. 8.7. Диалоговое окно Разметка слайда

Создание презентации без использования мастера и шаблонов

Создание презентации без использования мастера и шаблонов также несколько отличается от создания, например, нового документа в Word, или книги в Excel.

Существенная разница в том, что там документы создаются действительно «с нуля». PowerPoint тоже может предоставить такую возможность, но по умолчании при выполнении команды Новая презентация появляется диалоговое окно, аналогичное тому, которое открывалось из меню Формат ▸ Разметка слайда. Разница лишь в том, что здесь оно называется Создание слайда и предлагает не Применить текущие стили образца, а Выбрать автомакет. Точно так же в правом нижнем углу можно ознакомиться с описанием предназначения данного слайда и в том случае, если вариант подходит, нажать кнопку OK. В любой презентации, естественно, первым слайдом должен идти титульный. Если ни один из вариантов пользователя не устраивает или он хочет проявить максимум самостоятельности, то можно закрыть это окно и создавать презентацию «с нуля». Чаще всего делать этого не стоит, поскольку в общем случае чем более автоматизированы операции, тем проще и быстрее создается документ. Поэтому, при всем богатом творческом потенциале докладчика, не стоит пренебрегать встроенными функциями автоматизации, а также возможностями настройки, предлагаемыми PowerPoint. В частности, о преимуществах логического форматирования перед физическим говорилось в главе 5. Напомним вкратце, в чем их суть. *Физическое форматирование* — это такой способ, при котором автор выбирает для каждого элемента презентации параметры физического форматирования (тип, размер и цвет шрифта, отступы абзацев и т. п.). При таком способе оформления презентации автору приходится постоянно следить за единым стилем отображения одинаковых элементов презентации, что усложняет работу и может привести к ошибкам. Другой недостаток физического форматирования — сложность обеспечения единого стиля в случае, когда готовится много презентаций разными людьми и необходимо выдерживать их в единообразном оформлении. И, наконец, при этом программа-редактор презентации не получает

информации о логической структуре документа, что не позволяет реализовать многие возможности автоматизации обработки документа (например, конвертирование в формат HTML с целью последующей публикации в Интернете или применение единого шаблона для массива слайдов). При *логическом форматировании* автор размечает в презентации ее логические элементы и указывает, какое физическое форматирование будет иметь каждый тип элемента.

8.3.3. Оформление презентации

PowerPoint 2000 содержит обширные возможности по красочному оформлению слайдов, входящих в презентацию. Но основная информативная роль принадлежит содержащемуся в них тексту. Поэтому прежде всего остановимся на особенностях ввода и редактирования текста. Прежде чем приступать непосредственно к вводу текста, лучше сразу установить некоторые полезные параметры настройки.

Параметры работы с презентацией можно задать, выбрав в меню Сервис пункт Параметры. При этом появляется диалоговое окно, содержащее шесть вкладок. Окно изображено на рис. 8.8. Каждая из панелей предоставляет ряд возможностей по установке параметров отображения слайдов, печати, сохранению, проверке орфографии и т. п. На что стоит обратить особое внимание при работе с PowerPoint — это на панель Сохранение, поскольку в ней находятся важные для обеспечения сохранности документа и удобства работы с ним функции, а также на панель Орфография и стиль, потому что при проведении презентации любые грамматические и орфографические ошибки, а также неудачный подбор параметров отрицательно влияют на аудиторию.

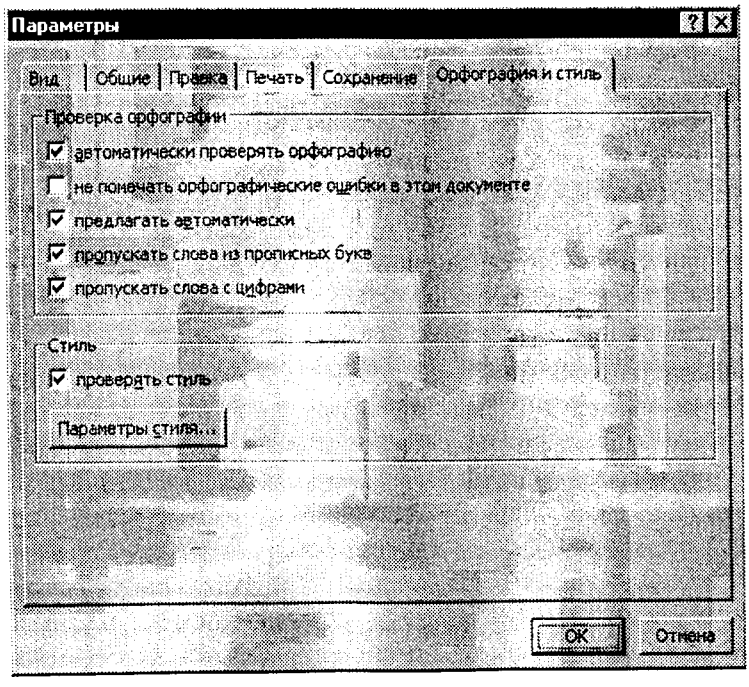


Рис. 8.8. Диалоговое окно настройки параметров работы с презентацией

На панели Сохранение наибольший практический интерес представляют:

- Разрешить быстрое сохранение — его лучше отключить, так как оно может привести к определенным неприятностям, таким как, например, уменьшение скорости обработки документа или неадекватное восприятие действий с ним;
- Автосохранение каждые — его, наоборот, лучше включить. Про плюсы и минусы этой команды можно прочитать в 5.3.2.

На панели Орфография и стиль важнейшей командой является функция Автоматически проверять орфографию. По умолчанию она должна быть задана, но лучше зайти в это диалоговое окно и проверить, стоит ли флажок напротив этой команды. Если да, то в дальнейшем во время набора текста PowerPoint будет подчеркивать орфографические ошибки красной волнистой линией, что позволяет докладчику сразу обратить на них внимание и исправить во время подготовки презентации к показу.

Ввод текста в PowerPoint осуществляется с клавиатуры и особых сложностей не вызывает, хотя и существенно отличается от ввода текста, например, в Word. Отличие обусловлено компоновкой рабочей области, а также специфическими особенностями создания презентаций. Исходя из этого, можно отметить, что способы ввода текста в слайды таковы.

При создании презентации на базе шаблона, включая шаблон Мастера автосодержания, а также при работе с Пустой презентацией, используя *встроенную авторазметку,* ввод текста производится в специальные местозаполнители. Разница лишь в том, что шаблоны предоставляют пользователю многоцветный дизайн и ряд предположений относительно содержания каждого слайда, а авторазметка — лишь простой фон и резервирование места на слайде для информации, которая будет туда занесена.

PowerPoint имеет удобное функциональное свойство: он помнит все производимые с документом действия и предоставляет возможность отказа от них, а также последующего возврата к ним. Отмена действий осуществляется при помощи меню Правка ▸ Отменить «указание совершенного действия» либо синей полукруглой стрелкой влево на панели инструментов Стандартная. Возврат к ним производится из того же меню Правка ▸ Повторить «указание совершенного действия» либо синей полукруглой стрелкой вправо на панели инструментов Стандартная.

Ввод текста в местозаполнители чем-то похож на аналогичную работу с шаблонами в Word. Окруженные пунктирным или штриховым контуром местозаполнители форматируются определенным шрифтом с заданным размером символов, а содержащиеся в них рекомендации по содержимому должны заменяться текстом докладчика. Для ввода в них авторского текста надо: выделить текст местозаполнителя; заменить его на авторский; затем щелкнуть мышью в любом месте за пределами местозаполнителя. Если ранее пользователь PowerPoint сталкивался с проблемой вылезания текста за пределы рамки, то в новой версии текст автоматически подгоняется под имеющееся свободное место. Единственное, за чем надо внимательно следить — это за тем, чтобы местозаполнитель вместе с текстом не выходил за

пределы слайда. В PowerPoint 2000 при необходимости можно передвигать границы местозаполнителей на слайде или менять размер шрифта в тексте. Основные функции ввода и редактирования текста, как, например, перемещение по тексту, использование пробелов, переход к следующему абзацу и другие, аналогичны заложенным в Word, поэтому ознакомиться с ними можно в главе 5. Вместе с тем, использование местозаполнителей предполагает ряд особенностей.

- Прежде всего, надо помнить, что при работе с шаблоном перед вводом *текст местозаполнителя выделяется обязательно,* в противном случае авторский текст будет вставлен перед текстом с рекомендациями по содержимому.
- Иногда требуется *включить в слайд пустой местозаполнитель.* Для этого надо выполнить команду меню Вставка ▸ Надпись и задать размеры нового местозаполнителя.

Наконец, немаловажной функцией, предоставляемой PowerPoint, является *проверка орфографии.* Обнаружение орфографических ошибок при проведении презентации не только ставит в неудобное положение докладчика и отрицательно сказывается на его выступлении, но и может вызвать раздражение зрителей до такой степени, что впечатление от презентации из позитивного превратится в негативное. Поэтому здесь на проверку орфографии стоит обратить особое внимание. Прежде всего, необходимо проверить, включена ли автоматическая проверка орфографии. Об этом уже говорилось ранее, когда речь шла о тематике настроек перед вводом текста. Кроме того, не лишним будет проверить правописание во всем документе при помощи элемента управления Орфография на панели инструментов Стандартная. При этом PowerPoint двигается по тексту в поисках слов с возможными орфографическими ошибками, не содержащихся в его стандартном и пользовательских словарях. При обнаружении такого слова он открывает диалоговое окно Орфография, где выписывает найденное слово и варианты замены при наличии таковых либо сообщение, что вариантов замены не знает. При этом предлагается несколько вариантов действий:

- Пропустить, чтобы оставить слово в таком написании;
- Пропустить все, чтобы впредь не останавливаться на этом слове во всей презентации;
- Добавить, чтобы внести это слово в пользовательский словарь;
- Заменить, чтобы заменить слово на выделенное предложенное;
- Заменить все, чтобы заменить данным образом все такие слова в данной презентации;
- Предложения, чтобы выбрать и произвести замену из предложенного списка.

Если в диалоговом окне отсутствует правильный вариант написания слова, то надо ввести его в области Заменить на и затем щелкнуть мышью на кнопке Заменить или Заменить все. Для большего понимания многообразия предоставляемых возможностей на рис. 8.9 приведен вид данного диалогового окна.

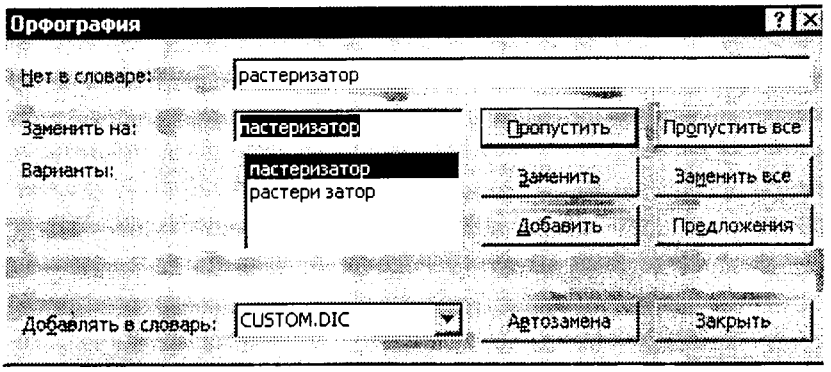


Рис. 8.9. Диалоговое окно проверки орфографии в PowerPoint

При работе с пустой презентацией можно выбрать два разных способа. Можно вводить текст с чистого листа, что, как правило, не очень удобно. Вторым, более грамотным, методом является организация замысла в виде структуры, в том числе и использование структуры готового документа в Word, и затем превращение ее в слайды. Структура состоит из последовательности заголовков с подчиненными им подзаголовками более низких уровней. Они помогают не только упорядочить авторский замысел при создании презентации, но и выделить относительную важность различных положений презентации. Работу со структурой лучше всего производить в Режиме структуры, хотя такая возможность предоставляется и в Обычном режиме.

При создании структуры в самом документе PowerPoint необходимо учитывать несколько моментов:

- В зависимости от предыдущего введенного уровня структуры сочетание клавиш Ctrl+Enter переключает программу между созданием нового слайда и маркированного элемента. То есть если последний введенный текст был заголовком слайда, то сочетание этих клавиш переключает на создание маркированного элемента, и наоборот, если последний введенный текст был маркированным элементом, то нажатие клавиш переключает на создание нового слайда.
- После добавления абзаца нажатие клавиши Enter создает новый абзац, уровень которого совпадает с уровнем предыдущего. При этом если последний введенный текст был маркированным элементом, то нажатие Enter создает еще один элемент, а если последний введенный текст был заголовком слайда, то нажатие клавиши создает новый слайд.
- PowerPoint предусматривает возможность создания документов с многоуровневой структурой. При необходимости создания структуры с несколькими уровнями иерархии используются кнопки Повысить уровень — для того чтобы поднять абзац в иерархии на один уровень, и Понизить уровень, чтобы понизить абзац в иерархии на один уровень или переместить заголовок слайда на предыдущий слайд.

- Также Режим структуры, равно как и Режим сортировщика слайдов позволяет перемещать слайды в презентации. Для того чтобы перенести слайд в другое место презентации, необходимо выделить его щелчком мыши на значке слайда в структуре. Затем, не отпуская кнопку мыши, надо переместить его в выбранное место. При этом самой распространенной ошибкой является вставка слайда внутрь другого слайда. Для того чтобы этого не произошло, необходимо тщательно следить за горизонтальной линией, показывающей конечное положение перемещаемого слайда в структуре. Как только линия окажется в выбранной позиции, кнопку мыши можно отпускать. Слайд окажется перенесенным и перенумерованным с соответствующим изменением всей структуры презентации. Те же особенности имеются при работе в Режиме сортировщика.

8.3.4. Показ презентации

В большинстве случаев презентация готовится для показа с использованием компьютера. Именно при таком показе можно реализовать все преимущества электронных презентаций.

В случае маленькой аудитории показ может осуществляться просто на экране монитора компьютера. Для больших аудиторий применяются большие мониторы или проекционные аппараты. Эти устройства имеют разную разрешающую способность, яркость и контрастность, поэтому, готовя презентацию, нужно учитывать возможности устройств, на которых она будет показана.

При работе с многочисленной аудиторией может потребоваться дополнительный монитор, работу с которым поддерживают Windows 98 или Windows 2000. При этом на основном мониторе будут появляться слайды презентации, а на вспомогательном — ее отображение в обычном трехпанельном режиме. Это предоставляет докладчику возможность пользоваться во время показа своими заметками и комментариями.

Существует несколько способов запустить показ презентации:

- в проводнике Windows щелкнуть правой кнопкой мыши на файле презентации PowerPoint и выбрать в меню команду Показать. Начнут показываться слайды презентации в режиме Показа слайдов. После завершения презентации вы опять окажитесь в проводнике Windows. Этот способ запуска наиболее удобен, когда вы пришли со своей готовой презентацией на чужой компьютер для демонстрации;
- в PowerPoint в любом из режимов можно выделить слайд, с которого начнется презентация, и щелкните по кнопке Показ слайдов в левом нижнем углу. Этот способ наиболее удобен для репетиции презентаций;
- выполните команду Начать показ в меню Показ слайдов или Показ слайдов в меню Вид.

Вне зависимости от способа запуска презентации на экране появится первый слайд. Существует несколько способов перехода от слайда к слайду при показе презентации:

- можно задать временные интервалы, через которые будут сменяться слайды. Ручная установка интервалов осуществляется следующим образом: необходимо выделить соответствующий слайд в режиме Сортировщика слайдов, затем щелкнуть мышью на кнопке Смена слайдов меню Показ слайдов. В появившемся диалоговом окне, изображенном на рис. 8.10, включите Автоматически после и укажите интервал, через который появится следующий слайд;

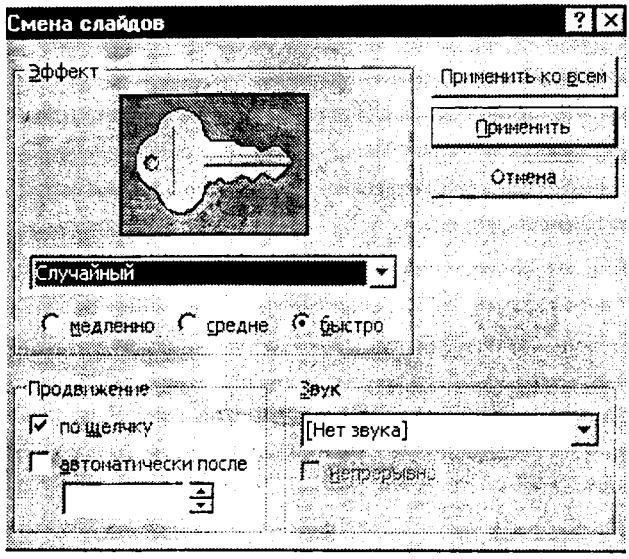


Рис. 8.10. Диалоговое окно Смена слайдов

- можно менять слайды вручную. Для этого можно пользоваться мышью или клавиатурой. При использовании мыши для смены слайда достаточно щелкнуть левой кнопкой. При использовании клавиатуры для этого надо нажать на Пробел или клавиши ←, →, Page Down, Enter;
- также для перехода к конкретному слайду или другой презентации во время демонстрации и запуска других приложений или для перехода на страницу в Интернете служат управляющие кнопки. Для создания управляющих кнопок необходимо в Обычном режиме отобразить тот слайд, в котором должен быть создан переход, а затем выбрать в меню Показ слайдов ▸ Управляющие кнопки и отметить один из предлагаемых вариантов. Если переход должен происходить по щелчку мыши, что предпочтительно при показе презентации, то надо перейти на панель По щелчку мыши. В зависимости от того, куда осуществляется переход, надо выбрать один из предлагаемых элементов раскрывающегося списка Перейти по гиперссылке. Элемент Слайд позволяет перейти к назначаемому слайду текущей презентации, элемент Другая презентация PowerPoint — к другой созданной презентации, переход к другому типу файла выполняется элементом Другой файл, а к странице в Интернете — элементом Адрес URL.

Во время демонстрации презентации в левом нижнем углу экрана появляется кнопка со стрелкой вверх, нажатие этой кнопки открывает контекстное меню, в

котором содержатся все необходимые команды управления. Например, при помощи пункта Указатель можно выбрать отображение указателя либо совсем скрыть его. По умолчанию он отображается в виде стрелки, но иногда бывает полезно переключиться в режим Перо. При помощи Пера можно снабжать слайды примечаниями или выделять какие-то области слайда.

8.3.5. Публикация презентации

Как отмечалось ранее, можно распечатать презентацию на бумаге, на цветных или черно-белых прозрачных пленках (или так называемых прозрачках), на 35-миллиметровых слайдах, а также опубликовать ее в Интернете.

Печать презентации на бумаге производится в тех случаях, когда презентация предназначена для неофициальной встречи или если требуется дополнительная раздача материала при показе электронной презентации. Прежде чем начать печать, полезно еще раз проверить настройку параметров страницы. Для этого выбирается команда из меню Файл ▸ Параметры страницы. В открывшемся диалоговом окне Параметры страницы в раскрывающемся списке Размер слайдов вводится подходящий размер носителя. Печать на бумаге выполняется из меню Файл ▸ Печать. При этом открывается диалоговое окно Печать, изображенное на рис. 8.11. В этом

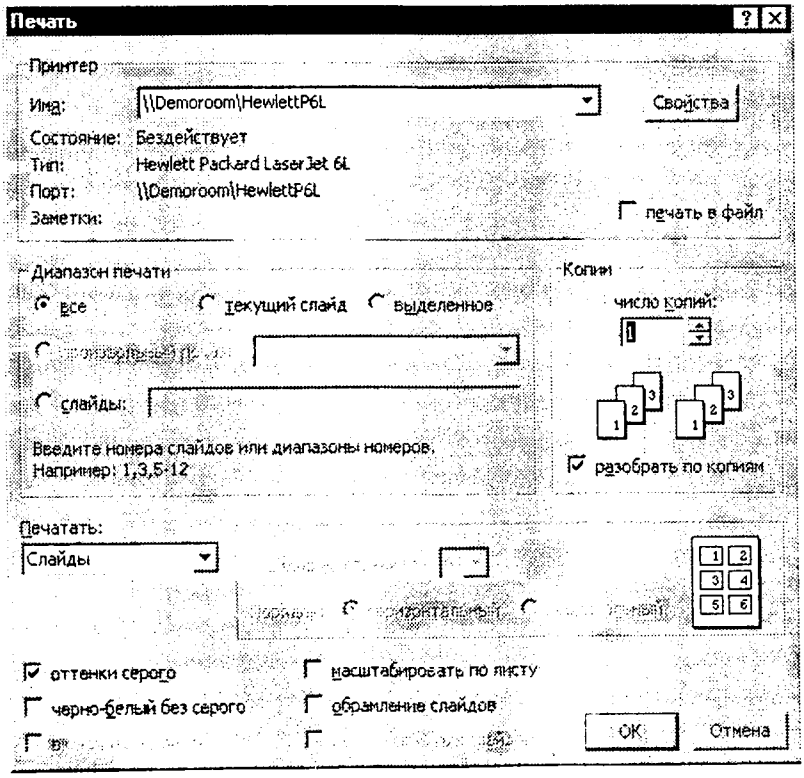


Рис. 8.11. Диалоговое окно Печать

окне назначается принтер, подключаемый для печати, число копий, печатаемые оформительские эффекты и скрытые слайды. В области Диапазон печати указываются конкретные слайды, выводимые на печать. Предоставляется возможность печати всей презентации, текущего слайда, выделенных слайдов, произвольной демонстрации или конкретного диапазона слайдов. Из раскрывающегося списка Печать выбирается тип печатаемых материалов. Допускается печать слайдов, материалов для выдачи, заметок и структуры презентации.

Печать на прозрачных пленках. Для придания презентации более профессионального вида можно вывести ее на прозрачные пленки на струйном или лазерном принтере. На рынке представлено несколько типов прозрачек, среди которых наиболее распространены пленки «3M Inkjet Printer Transparencies» и «3M Scotch Laser Printer Transparencies». Перед тем как печатать слайды для проектора, лучше изменить несколько параметров. Для этого надо снова вызвать диалоговое окно Параметры страницы. По умолчанию слайды, как и на бумаге, выводятся на печать в альбомной ориентации. Однако при этом некоторые модели проекторов срезают боковые части слайда. Для того чтобы этого избежать, лучше переопределить ориентацию печати. Для этого следует установить переключатель Книжная в области Слайды диалогового окна. После этого надо обязательно просмотреть слайды перед выводом на печать и, при необходимости, отрегулировать размер и положение объектов. На прозрачных пленках слайды можно печатать в оттенках серого, черно-белыми или цветными, так же, как и на бумаге. После выполнения всех необходимых настроек выполняется команда Файл ▸ Печать. Чтобы напечатать цветную презентацию на прозрачной пленке с использованием цветного принтера, необходимо выбрать модель из раскрывающегося списка Имя в области Принтер диалогового окна Печать. В результате получится набор высококачественных пленок, с которыми удобно работать на проекторе.

Использование 35-миллиметровых слайдов. Наилучшим способом представления презентации большой аудитории является проецирование 35-миллиметровых слайдов на большой экран. Создание таких слайдов является высококвалифицированной работой и осуществляется в специальных мастерских или бюро услуг. Для этого необходимо сохранить презентацию на дискетах, желательно в формате RTF, после чего отослать их в бюро услуг. Предварительно бюро услуг должно предоставить заказчику специальный драйвер, который для этого понадобится, а также инструкцию по подготовке файлов. В диалоговом окне Параметры страницы надо выбрать из списка Размер значение Слайд 35 мм, чтобы элементы презентации имели подходящий размер для изготовления слайдов, а также задать другие параметры, требуемые бюро услуг.

Публикация презентации в Интернете. В PowerPoint 2000, как и в других приложениях MS Office 2000, основным нововведением стало радикальное изменение порядка работы с Web-документами. Смещение акцента в сторону Web благоприятно отразилось на PowerPoint. Как и в других офисных приложениях, в этой программе сохранение в формате HTML стало неотъемлемой составляющей. Презентация в формате HTML окажется непосредственно на Web-сервере без участия системного администратора. Эта возможность реализована при помощи нового пункта меню Файл ▸ Сохранить как Web-страницу. При выборе этого пункта появляется диа-

логовое окно Сохранение документа, в котором в раскрывающемся списке Тип файла и предлагается сохранить документ как Web-страницу.

Ничего сложного и нового, по сравнению со стандартным сохранением документа, здесь нет, за исключением того, что:

- во-первых, при создании документа необходимо применять элементы форматирования и специальные эффекты, обычно используемые для Web-страниц;
- во-вторых, в данном диалоговом окне предусмотрена также возможность изменения заголовка страницы, появляющегося при просмотре презентации. Как правило, этим следует пользоваться для привлечения внимания зрителей (что, собственно говоря, и является главной целью докладчика).

Не имеет смысла публиковать презентацию, не представляя себе, как она будет смотреться при публикации. Для того чтобы просмотреть, как будет выглядеть презентация, как Web-страница, надо выбрать в меню Файл ▸ Предварительный просмотр Web-страницы. Причем желательно это сделать до того, как сохранять презентацию таким образом, с тем, чтобы сразу отредактировать ее для аудитории Интернета.

8.4. Презентационное оформление экономической документации средствами MS PowerPoint

Решение задач презентационного представления экономической документации в целом также мало чем отличается от представления иных научных и коммерческих трудов. Основная специфика заключается в том, что в представлении экономических документов зачастую ведущую роль играют такие специальные средства, как таблицы, списки и разного вида диаграммы. Маркированные списки вообще являются распространенным средством, включаемым в оформление любой презентации. Естественно, при представлении экономических, как и любых других документов, требуются все возможности ввода и редактирования, но большую часть мы рассмотрели ранее, поэтому сконцентрируем внимание на ранее указанных особенностях презентации экономических трудов. Возьмем для примера тот же упрощенный гипотетический *инвестиционный проект*, оформление которого осуществлялось средствами текстового процессора MS Word 2000, *Проект завершения строительства станции метрополитена «Адмиралтейская»*, и создадим для него пример презентации, целью которой является представление проекта широкой аудитории и, в конечном итоге, привлечение инвестиций. При всем уважении к нашему труду, хотим обратить особое внимание читателя на то, что эта презентация именно **гипотетическая**, то есть в реальном представлении инвестиционных проектов и прочих серьезных разработок таким оформлением пользоваться, естественно, не стоит. Обилие всяческих эффектов и картинки, иногда с трудом выдерживающие критику, были применены, прежде всего, для наглядности и иллюстрации возможностей PowerPoint, хотя некоторые из них могут быть применены в том числе и в таком виде.

Для этого оформим в Word ключевые моменты проекта, которые мы хотим обратить в заголовки слайдов, стилем Заголовок1, а то, что хотели бы взять за основу

содержимого слайдов — стилями Заголовок2 и более низкого уровня. Затем надо перейти в PowerPoint и выбрать команду Создать из меню Файл. В появившемся диалоговом окне лучше выбрать вариант авторазметки, который в дальнейшем будет применяться при создании презентации. Потом для некоторых слайдов мы будем менять Разметку слайда в зависимости от его содержимого. В открывшейся пустой презентации выберем в меню Вставка ▸ Слайды из структуры. В открывающемся диалоговом окне Вставить структуру выбираем нужный нам документ Word и щелкаем мышью на кнопке Вставить. PowerPoint перенесет структуру нашего документа и оформит ее в виде слайдов. Теперь выберем в меню Формат ▸ Применить шаблон оформления один из предлагаемых шаблонов. Для данной презентации нам показался подходящим шаблон Капсулы. Теперь можно приступать к редактированию содержимого каждого слайда. При перенесении документа Word программа PowerPoint оставляет титульный слайд пустым с тем, чтобы докладчик мог оформить его самостоятельно, и вставляет структуру, начиная со второго слайда.

Введем для титульного слайда заголовок: «Концепция строительства станции метрополитена "Адмиралтейская"» и поэкспериментируем с изменением формата шрифта и границ местозаполнителя заголовка с тем, чтобы первый слайд презентации выглядел наиболее привлекательно (рис. 8.12).

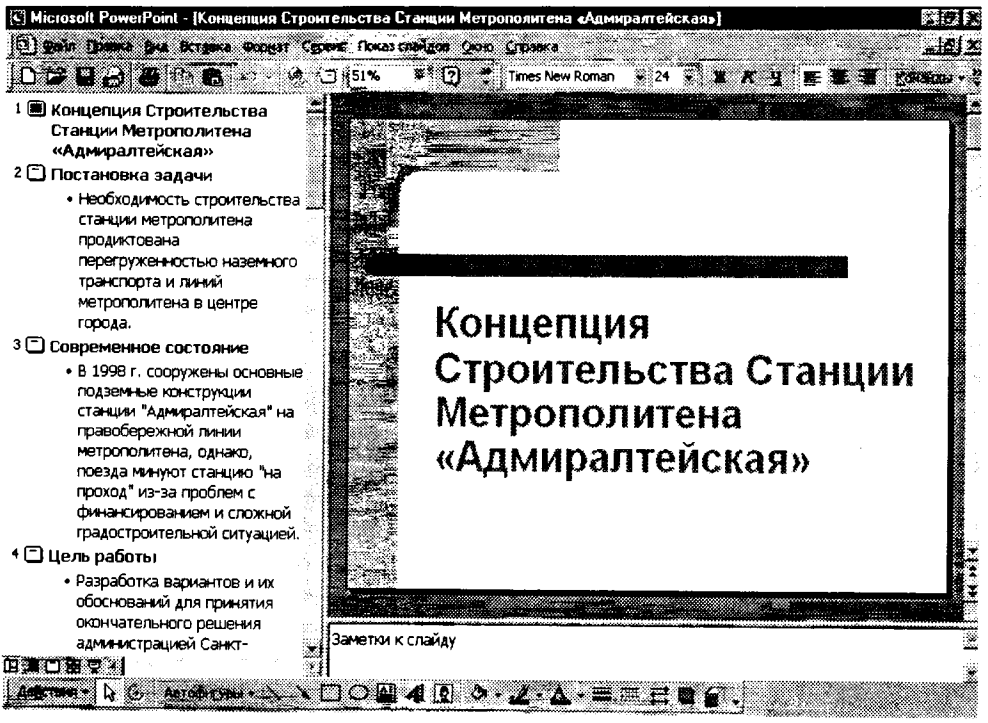


Рис. 8.12. Титульный слайд и структура презентации

Для оформления второго слайда презентации (рис. 8.13) мы не стали применять особых приемов, а лишь выделили ключевое слово в тексте слайда при по-

мощи меню Формат ▸ Шрифт, где поменяли размер и начертание шрифта, а также задали атрибут Тень для большей выразительности и наглядности.

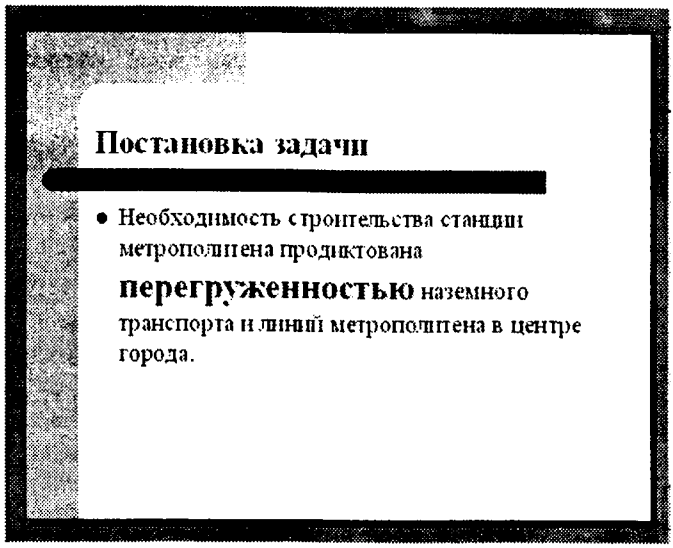


Рис. 8.13. Второй слайд презентации с изменением формата шрифта и теневым эффектом

При редактировании третьего слайда (рис. 8.14), помня конечную цель нашей презентации — привлечения инвестиций к проекту, — внизу слайда мы вставили объект Word Art **Необходимо финансирование**, выбрав для него контрастный цвет, с тем, чтобы сконцентрировать внимание потенциальных инвесторов на данной проблеме. Объект Word Art вставляется с помощью панели инструментов Рисование.

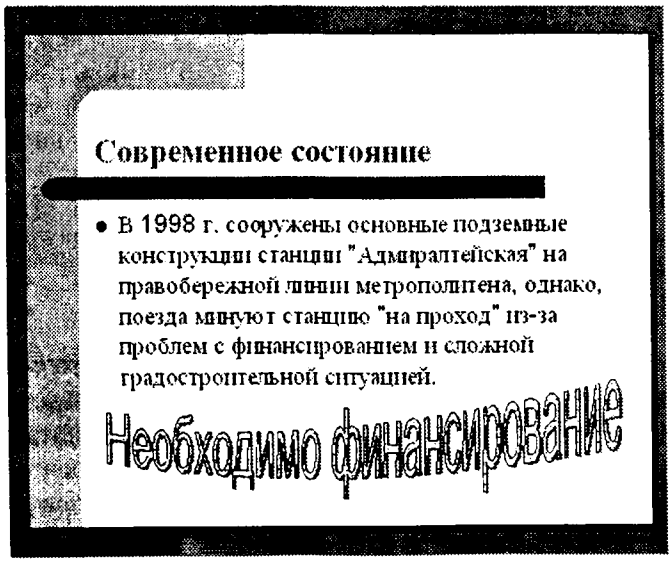


Рис. 8.14. Третий слайд презентации со вставкой объекта WordArt

Следующий слайд, который сопровождает ту часть доклада, которая посвящена разработке вариантов строительства, мы снабдили, для большей наглядности, автофигурой, символизирующей множество существующих вариантов завершения строительства станции метрополитена (рис. 8.15). Автофигуры доступны из команд панели инструментов Рисование ▸ Автофигуры. Здесь содержатся несколько блоков разнообразных графических изображений.

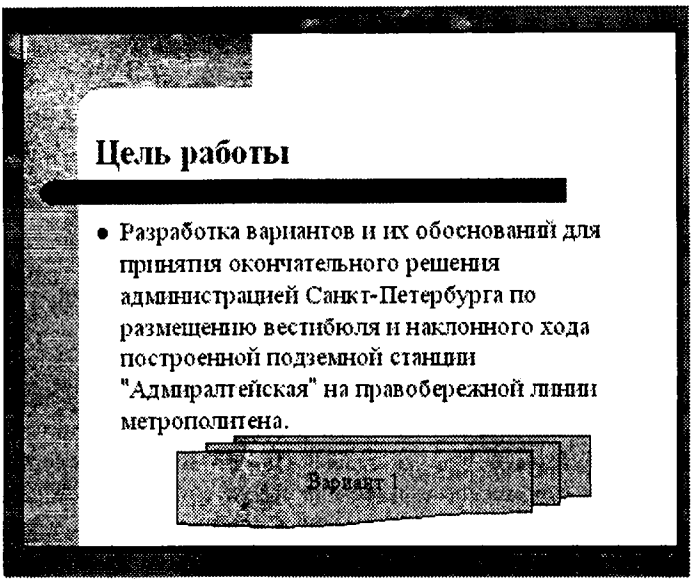


Рис. 8.15. Четвертый слайд со вставкой автофигуры

На пятом слайде презентации для привлечения внимания к методу реализации цели разработки инвестиционного проекта был применен видеоклип, взятый из меню Вставка ▸ Объект ▸ Видеоклип (рис. 8.16). Команды меню Вставка ▸ Объект позволяют вставлять объекты, созданные другими приложениями MS Office или другими OLE-серверами. В частности, экономистам будет удобно использовать объекты MS Excel — таблицы и диаграммы или организационную диаграмму MS Organization Chart, для которой в PowerPoint создана специальная разметка слайда.

Помимо этого для пятого слайда была изменена разметка: выбрана разметка слайда Текст и графика.

Для шестого слайда так же, как и для пятого, была изменена разметка с выбором другого варианта: Графика и текст. Вместо заполнителя Графика был вставлен рисунок из меню Вставка ▸ Рисунок ▸ Картинки для того, чтобы оживить презентацию и обратить внимание аудитории на многообразие вариантов строительства и сложность оценки каждого из них с целью последующего выбора конкретного варианта (рис. 8.17). Этот рисунок, в отличие от автофигуры, например, является растровым объектом, поэтому он может искажаться при масштабировании и печати.

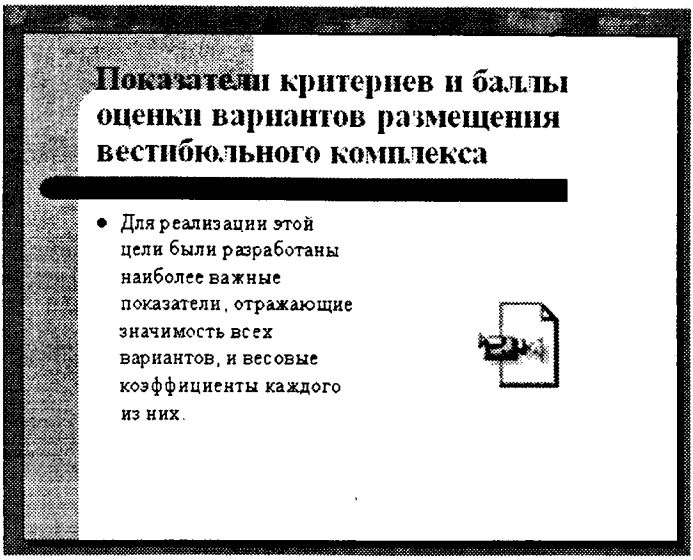


Рис. 8.16. Пятый слайд презентации с изменением разметки слайда и применением видеоклипа

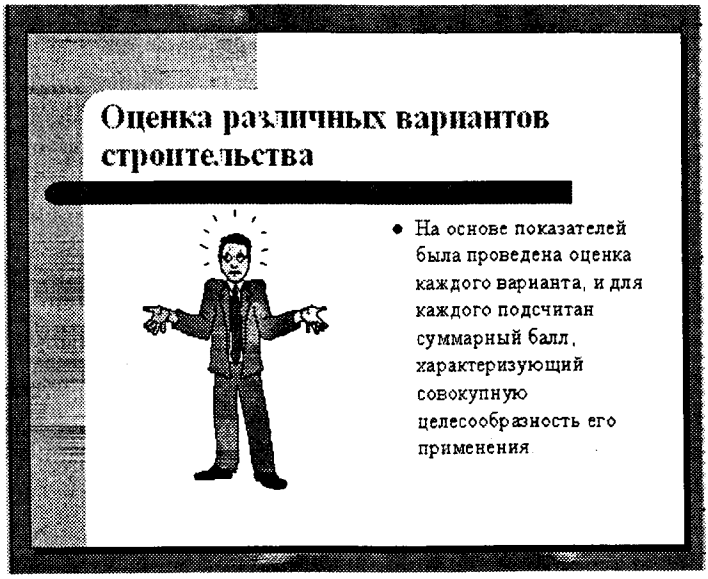


Рис. 8.17. Шестой слайд презентации с изменением разметки слайда и вставкой рисунка

На седьмом слайде имеется другой распространенный элемент оформления — таблица, — который часто используется для представления экономической информации (рис. 8.18). Для создания слайда с таблицей существует специальный вариант разметки: Таблица. Если таблица используется в качестве элемента оформления презентации, она, в отличие от таблиц, с которыми оперируют другие приложения

MS Office, не должна содержать большого количества строк и столбцов, иначе будет потеряна наглядность и информативность. Как правило, лучше всего использовать таблицы с четырьмя-пятью столбцами и двумя-тремя строками.

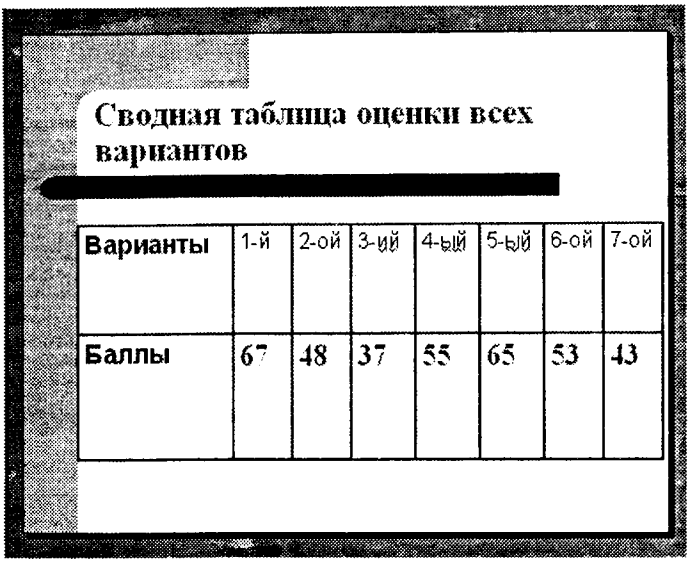


Рис. 8.18. Седьмой слайд презентации с изменением разметки и вставкой таблицы

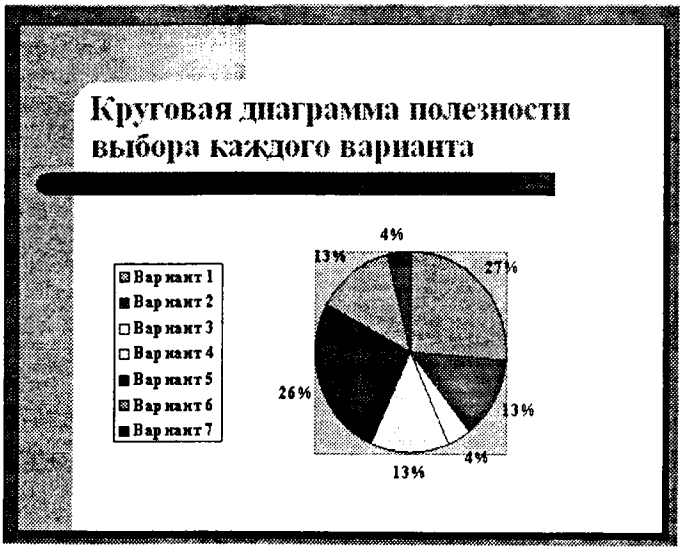


Рис. 8.19. Восьмой слайд с изменением разметки и размещением диаграммы

Следующий слайд отображает данные таблицы в виде диаграммы (рис. 8.19). Диаграммы на слайдах презентаций способны быстро и наглядно показать те тенденции и соотношения, которые докладчик хочет донести до аудитории. Для

вставки диаграммы предназначены три варианта разметки слайда: на одном из них присутствует только диаграмма, на двух других — комбинации диаграммы и текста. Мы выбрали для данного случая тот вариант разметки, на котором слева содержится текст, а справа — соответствующая диаграмма. Проще всего вставка диаграммы осуществляется в меню Вставка ▸ Диаграмма.

На девятом слайде мы решили применить управляющую кнопку для возможного перехода, например, к седьмому слайду, содержащему наглядную информацию о совокупной оценке каждого варианта строительства, для того чтобы напомнить аудитории сложившуюся ситуацию (рис. 8.20). Об управляющих кнопках и их предназначении было рассказано в 8.3.4.

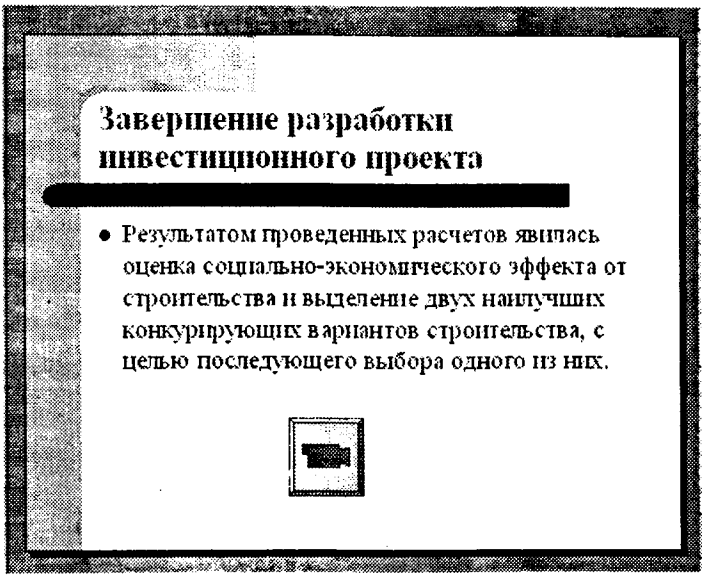


Рис. 8.20. Девятый слайд с использованием управляющей кнопки

На десятом слайде, посвященном оценке социально-экономического эффекта, мы решили применить один из наиболее традиционных способов представления информации на слайдах — нумерованный список (рис. 8.21). Нумерованные списки создаются с помощью команды меню Формат ▸ Список, которой вызывается диалоговое окно содержащее две панели: Маркированный и Нумерованный. Для того чтобы создать список, необходимо выделить те абзацы, которые мы хотим сделать элементами списка, затем вызвать это диалоговое окно и сделать активной соответствующую панель, в данном случае Нумерованный.

На следующем слайде мы применили маркированный список, создаваемый тем же способом, но при другой активной панели данного диалогового окна (рис. 8.22). Эта панель, помимо имеющихся на панели Нумерованный возможностей выбора цвета маркера и размера его в процентном соотношении к шрифту, позволяет выбрать в качестве маркеров не только содержащиеся на этой панели маркеры. Также можно использовать и специальные символы, выбираемые из диалогового окна, появляющегося при нажатии на кнопку Знак, и рисунки из файла при нажатии на кнопку Рисунок.

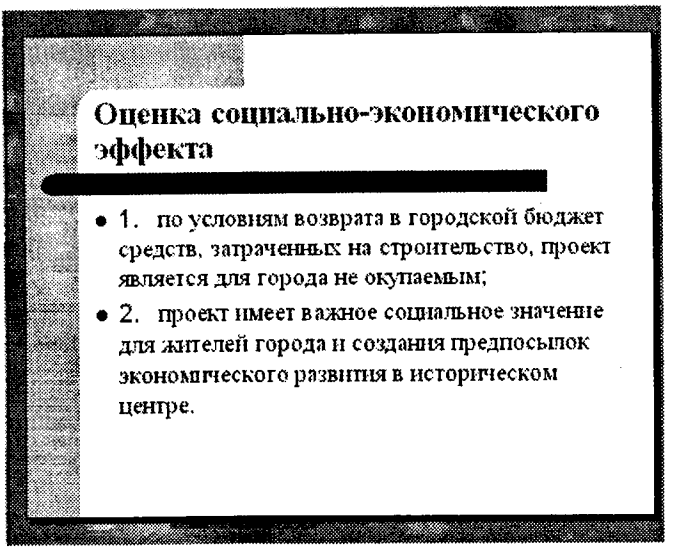


Рис. 8.21. Десятый слайд с применением нумерованного списка

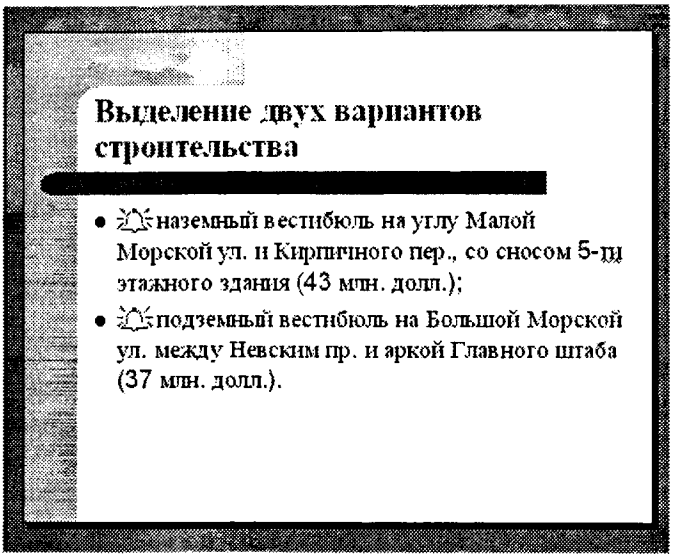


Рис. 8.22. Одиннадцатый слайд с применением маркированного списка

На заключительном слайде, который является завершающим моментом презентации, мы использовали три элемента оформления (рис. 8.23). Во-первых, выделили текст слайда рамкой, во-вторых, применили различные стили шрифта для выделения ключевых слов, в-третьих, поместили на слайд звуковой эффект, чтобы сделать завершение презентации более занимательным и убедительным.

Для обрамления текста рамкой можно применять различные способы, например использовать автофигуры из панели инструментов Рисование. Добавление звука

осуществляется в меню Вставка ▸ Фильм и звук путем выбора или готового звука из коллекции Clip Gallery (команда Звук из коллекции), или звука из существующего файла (команда Звук из файла).

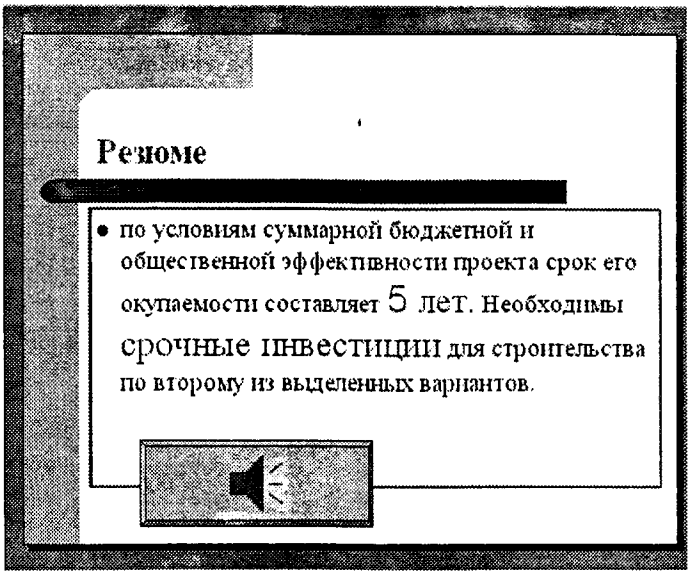


Рис. 8.23. Заключительный слайд презентации с изменением размера шрифта и звуковым эффектом

Ключевые понятия

- презентация;
- слайд;
- растровый и векторный форматы;
- местозаполнитель;
- авторазметка;
- автофигура;
- режимы работы с презентацией;
- шаблоны презентаций;
- публикация.

Контрольные вопросы

1. Каковы основные этапы работы с презентацией?
2. Какие редакторы презентаций вы знаете?
3. В чем отличие рисунков от изображений и для чего они применяются?

4. В чем преимущество логического форматирования перед физическим?
5. Сколько режимов просмотра документа в PowerPoint и какова их специфика?
6. Какие возможности по созданию презентаций заложены в PowerPoint и в чем их различия?
7. Для чего применяются структуры в PowerPoint?
8. Для чего создаются управляющие кнопки?
9. Каким образом можно при показе презентации воспользоваться заметками?
10. Какими способами можно опубликовать презентацию?

Литература

1. Microsoft Office 2000: справочник/Под ред. Ю. Колесникова. СПб.: Питер, 1999.
2. PC World. 1999–2000 гг.
3. Стоцкий Ю. Самоучитель Office 2000. СПб.: Питер, 1999.
4. Хэлворсон М., Янг М. Эффективная работа с Microsoft Office 2000. СПб.: Питер, 2000.

Глава 9. Программное обеспечение управления проектами

В данной главе мы рассмотрим программное обеспечение, применяемое при решении задач организационно-управленческой деятельности предприятий. Очевидно, что эта область является чрезвычайно широкой и разнообразной. Поэтому в рамках настоящего учебника представляется возможным остановиться только лишь на отдельных ее сторонах. Другой отличительной особенностью этой главы является то, что в ней описанию пользовательского интерфейса конкретного программного обеспечения уделено несколько больше места, чем это делалось в предыдущих главах. Последнее сделано прежде всего потому, что справочная литература по программным средствам данного класса встречается редко.

9.1. Понятие о проектах и основных методах управления ими

9.1.1. Программные средства автоматизации процессов делопроизводства и бизнес-планирования

Не вызывает сомнений тот факт, что задачи, решаемые в ходе организации работы фирм и предприятий, существенно зависят от их отраслевой и территориальной принадлежности, вида, формы собственности, размера и других параметров. Однако можно выделить и много общих черт, позволяющих провести классификацию как самих задач, так и соответствующих программных средств, применяемых при их решении. Можно выделить два класса задач:

- задачи автоматизации процессов делопроизводства и документооборота;
- задачи управления бизнес-проектами.

В повседневной жизни мы повсеместно сталкиваемся с проблемами, порождаемыми неправильным и несвоевременным оформлением документов, их утратой

или опозданием в получении. Каждый из нас может припомнить случаи из жизни, когда приходилось терпеть неудобства из-за искажений данных в документах или же невозможности их корректной интерпретации. Видимо, нерадивость или злая воля управленческого персонала, готовящего и оформляющего необходимые нам бумаги, не могут быть единственной причиной перечисленных трудностей. В значительной мере они порождаются причинами, связанными с несовершенством организации процесса делопроизводства. Одним из путей их преодоления является активное внедрение в деятельность предприятий, фирм и организаций *систем автоматизации офисной деятельности.*

В предыдущих главах было рассмотрено программное обеспечение, позволяющее создавать электронные документы (Word — электронные аналоги текстов, Excel — электронные таблицы, Power Point — электронные презентации и т. д.). ПО автоматизации делопроизводства и документооборота, соответственно, осуществляет логическое развитие преимуществ, заложенных в системах создания электронных документов, а именно, они организуют автоматическое или автоматизированное взаимодействие документов, создаваемых и редактируемых в рамках коллективной работы группы пользователей, решающих взаимозависимые задачи. Технологии, заложенные в основу систем автоматизации делопроизводства, позволяют «направлять» деятельность того или иного сотрудника в соответствии с логикой движения тех документов, с которыми он должен работать, а также открывают принципиально новые возможности для его руководителей по контролю и анализу информационных потоков.

Одновременно необходимо отметить, что подробный и содержательный разговор о программных продуктах для поддержки коллективной работы группы пользователей (англ. *groupware*) невозможен без знакомства с технологиями работы вычислительных сетей, как локальных, так и глобальных. Эти вопросы будут рассмотрены в главах 10 и 11, поэтому здесь мы ограничимся лишь самой общей характеристикой ПО автоматизации делопроизводства и офисной деятельности.

Современная среда коллективной работы, как правило, помимо перечисленных выше функций реализует функции встроенной почтовой среды, синхронизации файлов, расширенных механизмов поиска, коллективного редактирования и организации телеконференций.

Среди конкретных систем данного класса в первую очередь могут быть названы Lotus Notes (фирма Lotus), Microsoft Exchange (фирма Microsoft), Novell GroupWise (фирма Novell), SiuteSpot (фирма Netscape). По имеющимся на настоящий момент данным, на их долю приходится 80% рынка программного обеспечения для рабочих групп[1].

Разработки фирмы Lotus исторически были первыми в данной сфере. Изначально (в 1989 г.) ею была выпущена версия программного обеспечения Lotus Notes, рассчитанная на работу в рамках больших корпоративных сетей (минимально допустимая лицензия на 200 пользователей). В дальнейшем появились варианты данного ПО существенно меньшего масштаба и стоимости, ориентированные на

[1] См., например, http://www.lotus.com.

более широкий круг потребителей. К числу последних разработок, представленных Lotus, относится Lotus Domino 4.5. Он представляет собой сервер приложений и сообщений с интегрированным набором услуг, позволяющих легко и просто создавать безопасные интерактивные деловые решения для глобальных и корпоративных сетей. Серьезным конкурентом Lotus, особенно на российском рынке, являются системы организации офисной деятельности, построенные на основе программного обеспечения фирмы Microsoft — Microsoft Exchange. Их безусловно сильно стороной является тесная интеграция с операционными системами серии Windows и программными продуктами, входящими в семейство Microsoft Office.

Другим видом программного обеспечения, как уже упоминалось, применяемым при решении задач управления фирмой или предприятием, является ПО управления проектами. Изучением его свойств и особенностей мы и займемся в настоящей главе.

9.1.2. Общие проблемы теории управления проектами

В узком смысле под термином *проект* понимают набор документов конструкторского плана, описывающих некоторое изделие или техническое решение. Однако в последние десятилетия это понятие используют в расширительном плане, а именно, как совокупность (комплекс) мероприятий, в результате выполнения которых к установленному сроку при ограниченных материальных ресурсах должна быть достигнута заранее определенная система целей. *Характеристическим свойством проекта* является то, что он представляет собой комплекс работ (операций), которые, как правило, не повторяются.

Вообще говоря, с позиций современных теорий научного менеджмента процесс планирования и управления проектами может быть рассмотрен на различных уровнях:

- во-первых, с точки зрения *стратегического управления* фирмой (как самостоятельным экономическим субъектом), в рамках которой реализуется проект. Под стратегическим понимается такое управление, которое ориентировано на перспективу, сосредоточено на выработке глобальных целей и направлений развития и, как правило, имеет горизонт планирования в несколько лет;
- во-вторых, с точки зрения оперативного управления, то есть управления, ориентированного на обеспечение устойчивого функционирования фирмы, создание потенциала для ее развития, обычно во временном окне от месяца до года;
- в-третьих, с точки зрения текущего управления, основу которого составляет регистрация, сбор и анализ информации об отклонениях управляемого процесса от запланированного состояния и последующая выработка и реализация решений по устранению (минимизации) нежелательных отклонений.

В дальнейшем речь пойдет исключительно о методах, применимых к решению задач управления проектами на оперативном и текущем уровнях, или, как еще говорят, задач *оперативно-календарного планирования и управления*. Параллель-

но следует обратить внимание читателя на то, что проблемы разработки и изучения соответствующих методов составляют предмет целого ряда экономических и экономико-математических дисциплин, и прежде всего исследования операций. Однако любой алгоритм, основанный на той или иной экономико-математической модели, приобретает практическое значение лишь тогда, когда реализуется в виде конкретного программного инструмента. И в этом плане задачи управления проектами составляют предмет для изучения со стороны экономической информатики. Также существенное перекрытие «сфер интересов» экономико-математических методов оперативно-календарного планирования и экономической информатики связано с проблемами организации информационной базы для программного обеспечения управления проектами.

Любое управление явно или неявно подразумевает наличие целевой функции, позволяющей оценивать его результаты и эффективность. В соответствии с традиционными подходами цели оперативно-календарного планирования определяются как минимизация продолжительности выполнения проекта при ограничениях на имеющиеся ресурсы.

Существенным рубежом в развитии методов календарного планирования и управления стала середина 50-х годов двадцатого века. Собственно говоря, тогда они и оформились в качестве самостоятельной области прикладных и теоретических исследований. В 1956–1958 гг. почти одновременно двумя независимыми группами разработчиков были созданы *метод критического пути* — МКП (фирма E.I. du Pont de Nemours & Company) и *метод оценки и пересмотра программ* — PERT[1]. Последний был разработан по заказу оборонного ведомства США для проведения проектно-конструкторских работ по ракете «Поларис». В обоих методах много общего. Существенным отличием метода PERT было то, что в нем продолжительность отдельных работ (операций) рассматривалась как случайная величина. К настоящему времени оба метода составляют основу теоретического аппарата метода *сетевого планирования и управления проектами*.

Сетевое планирование и управление включает три основных этапа: *структурное планирование*, *календарное планирование* и *оперативное управление*.

В рамках этапа **структурного планирования** производится разбиение проекта на отдельные операции. Под операцией понимается деятельность или процесс, выполнение которых требует некоторых временных и материальных затрат. Затем составляется логическая схема связей между операциями, или, как еще говорят, *сетевая модель*.

На этапе **календарного планирования** строится так называемый *календарный график*, определяющий моменты начала и окончания каждой операции. Календарный график в совокупности с сетевой моделью позволяет выявить критические операции, то есть такие операции, изменение длительности которых может повлиять на срок завершения проекта в целом.

Завершающим этапом является **оперативное управление** процессом реализации проекта. Данный этап подразумевает использование сетевой модели и календар-

[1] От англ. Project Evaluation and Review Technique — PERT.

ного графика для формирования отчетной информации о ходе выполнения проекта, с одной стороны, и их возможные корректировки по результатам анализа фактического положения дел, с другой.

Логическую последовательность операций удобно иллюстрировать с помощью графов[1]. В теории сетевого планирования наибольшее распространение получили *вершинные* и *стрелочные графы*.

В **стрелочных графах** операции представляются стрелками (дугами). Их направление отражает ход времени. Традиционно последовательность стрелок ориентируется слева направо. Начало и окончание операций называют событиями. События обычно изображают с помощью точек (кружков).

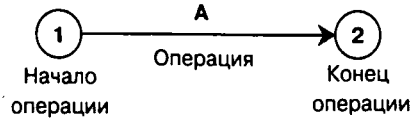


Рис. 9.1. Изображение операции на стрелочном графе

Простейший стрелочный граф изображен на рис. 9.1. На нем для обозначения операции используется буква A, а для событий — цифры 1 и 2. События принято нумеровать таким образом, чтобы номер предшествующего события был меньше, чем номер последующего. Событие не считается свершившимся, если не завершены все операции, отвечающие стрелкам, входящим в обозначающий его узел. Так, на рис. 9.2 отображена схема проекта, в котором операция C не может начаться ранее момента завершения операций A и B.

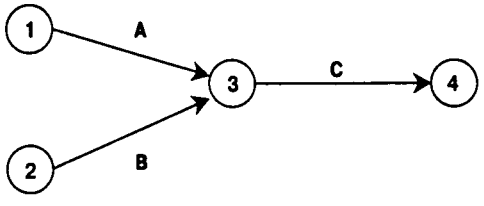


Рис. 9.2. Пример отображения последовательности операций с помощью стрелочного графа

Если некоторые события непосредственно не связаны между собой реальными общими операциями, но в силу причин содержательного характера не могут произойти одно без другого, то для описания их отношений используют *фиктивные логические операции*. Фиктивным операциям приписывается нулевая продолжительность. Например, если операция B непосредственно зависит от операции A, а C — от D, но D также нельзя начать, пока не закончена A, то данный проект может быть описан с помощью графа, показанного на рис. 9.3.

Если две и более операции имеют одни и те же начальное и конечное события, то для того чтобы их было удобно отличать друг от друга (последнее осо-

[1] На интуитивном (нестрогом) уровне граф может быть определен как объект, состоящий из вершин (узлов) и соединяющих их дуг (стрелок).

бенно важно при решении вопросов организации информационного обеспечения для соответствующих программных пакетов), используют так называемые фиктивные операции идентификации. Пример решения этой проблемы показан на рис. 9.4.

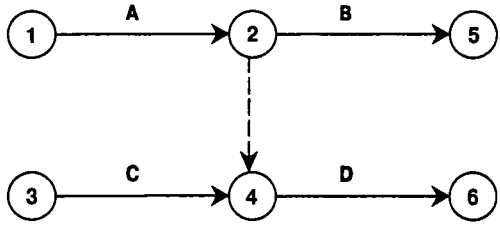


Рис. 9.3. Пример использования фиктивных логических операций

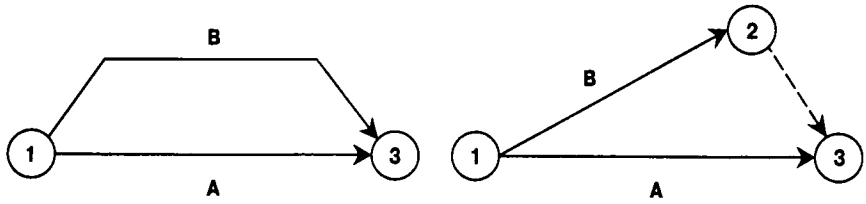


Рис. 9.4. Применение фиктивной операции идентификации

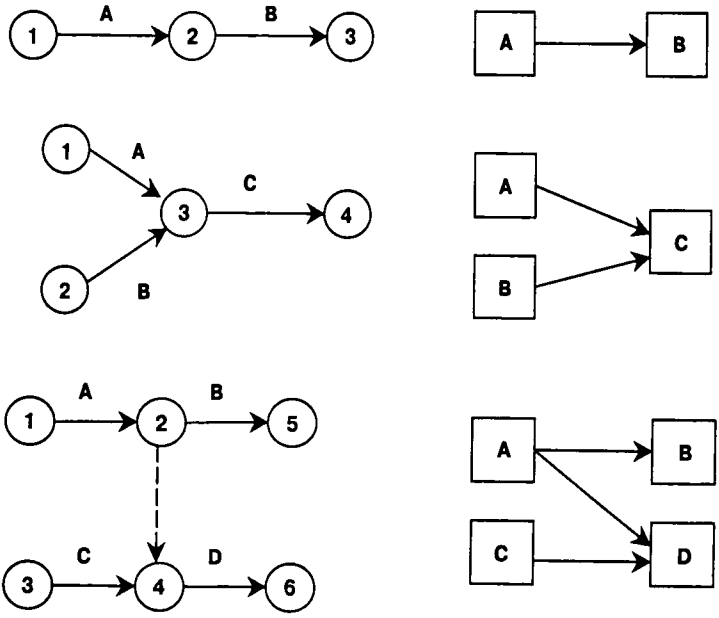


Рис. 9.5. Сравнение представлений схем проектов с помощью стрелочных и вершинных графов

В **вершинных графах** операции проекта представляются узлами (вершинами), а их взаимосвязи отражаются с помощью стрелок. К преимуществам вершинных графов следует отнести отсутствие в них необходимости вводить фиктивные операции, но, в то же время, при их использовании труднее представить общую картину перехода от одной операции к другой. На рис. 9.5 показаны альтернативные представления одних и тех же проектов с помощью стрелочных и вершинных графов.

Говоря о способах представления проектов, нельзя не упомянуть о таком известном инструменте, как *ленточный (линейный) график Гантта (диаграмма Гантта)*. В графике Гантта сроки начала и окончания операций откладываются на горизонтальной шкале времени. Соответственно, размеры линий графика, отражающих отдельные операции, пропорциональны их продолжительности. Недостаток данного метода заключается в том, что он не позволяет устанавливать зависимости между различными операциями. В то же время он хорошо отражает распределение ресурсов времени между отдельными частями проекта.

9.1.3. Программное обеспечение MS Project

В качестве конкретных представителей класса программ управления проектами могут быть названы такие системы, как TimeLine (TimeLine Inc.) и MS Project (Microsoft). В дальнейшем технологию работы с программными инструментами, автоматизирующими процесс планирования и управления проектами, мы будем рассматривать на примере программного пакета MS Project.

Первые версии системы управления проектами Project были разработаны компанией Microsoft для операционной системы MS DOS. В дальнейшем появились версии для Windows 3.x – MS Project v.3. и MS Project v.4. На настоящий момент широко эксплуатируется MS Project 98 (на ее базе будет построено дальнейшее изложение), и уже вышел MS Project 2000.

Необходимо отметить, что ПО MS Project непосредственно не входит в состав программного пакета MS Office и распространяется в виде отдельного продукта. Однако оно является членом семейства MS Office, то есть его интерфейс и основные концепции работы тесно согласованы с принципами и идеологией, заложенными в основу других офисных продуктов, таких как MS Word, MS Excel и т. д. Несомненным преимуществом MS Project является и то, что в нем широко используются возможности операционной среды MS Windows по организации совместной работы нескольких пользователей в рамках вычислительной сети.

Отличительной особенностью настоящей главы (по сравнению с предыдущими) является использование в изложении англоязычной версии ПО MS Project. Последнее объясняется тем, что на момент написания текста локализованной русскоязычной версии просто не существовало.

9.1.4. Пример задачи управления проектом

По аналогии с предыдущими главами, технологию применения программного обеспечения MS Project будет удобно рассмотреть на конкретном примере. В качестве такого примера возьмем задачу создания автоматизированной системы управления для некоторого предприятия среднего размера (например, фирмы, оказывающей консалтинговые услуги, или же фирмы, занимающейся торгово-закупочными мероприятиями по относительно небольшой номенклатуре товаров).

Для начала дадим содержательное описание ситуации. Итак, некоторое предприятие, занимающееся оказанием информационно-консалинговых услуг на финансовом рынке и имеющее в своем штате около 60 сотрудников, приняло решение о радикальной модернизации существовавшей у нее ранее системы управления. Базой для планируемых изменений является создание и ввод в действие новой автоматизированной системы. Для этого она обратилась к предприятию, специализирующемуся на разработке интегрированных программно-аппаратных решений в сфере управления (дадим ему условное название «Автоматика-Север»). Рассмотрим более подробно комплекс задач, стоящих перед менеджерами «Автоматики-Север» на момент разработки ими плана выполнения работ по будущему проекту (дадим ему условное «ТЕСТ»). Для проектов такого рода естественными представляются следующие этапы:

- A — обследование объекта автоматизации;
- B — разработка (по результатам выводов, полученных в ходе обследования) технического задания на создание автоматизированной системы;
- C — согласование с руководством фирмы-заказчика содержания технического задания и возможное внесение в него корректив;
- D — разработка по согласованному техническому заданию детального плана проведения работ;
- E — создание аппаратной (технической) базы автоматизированной системы;
- F — разработка программного обеспечения автоматизированной системы;
- G — опытная эксплуатация системы;
- H — обучение персонала работе с созданной системой;
- I — разработка и оформление технической документации;
- J — подписание акта о завершении работ и вводе автоматизированной системы в эксплуатацию.

Очевидно, что перечисленные этапы в силу причин технологического характера должны следовать в определенной последовательности друг относительно друга. При этом, допустим, работы по этапам E и F, а также этапам G, H и I могут вестись параллельно. Логика следования этапов может быть задана как с помощью вершинного графа, изображенного на рис. 9.6, так и с помощью стрелочного графа (рис. 9.7).

Обратим внимание на то, что в стрелочном графе для отражения того факта, что этапы G, H, I могут начаться лишь по завершении этапов E и F, используется фиктивная стрелка, соединяющая состояние 5 с состоянием 6. Аналогичным образом учтен и факт предшествования этапов G, H, I этапу J.

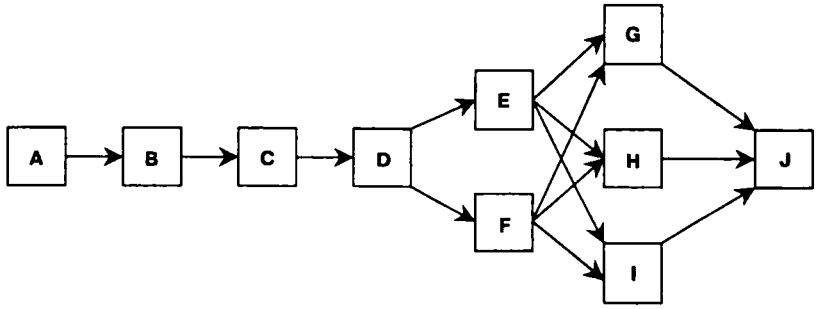


Рис. 9.6. Представление проекта «ТЕСТ» в виде вершинного графа

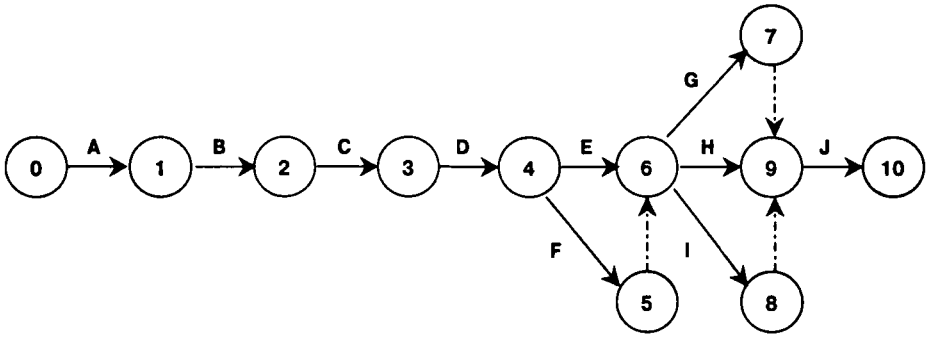


Рис. 9.7. Представление проекта «ТЕСТ» в виде стрелочного графа

9.2. Создание описания и графика проекта в MS Project

9.2.1. Ввод данных по проекту

Процесс работы с проектом в системе MS Project начинается с ввода данных (формирования описания проекта). Основными этапами этого процесса являются:

- определение опорных дат проекта;
- создание перечня операций с оценкой их продолжительности;
- формирование (при необходимости) иерархии операций;
- задание связей между операциями;
- создание списка ресурсного обеспечения;
- назначение ресурсного обеспечения операциям.

Данные, относящиеся к отдельному проекту, сохраняются в файле, имеющем расширение `*.mpp`. В частности, данные по описанному выше учебному примеру предлагается сохранить в файле `TestProject.mpp`.

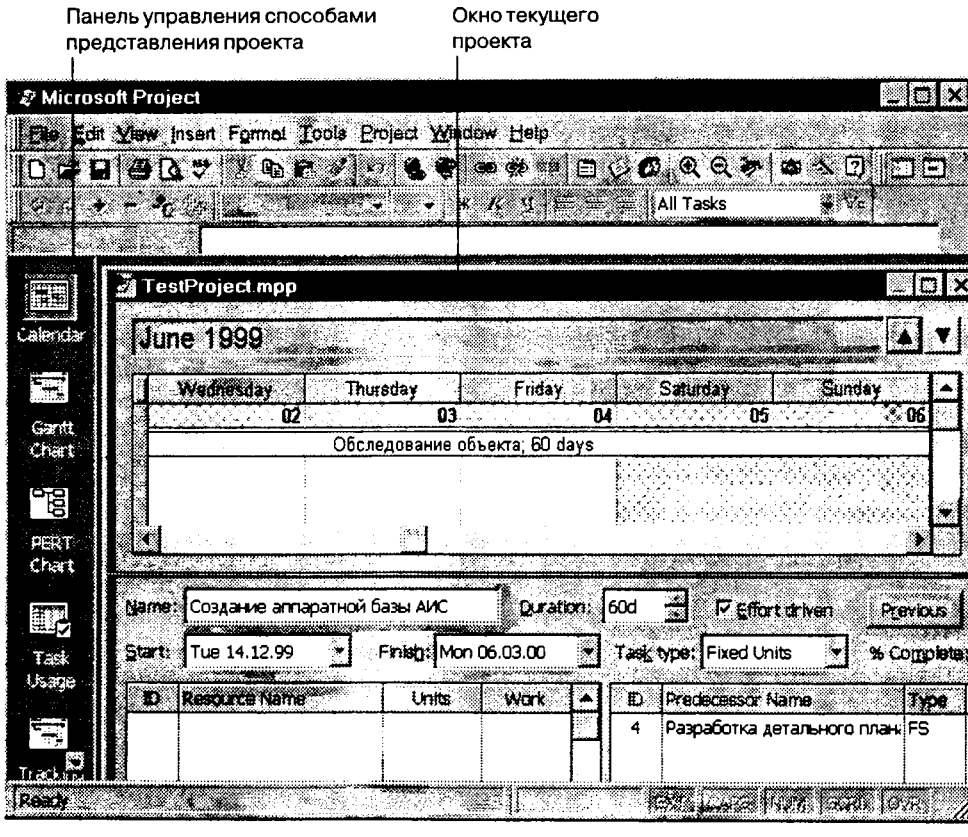


Рис. 9.8. Общий вид окна ПО MS Project

На рис. 9.8 представлен общий вид окна ПО MS Project. Читателю, знакомому с другими продуктами семейства MS Office, будет нетрудно сориентироваться в общих принципах организации пользовательского интерфейса данной программы. Как видно из рис. 9.8, основными элементами стандартной среды MS Project являются собственно окно проекта (вообще говоря, можно одновременно работать сразу с несколькими проектами и открыть несколько окон) и панель управления представлениями (видами) проекта. Альтернативный способ управления представлениями проекта доступен из меню View.

Ввод общей информации по проекту

Создание описания любого проекта (меню File ▸ New) начинается с ввода общей информации по проекту, важнейшими элементами которой являются опорные даты проекта. У каждого проекта определяются две опорные даты: начала и окончания. Одна из них (по выбору пользователя) вводится, а другая — вычисляется автоматически. Пример изменения общей информации по проекту показан на рис. 9.9.

Впоследствии общую информацию по проекту можно корректировать (меню Project ▸ Project Information).

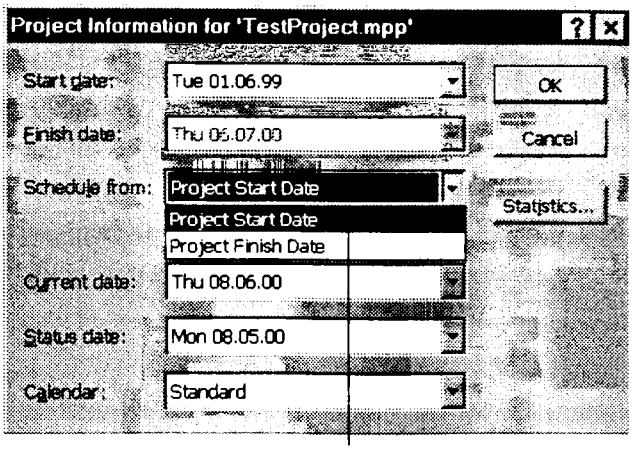


Рис. 9.9. Ввод общей информации по проекту

Создание перечня операций

Информация, позволяющая сформировать последовательность операций, получается как результат этапа структурного планирования (см. 9.1.2). Перечень операций, на которые разбивается проект, может быть создан вручную либо получен за счет переноса данных из других программ. Например, из документа Word или таблицы Excel.

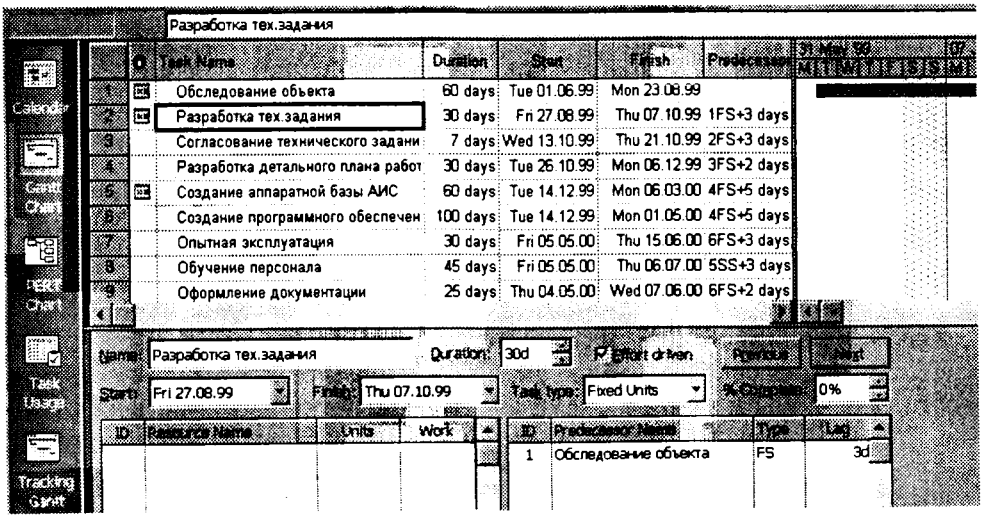


Рис. 9.10. Ввод списка операций

Для «ручного» ввода операций следует:

1. На панели управления представлениями проекта выбрать вид Gant Chart (последнее можно сделать также с помощью меню View ▸ Gant Chart).

2. В колонки таблицы в верхней части окна вводятся атрибуты отдельных операций: Task Name (наименование), Duration (продолжительность). Заметим, что при задании продолжительности можно явно указывать ее размерность. Например, **60 m** (минуты), **1 h** (часы), **10 d** (дни) и т. д. Одновременно предусмотрено задание размерности по умолчанию (меню Tools ▸ Options, вкладка Schedule, поле Duration is entered in:).

На рис. 9.10 показан процесс создания списка операций по рассмотренному в 9.1.4 проекту «ТЕСТ». Хочется обратить внимание на то, что при создании записи по очередной операции ей автоматически присваивается порядковый идентификатор ID[1].

Простейший способ автоматического импорта/экспорта данных может быть продемонстрирован с помощью электронных таблиц Excel. Например, для того чтобы перенести данные из MS Project на лист Excel, требуется:

1. Выделить строки с описанием копируемых задач в MS Project.
2. Вызвать функцию Copy (из меню либо щелчком левой клавиши мыши).
3. Перейти в Excel.
4. Вызвать функцию Вставить (Paste).

Мы рекомендуем заинтересованному читателю проделать данную операцию на практике хотя бы для того, чтобы наглядно ознакомиться с тем, какую структуру приобретает описание списка операций на листе электронных таблиц. Очевидно, что аналогичным способом исходные данные могут быть импортированы в MS Project из Excel.

ПРИМЕЧАНИЕ

Следует заметить, что MS Project обладает свойствами OLE-сервера и его данные могут «погружаться» в другие программы. В Excel и Word это можно сделать с помощью функции Специальная вставка (Paste Special).

Задание связей между операциями

Взаимосвязи между операциями, вообще говоря, могут быть заданы уже в ходе создания их описаний. Для этого по тем операциям, которые зависят от других, заполняется специальная таблица Predecessors (предшественники). В частности, на рис. 9.11 в качестве примера показан процесс задание операций, предшествующих операции «Опытная эксплуатация» рассматриваемого учебного примера.

В MS Project допускаются разные типы связей между операциями (табл. 9.1).

Для любой операции можно определить временной лаг (как положительный, так и отрицательный) относительно предшествующей (управляющей) операции.

[1] Напомним, что в 9.1.2 и 9.1.4 для идентификации операций в проекте мы по соображениям наглядности описания использовали буквенные обозначения, так как цифрами обозначались состояния.

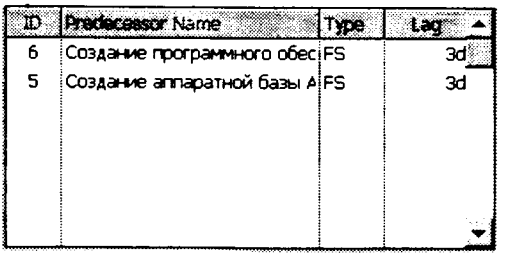


Рис. 9.11. Ввод описания операций-предшественников

Таблица 9.1. Типы зависимостей между операциями

Обозначение	Форма связи
FS	Начало предшествующей операции совпадает с концом зависимой
FF	Моменты завершения предшествующей и зависимой операций совпадают
SS	Моменты начала предшествующей и последующей (зависимой) операций совпадают
SF	Момента начала предшествующей (наверное, в данном случае корректнее будет сказать «влияющей») операции совпадает с моментом завершения зависимой операции

Задание иерархии операций в проекте

В подавляющем большинстве случаев операции в проектах не являются равноправными. Чаще всего можно выделить составные операции, которые распадаются на ряд более простых работ. При этом такая вложенность может иметь несколько уровней глубины. Возможности по заданию иерархической структуры операций, входящих в проект, достаточно удобно реализованы в MS Project.

Для того чтобы сделать операцию подчиненной по отношению к вышестоящей, необходимо выделить запись, соответствующую этой операции, и щелкнуть мышью на пиктограмме Indent, по умолчанию находящейся в панели инструментов Format (Форматирование) (рис. 9.12). Функция Outdent, соответственно, обратно повышает уровень операции в иерархии.

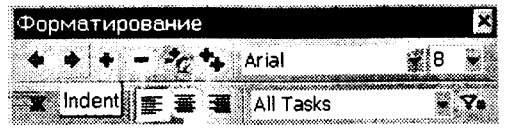


Рис. 9.12. Средства задания иерархии для операций

Также функции управления иерархией операций доступны из меню Project ▸ Outline.

Для каждой составной операции показатели продолжительности, трудоемкости и финансовых затрат определяются автоматически по совокупности входящих в нее подопераций. Помимо этого в плане интерфейсного сервиса предусмотрены возможности по выводу или сокрытию уровней иерархии.

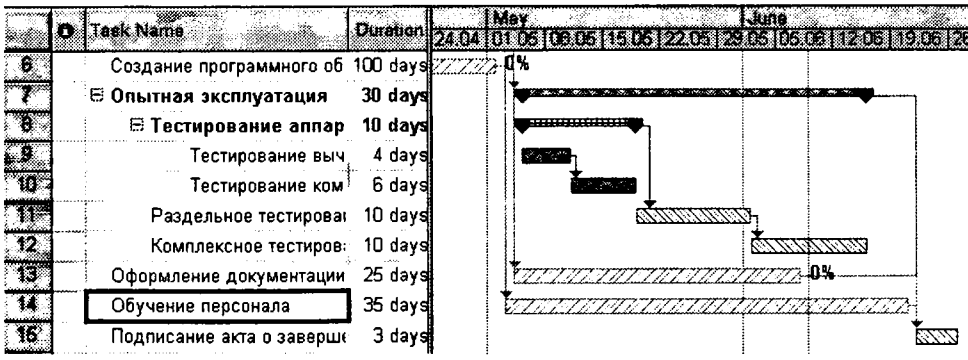


Рис. 9.13. Пример иерархической организации системы операций в проекте

На рис. 9.13 в качестве примера составной операции приведен этап G «Опытная эксплуатация системы» проекта «ТЕСТ». Он разбит на три составные части:

- тестирование аппаратного обеспечения;
- раздельное тестирование подсистем программного обеспечения;
- комплексное тестирование автоматизированной системы.

В свою очередь, первая подоперация делится на части:

- тестирование вычислительной техники;
- тестирование коммуникационного оборудования.

Задание ограничений по времени

Помимо задания логических связей между операциями, отдельные «ключевые» операции можно привязать к конкретным моментам времени.

Для того чтобы осуществить привязку текущей операции, необходимо:

1. Выбрать пункт меню Project ▸ Task Information (либо просто щелкнуть мышью по нужной записи в перечне операций).
2. В появившемся окне диалога выбрать вкладку Advanced.
3. Изменить значение полей, входящих в группу Constrain Task: Type (тип ограничения) и Date (момент времени, к которому производится привязка) (рис. 9.14).

В MS Project допускается задание различных типов привязки. Например, «Start No Later Than» (начало не позднее, чем...) или «Finish No Earlier Than» (завершение не ранее, чем...) и т. п.

Для операций, по которым заданы соответствия конкретным моментам времени, в таблице с перечнем в специальной колонке выводится специальная пиктограмма.

Отметим, что данной возможностью следует пользоваться крайне осторожно, так как наличие конкретных временных ограничений блокирует многие функции по автоматическому перевычислению графика проекта, поэтому по умолчанию для операций задается тип привязки «As Soon As Possible» (начинать как можно раньше).

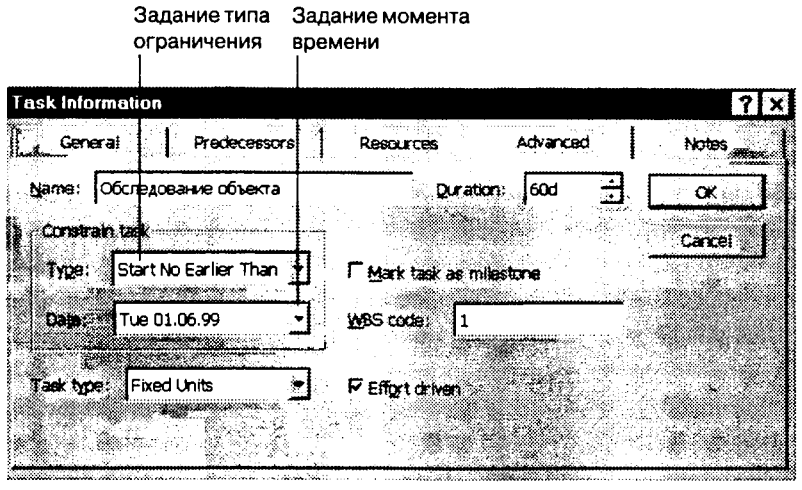


Рис. 9.14. Определение параметров временных ограничений

Формирование ресурсного обеспечения проекта

Непременным элементом процесса управления проектами является планирование и управление ресурсами, которые необходимы для выполнения предусмотренных работ. Например, в контексте учебного примера «ТЕСТ» в качестве таковых могут рассматриваться вычислительная и множительная техника, группы разработчиков прикладного ПО, системных программистов и т. д.

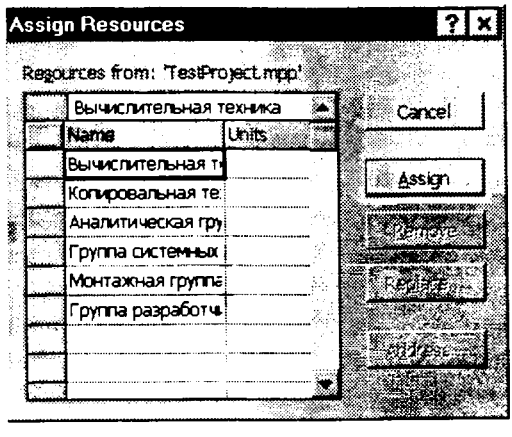


Рис. 9.15. Формирование списка ресурсов

Для формирования списка доступных ресурсов проекта следует вызвать функцию меню Tools ▸ Resources ▸ Assign Resources (либо нажать комбинацию клавиш Alt+F10), после чего появляется диалоговое окно с именем Assign Resources (рис. 9.15), позволяющее формировать список ресурсов и присваивать те или иные ресурсы текущей операции.

Присвоение некоторого количества того или иного ресурса текущей операции возможно либо непосредственно из окна Assign Resources (для этого в нем предусмотрена специальная кнопка Assign), либо из вкладки Resources окна Task Information (напомним, что для его вызова достаточно щелкнуть мышью на записи, относящейся к нужной операции) (рис. 9.16).

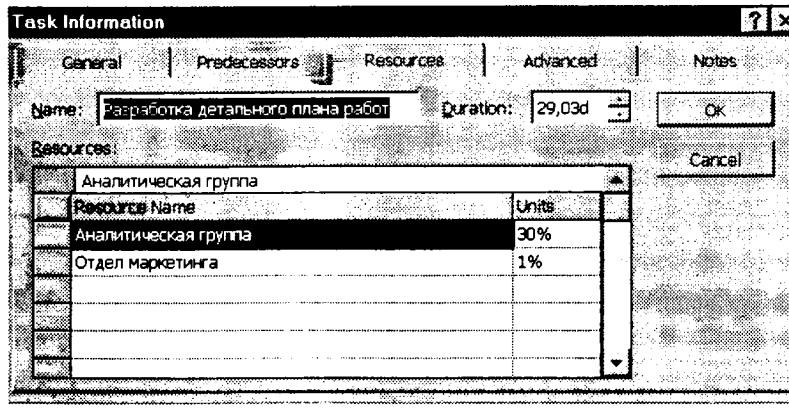


Рис. 9.16. Назначение ресурсов для операции

При обсуждении вопросов формирования информационной базы по ресурсному обеспечению проекта необходимо особо остановиться на том содержании, которое вкладывается в понятие «ресурс». На самом общем уровне ресурсы могут быть классифицированы на *возобновляемые* и *невозобновляемые*. К возобновляемым относятся такие ресурсы, как рабочее время работников или некоторые постоянно применяемые технические средства (основные фонды). Примером невозобновляемых ресурсов могут служить финансы, материалы, энергия и т. п.

В MS Project предусмотрена возможность работы только с одним видом невозобновляемых ресурсов — финансами, или, другими словами, предполагается, что любые виды невозобновляемых ресурсов будут сведены к денежному эквиваленту[1]. Вообще говоря, в профессиональной литературе по данной программной системе предлагаются определенные пути преодоления этого несомненного неудобства, однако они носят весьма частный и искусственный характер.

В то же время арсенал средств по управлению возобновляемыми ресурсами, доступный пользователям MS Project, достаточно широк, поэтому в дальнейшем мы сконцентрируем наше внимание на нем и под термином «ресурсы» по умолчанию будем понимать именно возобновляемые ресурсы. Очевидно, что их количественные характеристики однозначно связаны с временными параметрами тех операций, которые они обеспечивают. Для описания этой зависимости в MS Project используются понятия: *продолжительность операции* (Duration), *трудоемкость* (Work)[2], *количество единиц ресурса* (Units). Их связывает соотношение Work = Duration×Units.

[1] Вопросы управления финансами будут рассмотрены далее.

[2] Возможно, более точным был бы термин «ресурсоемкость».

Размерность переменной Work определяется размерностью переменной Duration, то есть может измеряться в минутах, часах, днях и неделях. Количество единиц ресурса может измеряться в долях единицы или процентах. При этом значение потребности в ресурсах (Units) для любой операции может быть задано в пределах от 0 до 10 000 (1 000 000 %), (рис. 9.16). Еще раз подчеркнем, что такой подход к измерению ресурса обусловлен его возобновляемой природой, при которой его физический объем может считаться постоянным, и принципиальным является то, какая его часть задействуется на обеспечение той или иной операции.

Правила формирования значений продолжительности, трудоемкости и количества ресурсов для операции зависят от ее типа. Существуют следующие типы:

- операции с фиксированной продолжительностью;
- операции с фиксированным количеством единиц ресурса;
- операции с фиксированной трудоемкостью.

Схематично правила преобразования параметров операции могут быть описаны с помощью табл. 9.2.

Таблица 9.2. Преобразование параметров операции в зависимости от ее типа

Измененный показатель	**Влияние на исходные значения показателей**		
	Продолжительность, *D*	**Трудоемкость, *W***	**Число единиц ресурса, *U***
	Операции с фиксированной продолжительностью		
Продолжительность, $D_н$	$D_н$	$W = D_н U$	U = const
Трудоемкость, $W_н$	D = const	$W_н$	$U = W_н / D$
Число единиц ресурса, $U_н$	D = const	$W = DU_н$	$U_н$
	Операции с фиксированной продолжительностью		
Продолжительность, $D_н$	$D_н$	$W = D_н U$	U = const
Трудоемкость, $W_н$	$D = W_н / U$	$W_н$	U = const
Число единиц ресурса, $U_н$	D = const	$W = DU_н$	$U_н$
	Операции с фиксированной продолжительностью		
Продолжительность, $D_н$	$D_н$	W = const	$U = W / D_н$
Трудоемкость, $W_н$	D = const	$W_н$	$U = W_н / D$
Число единиц ресурса, $U_н$	$D = W / U_н$	W = const	$U_н$

Правила интерпретации содержимого табл. 9.2 продемонстрируем на примере операций с фиксированной продолжительностью:

- если пользователь меняет величину продолжительности с D на $D_н$, то число единиц ресурса U остается неизменным, а трудоемкость пересчитывается по формуле $W = D_н U$;
- если меняется трудоемкость с W на $W_н$, то продолжительность D остается неизменной, а число единиц ресурса пересчитывается по формуле $U = W_н / D$;
- наконец, если меняется число единиц ресурса с U на $U_н$, то продолжительность D остается неизменной, а трудоемкость пересчитывается по формуле $W = DU_н$.

Правила преобразования параметров операций других типов, описанных в табл. 9.2, интерпретируются аналогично.

9.2.2. Формирование и представление графика работ

Простейшим (хотя порой весьма эффективным) инструментом планирования и управления проектом является визуальный анализ его графика. С этой точки зрения MS Project предоставляет менеджерам достаточно богатый арсенал средств.

Технологически данные задачи решаются на уровне задания видов (Views) проекта. Выбирать способы представления проекта можно либо с помощью специальной панели управления, расположенной в левой части главного окна программы (рис. 9.8), либо с помощью меню View.

Календарный график

Наиболее детальным способом визуального представления проекта является *календарный график* (Calendar) (рис. 9.17).

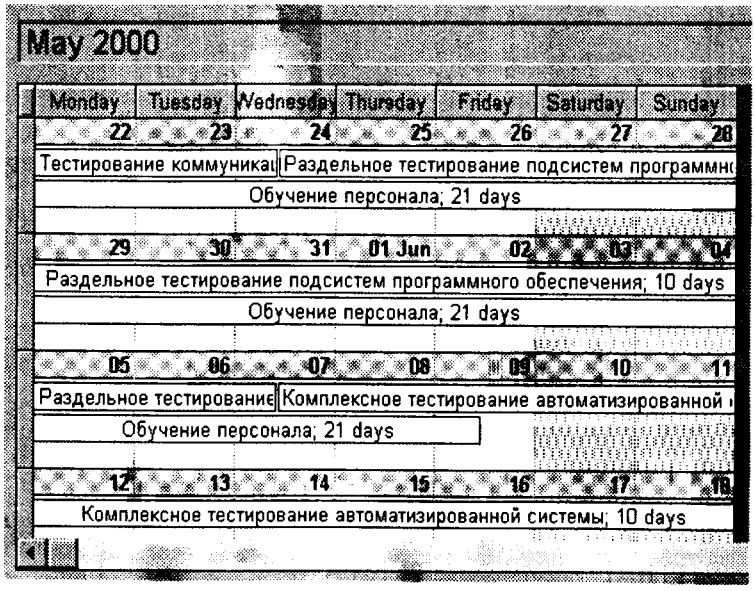


Рис. 9.17. Календарный график проекта

В данном режиме подробную информацию об операциях, выполняющихся в заданный день, можно получить, щелкнув мышью на его номере, в результате чего появляется окно, показанное на рис. 9.18.

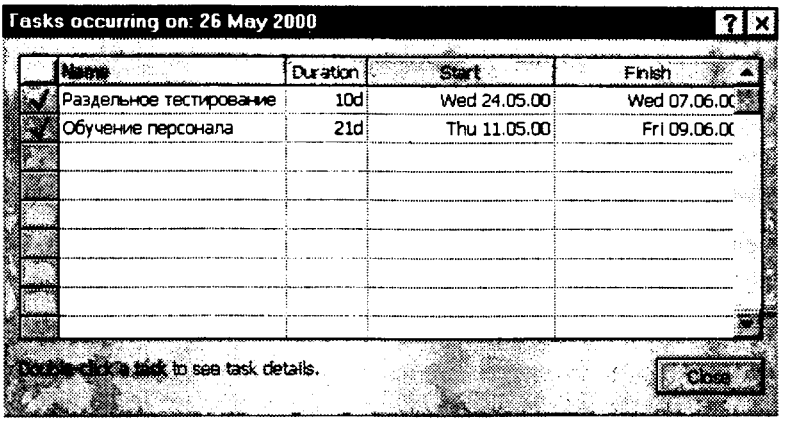


Рис. 9.18. Сведения по операциям (на заданную дату)

Обобщая свойства режима календарного графика, можно заметить, что при таком способе визуализации удобно получать ответ на вопрос «Какие операции должны выполняться на заданную дату?» и практически невозможно получить представление о продвижении проекта в целом.

Диаграмма Гантта

Другим способом визуального представления проектов являются *диаграммы Гантта*. Они, как уже отмечалось ранее, представляют собой исторически один из первых и весьма эффективный метод оперативно-календарного планирования, активно применявшийся самостоятельно еще до появления автоматизированных вычислительных систем. Собственно говоря, нами ранее по умолчанию уже использовался режим вывода информации по проекту в виде графиков Гантта. Это делалось, допустим, тогда, когда описывались методы задания иерархических связей между операциями. Вариант представления проекта в виде графика Гантта показан на рис. 9.19.

Весьма полезным оказывается предусмотренный в MS Project обширный набор возможностей по гибкому конфигурированию вида ленточных диаграмм. Наиболее используемые функции доступны из выпадающего меню, активизируемого при щелчке мышью по прямоугольным полосам графика. Например, если выбрать функцию Format, то появится окно диалога (рис. 9.20), позволяющее изменить информацию, выводимую по одной конкретно выбранной операции.

Так, на рис. 9.20 показан способ задания легенд, которые сопровождают прямоугольник, отображающий длительность операции (Bar). В данном случае справа от него будет выводиться информация о проценте выполнения работ по операции (% Complete).

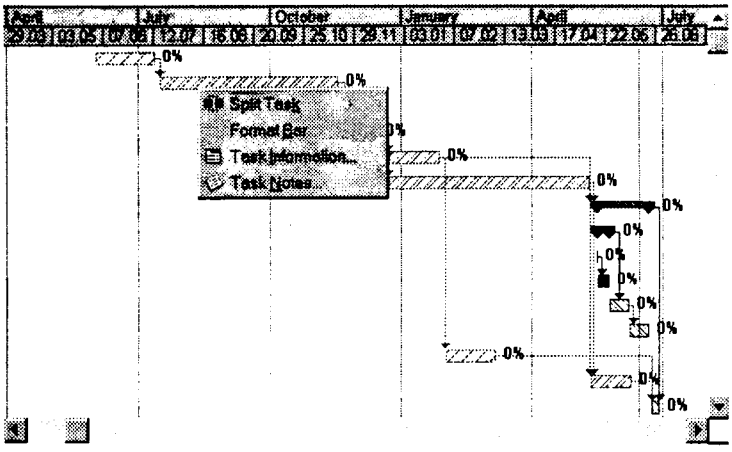


Рис. 9.19. Визуальное представление проекта с помощью диаграмм Гантта

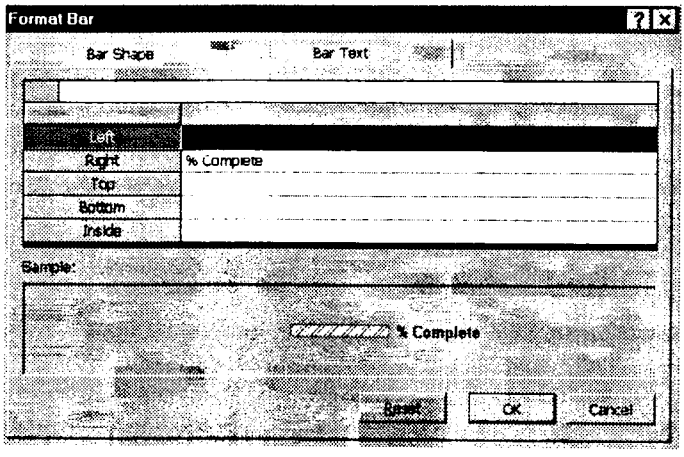


Рис. 9.20. Конфигурация вида операций в диаграмме

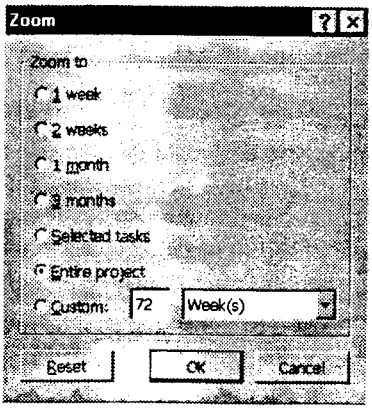


Рис. 9.21. Меню выбора масштаба графика проекта

Удобным и часто используемым средством работы с визуальными представлениями проектов является функция Zoom, доступная из контекстного меню, активизируемого при щелчке мышью по области дат, либо из меню View ▸ Zoom. Она дает возможность выбрать масштаб, в котором будет выводиться график проекта (рис. 9.21). В частности, на рис. 9.21 показан выбор режима масштабирования Entire Project, который позволяет вывести в максимально крупном виде весь проект.

Сетевая диаграмма

Название следующего способа представления проекта — PERT Chart[1]. В русском переводе это термин обычно называют *сетевой диаграммой*. Данный способ представляет совокупность операций проекта в виде логической схемы типа вершинного графа (см. 9.1.2), то есть операции изображаются с помощью прямоугольников, а связи между ними — с помощью стрелок (рис. 9.22).

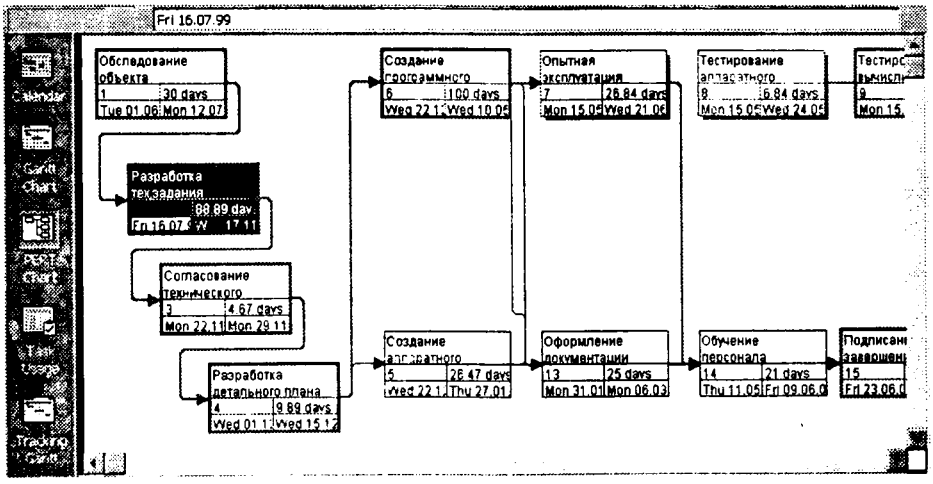


Рис. 9.22. Визуальное представление проекта с помощью сетевой диаграммы

Очевидно, что при решении задач, требующих оценки временных характеристик проекта, сетевая диаграмма оказывается малоинформативной, однако при структурно-логическом анализе она может быть весьма полезной.

Другие способы представления проекта

Если обратить внимание на панель управления видами проекта[2], то на ней помимо перечисленных можно обнаружить еще несколько других способов представления, а именно:

- Task Usage — таблица данных по операциям с детальным распределением ресурсов;
- Tracking Gantt — разновидность стандартной диаграммы Гантта (Gantt Chart), формат которой специально ориентирован на отображение процесса выполнения проекта;

[1] Нетрудно догадаться, что англоязычная версия названия данного режима визуализации непосредственно произошла от уже упоминавшегося метода сетевого планирования.

[2] Напомним, что нужный вид также может быть выбран с помощью меню View.

- Resource Graph — диаграмма загрузки ресурсов;
- Resource Sheet — сводная таблица загрузки ресурсов;
- Resource Usage — таблица загрузки ресурсов (с детализацией по операциям).

Наконец, последняя пиктограмма на панели управления видами называется More Views (Другие виды). При выборе ее происходит переход в режим, позволяющий пользователю создавать собственные варианты представлений проекта (рис. 9.23).

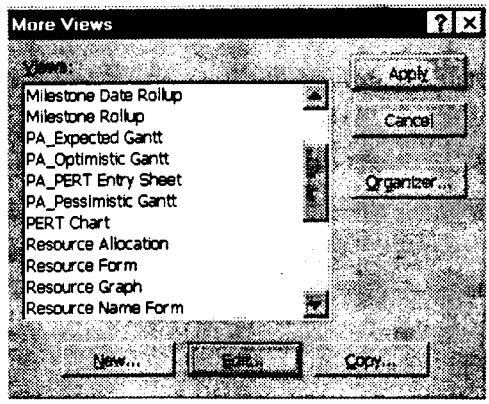


Рис. 9.23. Меню режима управления представлениями (видами)

Как видно из рис. 9.23, представления проекта можно создавать (кнопка New) и копировать (кнопка Copy), а также редактировать их отдельные параметры (кнопка Edit).

Формы и таблицы, визуализирующие проект, заполняются на основе содержания единой базы данных MS Project. Это гарантирует, что любое изменение, произведенное в одном режиме (виде), автоматически отображается во всех других видах.

Следует иметь в виду, что различные формы визуализации проекта являются взаимодополняющими инструментами и ни одна из них сама по себе не может дать полной информации обо всех его характеристиках, состояниях и параметрах. Поэтому важнейшим критерием уровня мастерства и квалификации специалиста по сетевому планированию является его умение создавать функционально достаточный и одновременно неизбыточный набор представлений. MS Project позволяет их комбинировать, разбивая экран по горизонтали на верхнюю и нижнюю части. При этом верхняя часть экрана содержит *основную форму*, а нижняя — *дополнительную*, которая уточняет информацию основной. Пример такого подхода к решению задач визуализации показан на рис. 9.24, где в верхней (основной) части задано представление типа Gantt Chart (диаграмма Гантта), а в нижней (дополнительной) — Task Usage (таблица загрузки ресурсов).

Фильтры

Важнейшей сервисной характеристикой любого программного обеспечения является наличие в нем средств по оперативному отбору той информации (из всей ее совокупности), которая необходима пользователю в некоторый фиксированный

момент времени. В рассмотренных выше программах (Word, Excel, Access) стандартным и удобным способом решения этой задачи являлись *фильтры*. Аналогичный механизм реализован и в MS Project.

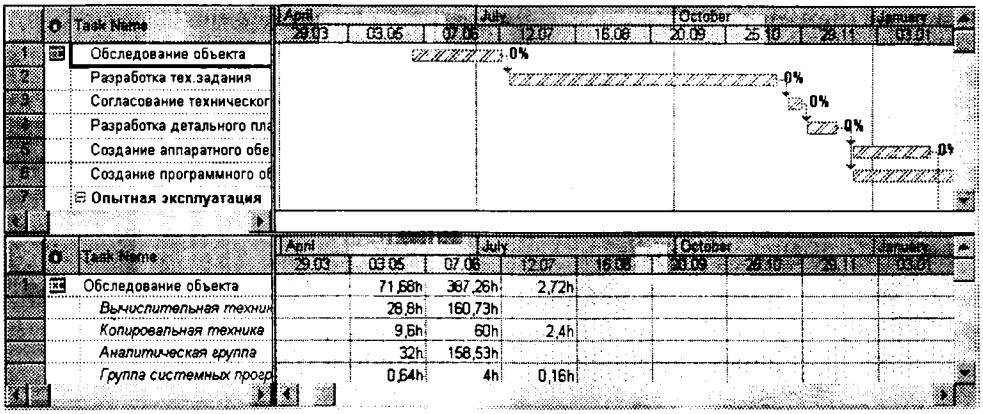


Рис. 9.24. Комбинированное представление проекта

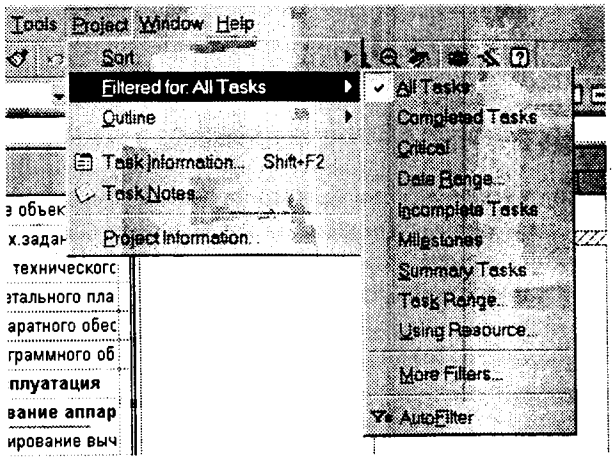


Рис. 9.25. Задание фильтра

С помощью **фильтров** (они доступны из панели инструментов либо из меню Project ▸ Filtered For ...) задаются логические условия вывода или скрытия данных в формы и таблицы (с учетом их содержания) (рис. 9.25). Пользователь может либо использовать заранее определенные фильтры, сохраняемые в системе настроек, либо создавать новые по своему усмотрению.

Полезный инструмент работы с данными проекта представляет собой функция Autofilter (Автофильтр), ее легко вызвать щелчком мышью на пиктограмме.

Принцип работы режима автофильтра такой же, как и у аналогичных режимов в Excel и Access, то есть при его активизации к заголовкам всех колонок добавляется

управляющая кнопка, по нажатии которой открывается раскрывающийся список всех возможных значений, содержащихся в текущей колонке (рис. 9.26). Выбор любого значения из списка приводит к тому, что в действующем представлении будут выводиться только те данные, для которых текущее поле имеет выбранное значение. Фильтрующие условия, накладываемые на различные колонки, могут дополнять друг друга по правилу логического сложения («И»).

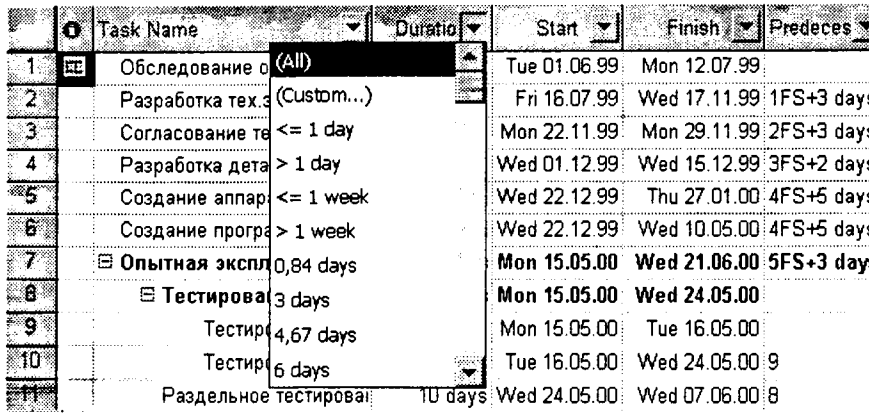


Рис. 9.26. Задание автофильтра

9.2.3. Дополнительные возможности ПО MS Project

Назначение данного пункта — кратко охарактеризовать те средства и возможности MS Project, о которых в силу ограниченности размеров учебного пособия мы не сможем рассказать подробно, но которые, по мнению авторов, имеют принципиальное значение при конкретной работе по планированию и управлению проектами.

Печать отчетов

Отчеты (Reports) представляют собой инструмент вывода на печать (жесткий носитель) информации, хранимой в базе данных проекта и отформатированной заданным образом. Управление отчетами осуществляется через меню View ▸ Reports... Из него последовательно вызываются окна выбора группы отчетов и выбора (из группы) отчета конкретного типа. Они показаны на рис. 9.27 и 9.28.

В MS Project отчеты делятся на группы:

- Overview — общие сведения о проекте;
- Current Activities — выполняемые работы;
- Costs — затраты;
- Assignments — назначение ресурсов;
- Workload — загрузка ресурсов;
- Custom — созданные пользователем.

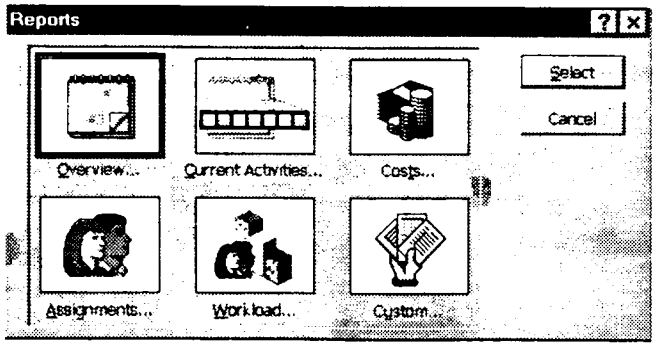


Рис. 9.27. Выбор группы отчетов

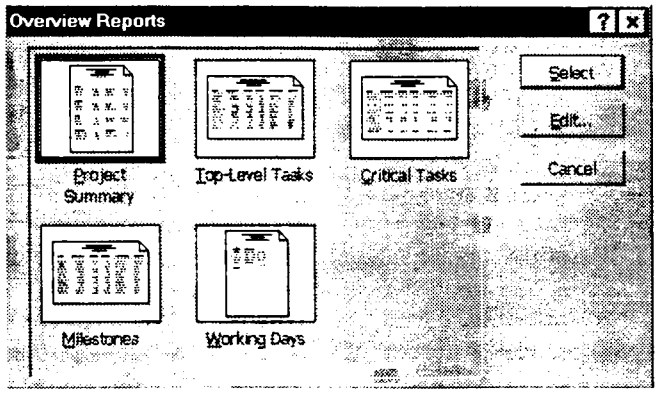


Рис. 9.28. Выбор типа отчета (из группы Overview Reports)

Пользователь может либо воспользоваться встроенными стандартными отчетами, либо разработать собственные типы отчетов. Основой для их формирования являются формы представления данных о графике проекта, таблицы, фильтры. Окно режима определения нового типа отчета (Task Report) показано на рис. 9.29. Оно вызывается кнопкой Edit из окна Custom Reports.

В отчеты могут быть включены только данные из табличной части проекта, однако с помощью команды File ▸ Print могут быть распечатаны любые представления проекта.

В заключение следует отметить, что в плане сервиса и интерфейса процесс печати в MS Project организован так же, как и в других программных продуктах семейства MS Office.

Создание программных расширений

В MS Project, как и в других программных продуктах, входящих в семейство MS Office, встроена гибкая система создания дополнительных программных надстроек, которые расширяют его функциональные возможности.

Данные функции доступны из меню Tools ▸ Macro. Макропрограммы создаются в среде встроенного языка программирования Visual Basic for Applications (VBA), являющейся универсальной для всего семейства MS Office.

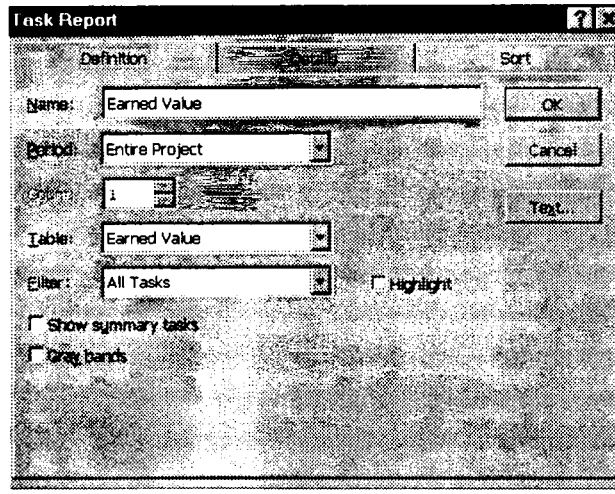


Рис. 9.29. Окно определения нового типа отчета

Используя редактор и отладчик VBA, пользователь, обладающий определенным опытом программирования, может создавать собственные процедуры и функции, автоматизирующие какие-либо часто повторяющиеся действия по обработке содержания проекта. При этом для обращения из вновь создаваемых процедур к данным проекта и работы с соответствующими функциями (методами) их обработки используется специальный объектно-ориентированный программный интерфейс. Общая структура объектной модели MS Project показана на рис. 9.30.

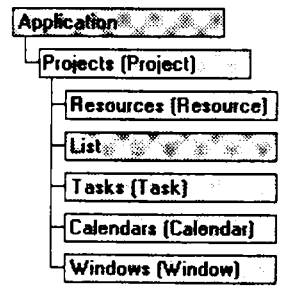


Рис. 9.30. Общая структура объектной модели MS Project

В завершении разговора о программировании в среде MS Project приведем в качестве примера листинг простейшей программы, которая выводит список всех открытых на текущий момент проектов.

```
'-------------------------------------------------------
' Данная процедура, выводит список всех открытых на
' текущий момент проектов
Sub OutOpenProjectNames()
' описание переменных
Dim i As Long, strNames As String
```

```
'  цикл по всем открытым проектам
'  свойство .Count содержит количество членов
'  в семействе Projects
       For i = 1 To Projects.Count
'      свойство.FullName содержит полное наименование
'      проекта, которое добавляется к строке strNames
           strNames = strNames & Projects(i).FullName & vbCrLf
       Next i
'  вывод строки со списком имен проектов
       MsgBox strNames
   End Sub
'-------------------------------------------------------
```

Работа с шаблонами и настройками

Анализируя многие из описанных ранее алгоритмов работы с системой MS Project, читатель может прийти к вполне обоснованному выводу о том, что существенная доля трудозатрат пользователя приходится на создание различного рода настроек и конфигураций способов представления данных в проектах, фильтров и т. п.

В большинстве случаев затраты времени пользователя могут быть уменьшены за счет грамотного применения им системы шаблонов и настроек. По умолчанию настройки разработанных и измененных пользователем форм, таблиц, представлений (видов), отчетов, макросов и других объектов запоминаются в специальном файле шаблонов GLOBAL.MPT. Фактически, он играет ту же роль, что и файл NORMAL.DOT в MS Word. Поскольку GLOBAL.MPT открывается всякий раз при запуске MS Project, то однажды созданные и запомненные в нем настройки становятся доступными для любых новых проектов, с которыми в дальнейшем будет вестись работа на данном компьютере. Одновременно пользователь при желании может сохранять настройки и в файле проекта. Это имеет смысл делать, если такие настройки носят уникальный характер или должны быть перенесены вместе с проектом на другой компьютер, где используется свой GLOBAL.MPT.

Для манипуляций с данными по шаблонам и настройкам, переноса их из проекта в GLOBAL.MPT и обратно в MS Project предусмотрен режим организатора, который может быть вызван из меню Tools ▸ Organizer. Вид окна режима организатора показан на рис. 9.31.

Поддержка технологий совместной работы

В MS Project реализованы возможности по организации совместной работы с общим проектом (или проектами) для группы пользователей (Workgroup), связанных в рамках некоторой вычислительной сети. В данном режиме реализованы следующие функции:

- пересылка плановых заданий на рабочие станции участников проекта;
- получение от членов рабочей группы проекта информации о его ходе и результатах;

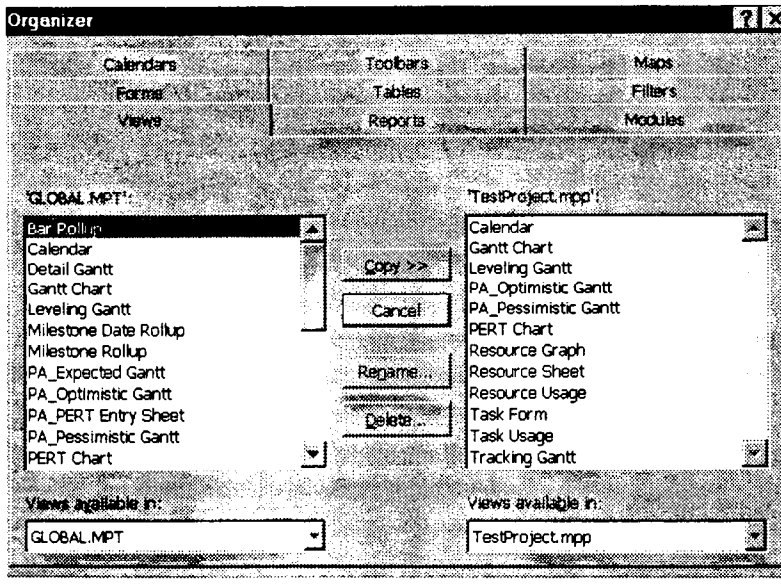


Рис. 9.31. Окно режима организатора

- автоматический пересчет графика проекта на основе данных, полученных от других пользователей;
- поддержка системы совместно используемых ресурсов.

9.3. Управление циклом реализации и ресурсами проекта

Вопросы, рассмотренные в предыдущем разделе, были, в основном, посвящены технологическим (вспомогательным) аспектам работы MS Project. Теперь мы переходим к проблематике, непосредственно связанной с реализацией в рамках данной программной системы экономических задач.

9.3.1. Критический путь и методы управления им

Одной из главнейших проблем управления проектами является управление *циклом реализации*. Как правило, задачи такого типа формулируются как *задачи минимизации* (или *ограничения в заданных пределах*) *времени выполнения проекта*.

В случае *последовательной структуры* проекта (то есть когда проект распадается на операции, выполнение которых не может быть совмещено) задача управления периодом его реализации с точки зрения теории сетевого планирования становится тривиальной. Однако если какие-то работы по проекту можно вести параллельно (то есть он имеет *параллельную* или *последовательно-параллельную структуру*, например, такую, как на рис. 9.32), то ответ на вопрос, как составить кратчайший по продолжительности график, значительно усложняется.

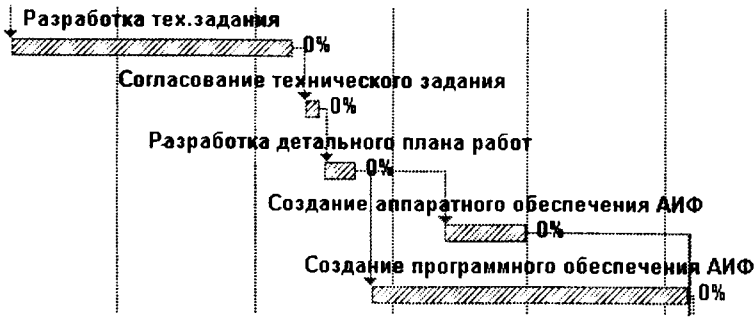


Рис. 9.32. Пример последовательно-параллельной структуры проекта

Существенную роль в методах решения данной проблемы играют понятия критической операции и критического пути. Для начала поясним их смысл на примере. Как нетрудно заметить из рис. 9.32, операция **«Создание аппаратного обеспечения»**, с одной стороны, может вестись одновременно с операцией **«Создание программного обеспечения»**, а с другой — имеет меньшую продолжительность, что позволяет изменять момент начала данной операции, не влияя на сроки начала последующих операций и момент завершения проекта в целом.

Момент графика проекта, начиная с которого, операция может быть поставлена на выполнение, называют *ранним стартом*, а момент, позже которого операция не может быть завершена, чтобы не сдвинулось начало зависимых от нее операций, — *поздним финишем*. Если из значения позднего финиша вычесть значения раннего старта и длительность самой операции, то получится *свободный резерв времени*. Операции, для которых свободный резерв времени равен нулю, называют *критическими*. Множество всех критических операций проекта образует его *критический путь*. Другими словами, критический путь представляет собой совокупность таких операций, увеличение длительности которых ведет к увеличению длительности проекта в целом.

Очень часто та степень точности, с которой мы можем оценить продолжительность той или иной операции, оказывается весьма приблизительной. MS Project позволяет учитывать эти обстоятельства, относя к критическому пути те операции, величина резерва времени которых отличается от нуля, но не превышает некоторого предела. Очевидно, что в этом случае появляется возможность существования в проекте сразу нескольких «параллельных» критических путей. Для установки данного режима необходимо выполнить следующую последовательность действий (рис. 9.33).

1. Вызвать команду меню Tools ▸ Options.
2. Выбрать вкладку Calculation.
3. В поле Tasks are critical if slack is less than or equal to ввести значение, определяющее предел, после которого в случае снижения значения свободного резерва времени операция будет включаться в критический путь.
4. Установить переключатель Calculate multiple critical paths, что позволит контролировать одновременно несколько критических путей.
5. Нажать кнопку OK.

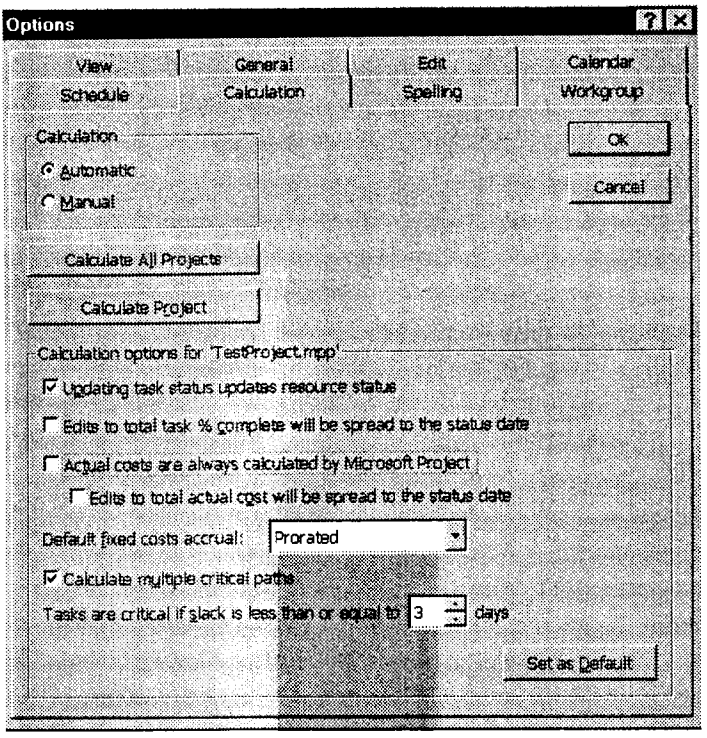


Рис. 9.33. Изменение параметров вычисления критического пути

Среди основных средств управления критическим путем проекта, доступных в MS Project, могут быть названы:

- визуальное выделение операций, относящихся к критическому пути, при помощи форматирования;
- применение фильтров для обеспечения возможности сосредоточить внимание исключительно на операциях критического пути;
- определение параметров критического пути (продолжительности, моментов начала и завершения операций).

Остановимся на этих методах работы более подробно.

Визуальное выделение операций критического пути

Для визуального выделения операций, входящих в критический путь, необходимо выполнить команду меню Format ▸ GanttChartWizard. Она вызывает специальную программную надстройку — мастер форматирования, которая, вообще говоря, в режиме диалога позволяет пользователю в соответствии с его потребностями гибко менять представление проекта в виде диаграмм Гантта. В данном случае, находясь на этапе 2 (step 2), следует установить переключатель Critical path (рис. 9.34).

Далее, следуя подсказкам мастера (последовательно вводя запрашиваемые параметры и нажимая кнопку Next, а в завершении — кнопку Format It), пользователь получает такой вид проекта, в котором отчетливо видны все операции, входящие в критический путь. По умолчанию критические операции выделяются красным цветом, а некритические — синим.

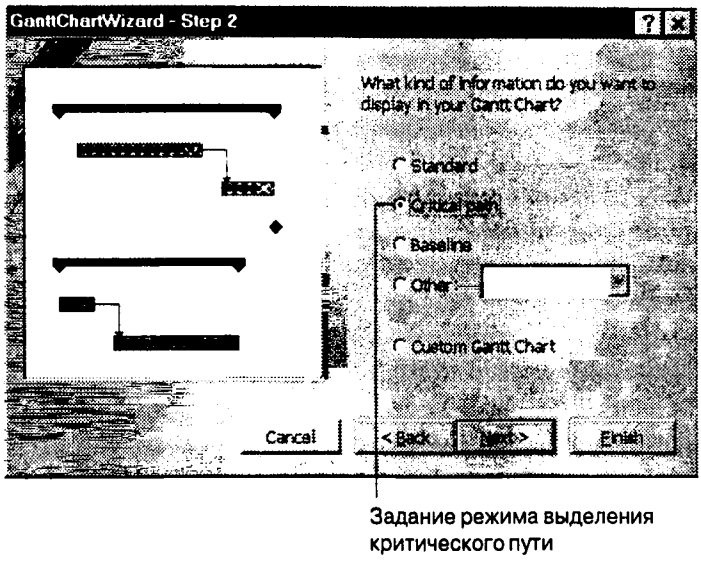


Рис. 9.34. Задание выделения критического пути

Применение фильтров для выделения операций критического пути

Для того чтобы оставить на диаграмме только критические операции (убрать некритические), следует в окне задания фильтров на панели инструментов (рис. 9.35) выбрать встроенный фильтр Critical.

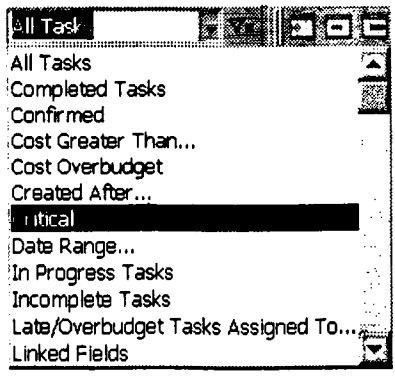


Рис. 9.35. Задание фильтра для выделения критических операций

ПРИМЕЧАНИЕ

При работе с описанным режимом визуализации необходимо учитывать следующую особенность — несмотря на то, что после внесения изменений в исходные данные проекта все его показатели автоматически пересчитываются, перевывод содержимого критического пути не производится, и поэтому процедуру задания фильтра Critical необходимо повторить.

Определение параметров критического пути

Точные значения начала и конца критического пути можно получить, выполнив команду меню Project ▸ Project Information. По ней вызывается окно Project Information, в котором поле Start Date содержит дату начала работ, а поле Finish Date — дату их окончания.

Для наглядного определения продолжительности операций критического пути в график проекта удобно ввести сводную операцию, объединяющую все его составные части. Это можно сделать с помощью команды Tools ▸ Options. В появившемся окне следует выбрать вкладку View (рис. 9.36). Если в группе Outline options установить переключатель Project summary task, то в перечень работ графика будет включена составная операция. Ее продолжительность (Duration) будет равна продолжительности критического пути проекта.

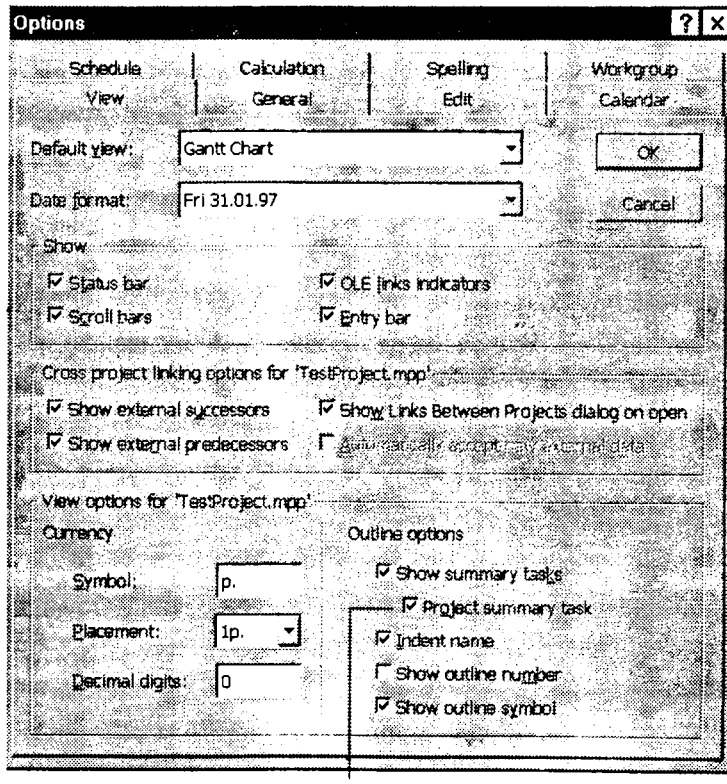


Рис. 9.36. Задание режима вывода в «сводной» операции

Перечисленные методы работы с критическим путем позволяют решать стандартные задачи по его сокращению. Как правило, оно может быть достигнуто за счет:

- сокращения длительности критических операций;
- дополнительного «распараллеливания» операций;
- изменения структуры проекта и его отдельных операций.

Очевидно, что в каждом конкретном случае выбор того или иного пути осуществляется, исходя из специфики и условий выполнения проекта.

9.3.2. Базовый план и контроль его выполнения

Как уже упоминалось ранее, завершающим этапом сетевого планирования является оперативное управление проектом. В его основе лежат: 1) сопоставление действительных результатов процесса выполнения проекта с изначально намеченным планом; 2) анализ имеющихся расхождений; 3) выработка на его основе корректирующих управляющих воздействий.

Технологически в MS Project задачи оперативного управления реализуются через функции и режимы, позволяющие:

- вводить и сохранять данные о базовом плане проекта;
- вносить в график проекта данные о его ходе и результатах;
- выявлять отклонения результатов выполнения от базового плана.

Базовый и промежуточный планы

Под *базовым планом* (Baseline) в MS Project понимают эталонный график следования и продолжительности операций, информация о котором сохраняется параллельно с данными о вносимых текущих изменениях.

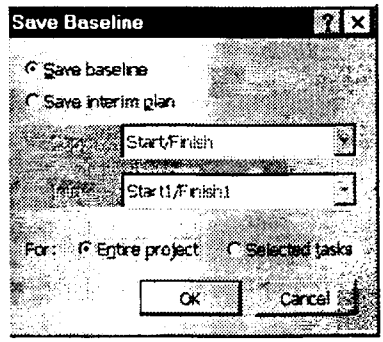


Рис. 9.37. Окно сохранения базового или промежуточного плана

Для того чтобы сохранить базовый план, нужно:

1. Выполнить команду меню Tools ▸ Tracking ▸ Save Baseline.
2. В появившемся диалоговом окне, показанном на рис. 9.37, выбрать опцию Save Baseline и нажать кнопку OK.

Описанные действия можно выполнить в любой момент работы с проектом. При этом каждый раз ранее сохраненные данные по базовому плану будут заменены текущими данными.

ВНИМАНИЕ

MS Project может запомнить только одну версию базового плана!

ПРИМЕЧАНИЕ

При первом (после создания) сохранении проекта пользователю выдается запрос о том, следует ли запоминать базовый план.

Как видно из рис. 9.37, при сохранении базового плана можно определить, будет ли запомнена (изменена) информация по проекту в целом или только по отдельным операциям.

Также обратим внимание читателя и на то, что в базе данных MS Project помимо базового плана может сохраняться и так называемый *промежуточный план* (Interim plan), см. соответствующую опцию (Save interim plan) в диалоговом окне на рис. 9.37.

Сохранение данных о ходе работ

Без преувеличения можно утверждать, что необходимым условием успеха любой управленческой деятельности является наличие точной и продуманной системы оперативной отчетности. Принципы формирования такой системы составляют предмет специальной экономической дисциплины — *управленческого учета*, и вряд ли будет целесообразным обсуждать их на страницах данной книги.

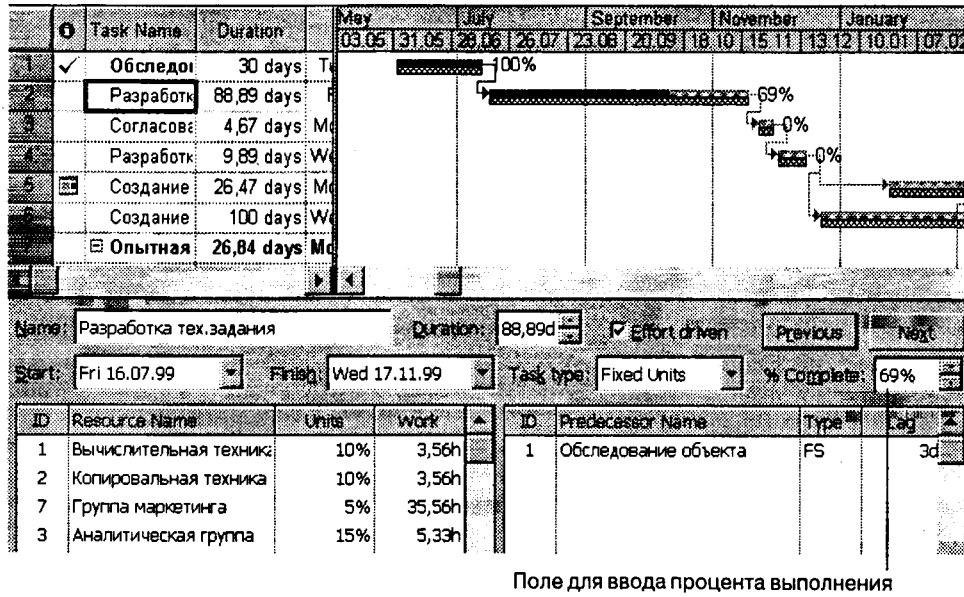


Рис. 9.38. Ввод информации о степени готовности операции

В данном случае для нас представляют интерес технологические возможности ПО MS Project по решению учетных задач. Вообще говоря, их арсенал достаточно

богат. Ограничимся описанием простейшего алгоритма внесения отчетной информации в график проекта:

1. Выбрать в качестве способа визуального представления проекта Tracking Gantt, который специально ориентирован на отображение процесса выполнения (рис. 9.38). Как видно из рис. 9.38, при выбранном способе представления каждой операции соответствуют две полосы: нижняя отображает временные характеристики операции в соответствии с базовым планом, а верхняя — фактические. При этом закраска верхней полосы отображает процент выполнения работ по соответствующей операции (его значение находится справа).
2. Выбрать в таблице, находящейся в левой части окна, нужную операцию.
3. В поле % Complete ввести значение доли выполненных работ по операции в процентах.

ПРИМЕЧАНИЕ

Интерфейс MS Project предусматривает возможность изменения значения степени готовности непосредственно на диаграмме с помощью мыши.

Дополнительно отметим и то, что форма представления Tracking Gantt является эффективным инструментом для выявления отклонений текущего состояния проекта от его базового плана.

9.3.3. Управление ресурсами проекта

Естественным и эффективным инструментом по управлению ресурсами проекта является коллекция представлений (видов) проекта, которые специально ориентированы на решение задач данного типа. В данном разделе мы более подробно остановимся на их возможностях и особенностях применения.

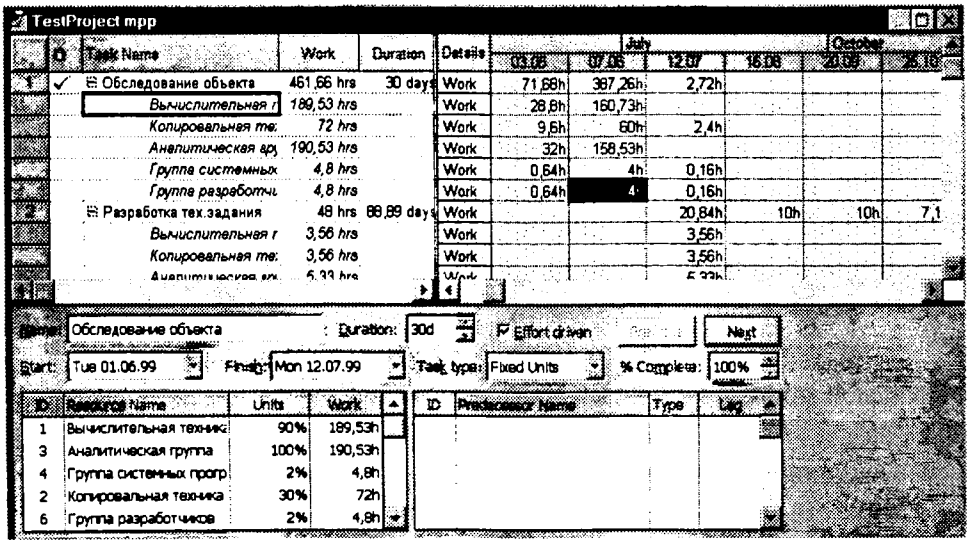


Рис. 9.39. Task Usage — таблица данных по операциям (с разбивкой ресурсов)

Представление проекта Task Usage — таблица данных по операциям с детальным распределением ресурсов — позволяет получить подробную информацию о том, какой вклад в трудоемкость каждой операции вносит каждый ресурс как в течение всего проекта, так и в течение отдельных периодов его реализации. При этом уровень интеграции этих периодов может определяться самим пользователем с помощью функции Zoom контекстного меню (см. комментарий к рис. 9.21).

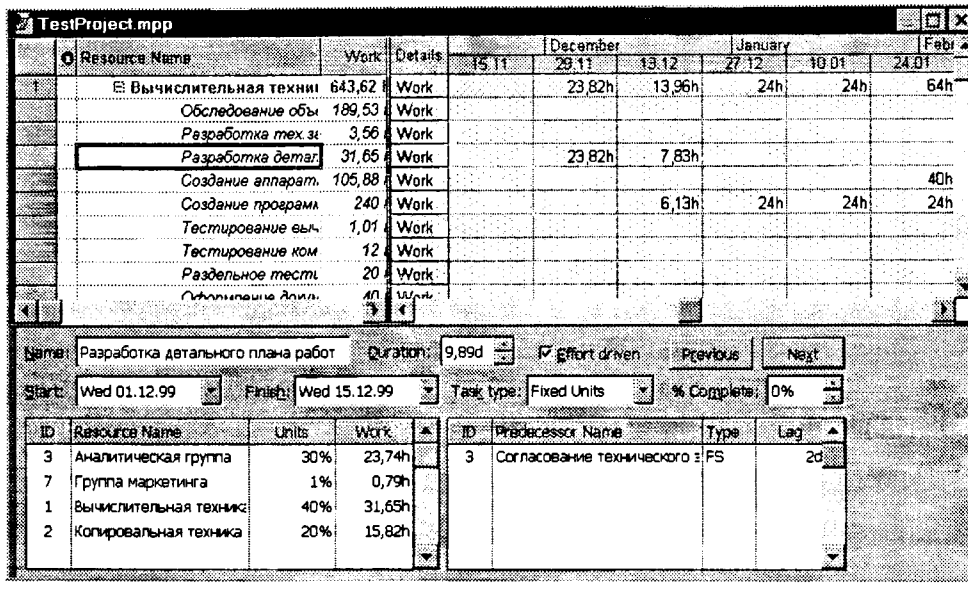


Рис. 9.40. Resource Usage — детальная таблица загрузки ресурсов

Представление Resource Usage — таблица загрузки ресурсов (с детализацией по операциям) — решает аналогичную задачу с той лишь разницей, что данные сначала группируются по видам ресурсов, а затем — по операциям. Таким образом, данное представление позволяет проводить анализ количественного распределения отдельно взятого ресурса по различным этапам проекта (рис. 9.40).

Иногда оказывается полезным совмещать количественный анализ распределения ресурсов во времени с визуальным представлением данного процесса, для чего служит Resource Graph — диаграмма загрузки ресурсов. Пример соответствующего вида проекта приведен на рис. 9.41. Как можно заметить, несомненным достоинством данного способа работы с информацией является наглядное выделение тех периодов, на которых происходит превышение заранее установленного предела на доступный объем ресурса (Max. Units). Также с помощью различной закраски столбцов диаграммы можно оценить соотношение доступной (Allocated) и требующейся сверх лимита (OverAllocated) долей ресурса.

Наконец, при решении задач сводного планирования и управления ресурсной базой проекта может быть применена таблица Resource Sheet — сводная таблица загрузки ресурсов (рис. 9.42).

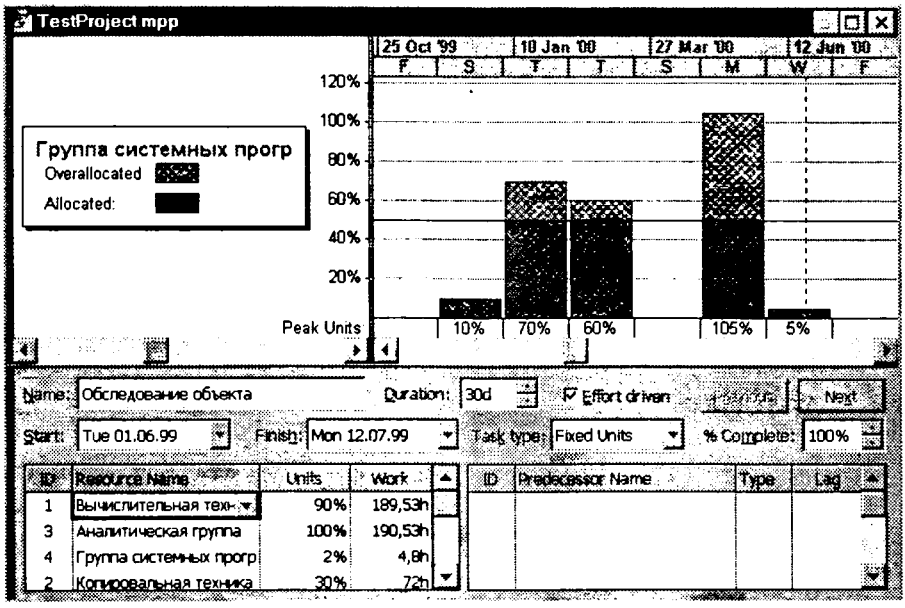


Рис. 9.41. Resource Graph — диаграмма загрузки ресурсов

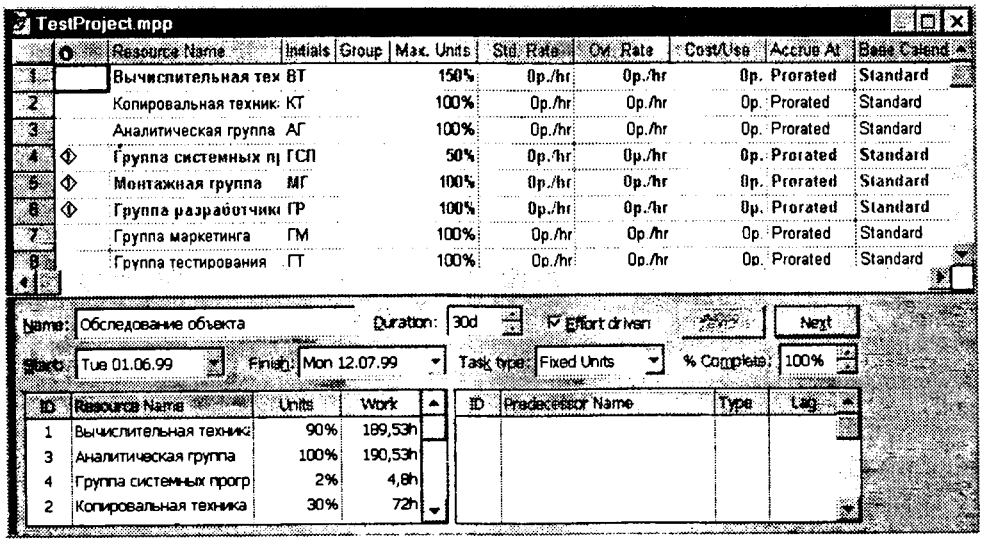


Рис. 9.42. Resource Sheet — сводная таблица загрузки ресурсов

Данная форма работы, во-первых, позволяет оперативно управлять атрибутами ресурса, а во-вторых — выявлять те ресурсы, по которым допускается превышение (на уровне всего проекта) ограничения на доступный объем ресурса, заданный в поле Max. Units. Записи таблицы по таким ресурсам выделяются цветом, а также, как это видно из рис. 9.42, помечаются пиктограммой.

9.4. Финансовое управление проектом

9.4.1. Технология управления финансами проекта в MS Project

Теория и практика управления финансами работ проектного типа является составной частью специального раздела экономической науки — *финансового менеджмента*. Принципиальной особенностью задач данного класса является то, что управление финансами проекта предполагает анализ и воздействие на будущие затраты и доходы, в то время как, допустим, бухгалтерский учет всегда обращен к уже произведенным затратам и полученным доходам.

Необходимо обратить внимание читателя на то, что здесь речь о финансовом управлении пойдет исключительно с точки зрения изучения информационной поддержки, которую обеспечивает ему MS Project. Более того, нужно четко выделить тот факт, что возможности данной программной системы позволяют ей работать лишь с весьма ограниченным кругом задач. Это прежде всего задачи, связанные с накоплением исходной информации по финансовым затратам, ее первичная обработка и относительно несложный анализ. Для решения более серьезных задач: обеспечения полномасштабного финансового менеджмента и бизнес-планирования — существуют специальные программные пакеты, например «Альт-Инвест», Project Expert, «Бизнес-План Про».

К типичным задачам финансового управления проектом, которые могут решаться в среде MS Project, относятся:

- предварительная оценка финансовой состоятельности проекта;
- управление финансовыми ресурсами на основе сопоставления и анализа фактических и плановых затрат.

Оценка предполагаемых потребностей в финансовых ресурсах основывается на исходной информации о графике проекта. При этом, вообще говоря, могут быть учтены компоненты трех типов:

- затраты, обусловленные фактом существования некоторой операции, или так называемые постоянные затраты на операцию;
- затраты, связанные с фактом использования ресурса некоторого вида (например, плата за заказ);
- затраты, пропорциональные количеству используемого ресурса.

Таким образом, формально затраты на некоторую операцию с номером i определяются как

$$\mathrm{Cost}_i = \mathrm{FixedCost}_i + \mathrm{PerUseCost}_j + \sum_j \mathrm{StandartRate}_j(\mathrm{Work}_{ij} - \mathrm{OvtWork}_{ij}) +$$

$$+ \sum_j \mathrm{OvertimeRate}_j \mathrm{OvtWork}_{ij},$$

где $Cost_i$ — суммарные финансовые затраты на операцию i; $FixedCost_i$ — постоянные затраты на выполнение i-й операции; $PerUseCost_j$ — плата за факт использования j-го ресурса (плата за заказ); $StandardRate_j$ — тарифная ставка за использование j-го ресурса в рамках плановой работы (например, рублей в день); $OvertimeRate_i$ — тарифная ставка за использование j-го ресурса на условиях сверхурочной работы (Overtime Work); $Work_{ij}$ — трудоемкость работы (использования) j-го ресурса для i-й операции (в единицах времени); $OvtWork_{ij}$ — трудоемкость работы (использования) j-го ресурса для i-й операции на условиях сверхурочной работы (в единицах времени).

Здесь необходимо пояснить тот смысл, который в MS Project вкладывается в понятие *«сверхурочная работа»*. Как уже отмечалось, продолжительность операции (Duration) прямо пропорционально зависит от ее трудоемкости (Work). В то же время, продолжительность может быть снижена за счет так называемой сверхурочной работы (Overtime Work), которая не входит в основной график, но должна оплачиваться по иным (как правило, повышенным) тарифным ставкам (Overtime Rate). Другими словами, часть «работы» Work переводится в категорию «сверхурочной (дополнительной) работы» Overtime Work, но за это следует платить больше.

Соответственно, затраты на весь проект определяются как сумма затрат на операции:

$$ProjectCost = \sum_i Cost_i .$$

Валюта, в которой будут измеряться финансовые показатели, определяется при конфигурации параметров программы (меню Tools ▸ Options..., вкладка View, секция Currency).

9.4.2. Формирование информационной базы для финансовых расчетов

Как нетрудно предположить, отправным пунктом для действий по организации управления финансами проекта являются операции по формированию исходной информационной базы:

Формирование таблиц тарифных ставок операций

Для формирования таблиц тарифных ставок можно воспользоваться следующим алгоритмом.

1. Выбрать вид проекта Resource Sheet (либо другое представление проекта, содержащее перечень ресурсов).
2. Щелкнув левой кнопкой мыши на записи, соответствующей выбранному ресурсу, активизировать окно диалога Resource Information, в котором выбрать вкладку Costs.

ПРИМЕЧАНИЕ

Вкладка Costs содержит набор таблиц тарифных ставок (в качестве их имен используются латинские литеры от A до E). Такая организация позволяет хранить в проекте одновременно несколько ценовых политик и оперативно переходить от одной к другой[1].

3. Выбрать нужную таблицу тарифов от A до E.
4. В строке тарифной таблицы заполнить поля (рис. 9.43):
   - Standard Rate — ставка за использование ресурсов в рамках установленных лимитов;
   - OverTime Rate — ставка за использование ресурсов в рамках сверх установленных лимитов;
   - Per Use Cost — плата за факт использования ресурса (плата за заказ).

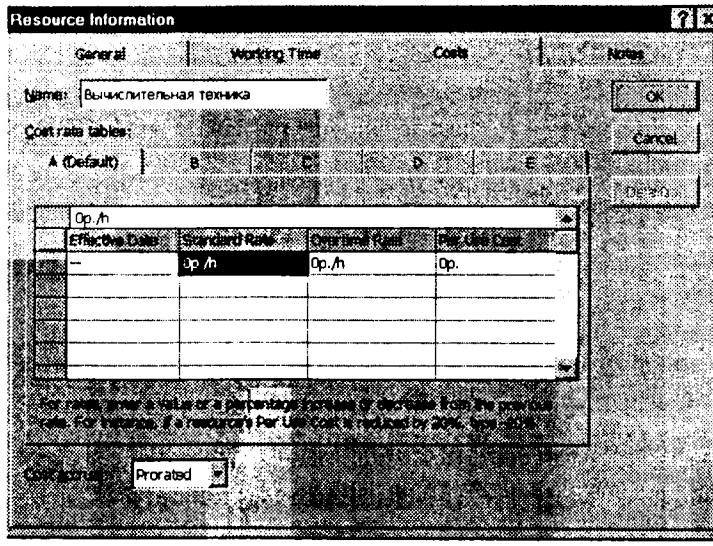


Рис. 9.43. Формирование таблицы тарифных ставок за использование ресурса

ПРИМЕЧАНИЕ

Как видно из рис. 9.43, структура тарифной таблицы позволяет достаточно гибко учитывать изменения системы цен, так как она позволяет привязывать значения тарифов к временным периодам (в поле Effective Date вводится дата, до которой действуют ставки, содержащиеся в текущей строке).

Поле Cost accrual определяет порядок вычисления затрат в процессе выполнения проекта. Оно может принимать значения:

- Start — финансовые затраты привязываются к моменту начала операции;
- End — финансовые затраты привязываются к моменту завершения операции;
- Prorated — финансовые затраты распределяются в процессе выполнения операции пропорционально интенсивности работы.

[1] По умолчанию используется первая таблица (A).

Для определения того, какую тарифную таблицу следует использовать при исчислении затрат, следует выполнить следующие действия:

1. Выбрать вид проекта Task Usage.
2. Если в таблице отсутствует колонка Cost Rate Table, то для того, чтобы ее вставить, необходимо:
   - выполнить команду меню Insert ▸ Column;
   - в поле Field Name в появившемся окне Column Definition установить значение Cost Rate Table и нажать кнопку OK.
3. Ввести в строку, соответствующую ресурсу, для которого меняется система цен, в поле Cost Rate Table код выбираемой тарифной таблицы (A, B, C, D или E) (рис. 9.44).

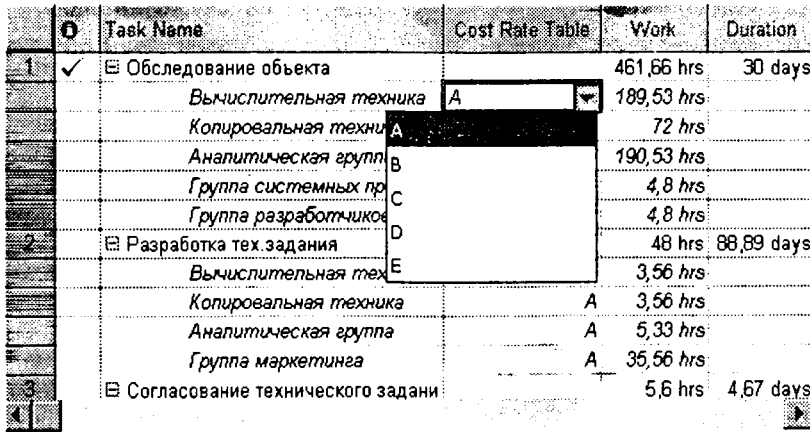


Рис. 9.44. Выбор тарифной таблицы

Объем сверхурочной работы по операции задается с помощью следующих действий:

1. Выбрать вид проекта Gantt Chart.
2. Окно представления должно быть составным и содержать дополнительную таблицу в нижней части. Если это не так, то выполнить команду меню Window ▸ Split.
3. Для таблицы в нижней части экрана выбрать тип Task Form:
   - выполнить команду меню View ▸ MoreViews...;
   - в появившемся диалоговом окне MoreViews выбрать представление проекта Task Form.
4. Выбрать Format ▸ Details ▸ Resource Work.
5. Настроенная данным образом таблица, расположенная в нижней части окна, содержит колонку Ovt. Work. В нее при необходимости могут вводиться объемы сверхурочных работ по соответствующим операциям (рис. 9.45).

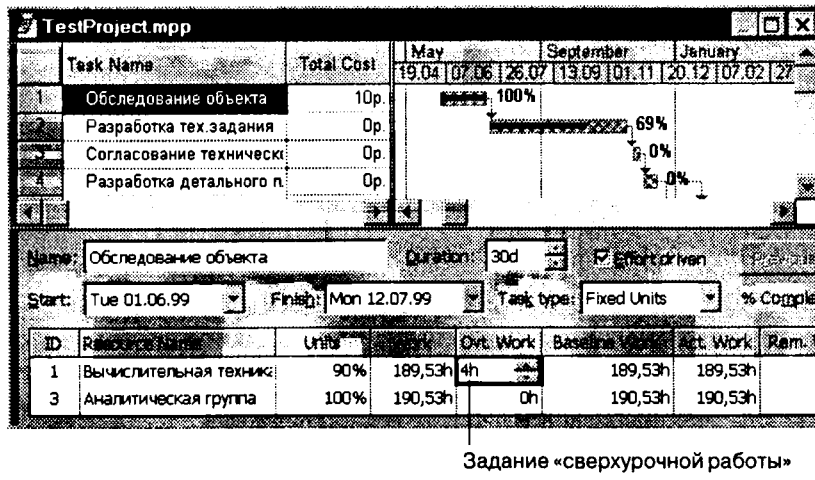


Рис. 9.45. Задание сверхурочного (дополнительного) использования ресурса по операции

ПРИМЕЧАНИЕ

Как нетрудно догадаться, необходимые колонки в таблицы могут быть добавлены с помощью функции меню Insert ▸ Column.

Ввод показателей прямых затрат

Напомним еще раз, что под *прямыми* (*постоянными*, Fixed Cost) *затратами* на операцию подразумеваются расходы, существование которых обусловливается самим фактом наличия данной операции в проекте, то есть они не связаны с такими ее характеристиками, как продолжительность и трудоемкость. Очевидно, что такой тип затрат присущ только определенным видам экономической деятельности. Одновременно хочется обратить внимание читателя на то, что именно наличие атрибута постоянных затрат позволяет на простейшем уровне вести учет невозобновляемых ресурсов проекта.

Для задания значения постоянных затрат следует:

1. Выбрать вид View ▸ Gantt Chart.
2. Выполнить команду View ▸ Table и выбрать из меню таблицу Cost.
3. По умолчанию выбранная таблица Cost содержит колонку, в которую следует вводить значение постоянных затрат по операциям (рис. 9.46).

9.4.3. Основные методы решения задач финансового управления проектом

Для решения простейших задач финансового управления проектом, перечисленных в 9.4.1, могут эффективно применяться такие формы представления проекта, как Task Usage, Resource Usage, Resource Graph. Другим полезным инструментом являются отчеты группы Costs (см. рис. 9.27, 9.47), в которую входят:

- Cash Flow — таблица распределения затрат по интервалам времени с разбивкой по операциям;
- Budget — таблица стоимостных показателей операций;

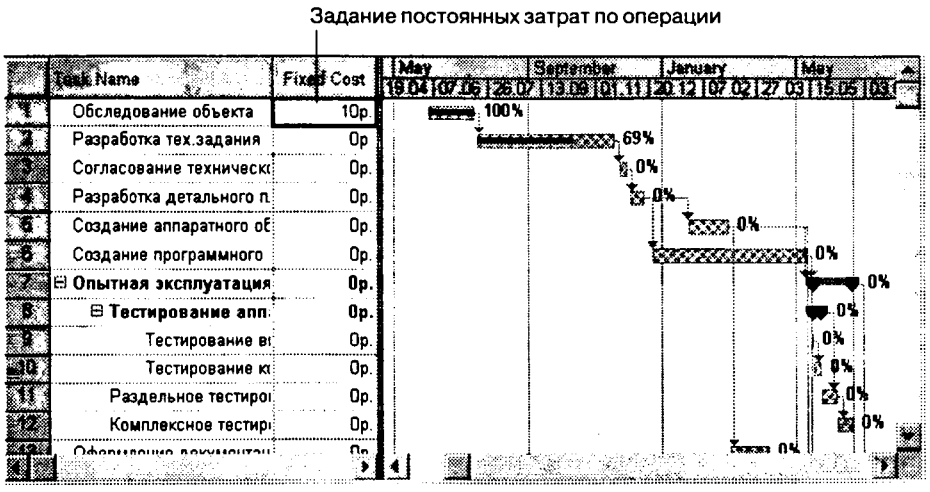


Рис. 9.46. Задание постоянных затрат по операции

- Overbudget Tasks — отчет по операциям, затраты на выполнение которых превысили исходные показатели базового плана;
- Overbudget Resources — таблица ресурсов (в стоимостном исчислении), затраты на которые превысили исходные показатели базового плана;
- Earned Value — таблица сопоставления плановых стоимостных данных с отчетными.

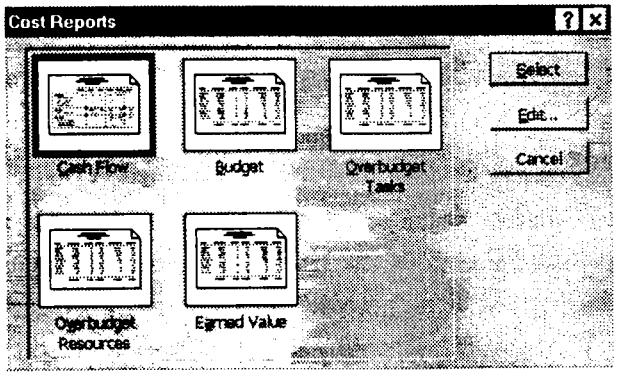


Рис. 9.47. Отчеты группы Costs

Перечисленные экранные формы и отчеты, в частности, могут быть использованы:

- на этапе предварительной оценки финансовой состоятельности проекта;
- при разработке плана финансовых затрат по проекту;

- при управлении финансовыми ресурсами в процессе выполнения проекта (сопоставление плановых и фактических показателей как в разрезе ресурсов, так и в разрезе исполнителей).

Ключевые понятия

- проект;
- сетевое планирование и управление;
- методы PERT и МКП;
- вершинные и стрелочные графы;
- график (диаграмма) Гантта;
- сетевая диаграмма (PERT Chart);
- критический путь;
- базовый план (Baseline);
- промежуточный план (Interim plan);
- операция;
- ресурс;
- визуальное представление проекта.

Контрольные вопросы

1. Охарактеризуйте основные направления автоматизации процессов управления предприятием (фирмой).
2. Приведите примеры программного обеспечения, решающего задачи управления проектами.
3. Охарактеризуйте основные классы задач, возникающих в области управления проектами.
4. Какие основные методы сетевого планирования и управления вы можете назвать?
5. Перечислите основные этапы процесса сетевого планирования.
6. Назовите основные способы представления логических схем проектов.
7. Перечислите известные вам программные средства, решающие задачи управления работами проектного типа.
8. Дайте характеристику программному обеспечению MS Project.
9. Раскройте содержание понятий «операция», «ресурс».
10. Как связаны продолжительность операции, ее трудоемкость и количество единиц ресурса?
11. Перечислите основные средства ПО MS Project, предназначенные для решения задач управления ресурсами проекта.

12. Дайте характеристику ПО MS Project в плане возможностей его применения для решения задач финансового управления проектом.
13. Поясните смысл понятия «сверхурочная работа» в MS Project.

Литература

1. Гербер Б. Microsoft Exchange Server 5.5 для профессионалов. СПб: Питер Ком, 1999.
2. Куперштейн В. И. Современные информационные технологии в делопроизводстве и управлении. СПб.: BHV, 1999.
3. Нейбауэр А. Эффективная работа с Microsoft Outlook 2000. СПб.: Питер, 2000.
4. Клейтон Р. Lotus Notes 5: учебный курс. СПб.: Питер, 2000.
5. Мак-Би Дж. Exchange Server 5.5. Эффективная работа. СПб.: Питер, 2000.
6. Эддоус М., Стэнсфилд Р. Методы принятия решений / Пер с англ:, Под ред. И. И. Елисеевой. М.: Аудит, ЮНИТИ 1997.

Глава 10. Локальные вычислительные сети

Неизбежным и достаточно очевидным этапом развития любой компьютерной системы является этап организации взаимодействия между различными вычислительными машинами. Средства и методы, которые используются при решении данных задач, составляют основное содержание настоящей главы.

10.1. Принципы построения вычислительных сетей

10.1.1. Программные и аппаратные компоненты вычислительной сети

Вычислительная сеть (далее сеть) — сложная система программных и аппаратных компонентов, взаимосвязанных друг с другом. Среди аппаратных средств можно выделить компьютеры и коммуникационное оборудование. Программные компоненты состоят из операционных систем и сетевых приложений.

В настоящее время в сети используются компьютеры различных типов и классов с различными характеристиками. Это основа любой вычислительной сети. Компьютеры, их характеристики определяют возможности вычислительной сети. Но в последнее время и *коммуникационное оборудование* (кабельные системы, повторители, мосты, маршрутизаторы и др.) стало играть не менее важную роль. Некоторые из этих устройств, учитывая их сложность, стоимость и другие характеристики, можно назвать компьютерами, решающими сугубо специфические задачи по обеспечению работоспособности сетей.

Для эффективной работы сетей используются специальные операционные системы (ОС), которые, в отличие от персональных операционных систем, предназначены для решения специальных задач по управлению работой сети компьютеров. Это *сетевые операционные системы*. Сетевые ОС устанавливаются на специально выделенные компьютеры.

Сетевые приложения — это прикладные программные комплексы, которые расширяют возможности сетевых ОС. Среди них можно выделить почтовые програм-

мы, системы коллективной работы, сетевые базы данных и др. В процессе развития сетевых ОС некоторые функции сетевых приложений становятся обычными функциями ОС.

Все устройства, подключаемые к сети, можно разделить на три функциональные группы:

- рабочие станции;
- серверы сети;
- коммуникационные узлы.

Рабочая станция (workstation) — это персональный компьютер, подключенный к сети, на котором пользователь сети выполняет свою работу. Каждая рабочая станция обрабатывает свои локальные файлы и использует свою операционную систему. Но при этом пользователю доступны ресурсы сети. Можно выделить три типа рабочих станций: рабочая станция с локальным диском, бездисковая рабочая станция, удаленная рабочая станция.

На рабочей станции с диском (жестким или гибким) операционная система загружается с этого локального диска. Для бездисковой станции операционная система загружается с диска файлового сервера. Такая возможность обеспечивается специальной микросхемой, устанавливаемой на сетевом адаптере бездисковой станции. Удаленная рабочая станция — это станция, которая подключается к локальной сети через телекоммуникационные каналы связи (например, с помощью телефонной сети).

Сервер сети (server) — это компьютер, подключенный к сети и предоставляющий пользователям сети определенные услуги, например хранение данных общего пользования, печать заданий, обработку запроса к СУБД, удаленную обработку заданий и т. д. По выполняемым функциям можно выделить следующие группы серверов.

Файловый сервер (file server) — компьютер, хранящий данные пользователей сети и обеспечивающий доступ пользователей к этим данным. Как правило, этот компьютер имеет большой объем дискового пространства. Файловый сервер обеспечивает одновременный доступ пользователей к общим данным. Файловый сервер выполняет следующие функции:

- хранение данных;
- архивирование данных;
- согласование изменений данных, выполняемых разными пользователями;
- передачу данных.

Сервер баз данных — компьютер, выполняющий функции хранения, обработки и управления файлами баз данных (БД). Сервер баз данных выполняет следующие функции:

- хранение баз данных, поддержку их целостности, полноты, актуальности;
- прием и обработку запросов к базам данных, а также пересылку результатов обработки на рабочую станцию;

- обеспечение авторизированного доступа к базам данных, поддержку системы ведения и учета пользователей, разграничение доступа пользователей;
- согласование изменений данных, выполняемых разными пользователями;
- поддержку распределенных баз данных, взаимодействие с другими серверами баз данных, расположенными в другом месте.

Сервер прикладных программ (application server) — компьютер, который используется для выполнения прикладных программ пользователей.

Коммуникационный сервер (communications server) — устройство или компьютер, который предоставляет пользователям локальной сети прозрачный доступ к своим последовательным портам ввода/вывода. С помощью коммуникационного сервера можно создать разделяемый модем, подключив его к одному из портов сервера. Пользователь, подключившись к коммуникационному серверу, может работать с таким модемом так же, как если бы модем был подключен непосредственно к рабочей станции.

Сервер доступа (access server) — это выделенный компьютер, позволяющий выполнять удаленную обработку заданий. Программы, инициируемые с удаленной рабочей станции, выполняются на этом сервере. От удаленной рабочей станции принимаются команды, введенные пользователем с клавиатуры, а возвращаются результаты выполнения задания.

Факс-сервер (fax server) — устройство или компьютер, который выполняет рассылку и прием факсимильных сообщений для пользователей локальной сети.

Сервер резервного копирования данных (back up server) — устройство или компьютер, который решает задачи создания, хранения и восстановления копий данных, расположенных на файловых серверах и рабочих станциях. В качестве такого сервера может использоваться один из файловых серверов сети.

Следует отметить, что все перечисленные типы серверов могут функционировать на одном выделенном для этих целей компьютере.

К *коммуникационным узлам* сети относятся следующие устройства:

- повторители;
- коммутаторы (мосты);
- маршрутизаторы;
- шлюзы.

Протяженность сети, расстояние между станциями в первую очередь определяются физическими характеристиками передающей среды (коаксиального кабеля, витой пары и т. д.). При передаче данных в любой среде происходит затухание сигнала, что и приводит к ограничению расстояния. Чтобы преодолеть это ограничение и расширить сеть, устанавливают специальные устройства — повторители, мосты и коммутаторы. Часть сети, в которую не входит устройство расширения, принято называть *сегментом* сети.

Повторитель (repeater) — устройство, усиливающее или регенерирующее пришедший на него сигнал. Повторитель, приняв пакет из одного сегмента, передает

его во все остальные. При этом повторитель не выполняет развязку присоединенных к нему сегментов. В каждый момент времени во всех связанных повторителем сегментах поддерживается обмен данными только между двумя станциями.

Коммутатор (switch), или *мост* (bridge) — это устройство, которое, как и повторитель, позволяет объединять несколько сегментов. В отличие от повторителя, мост выполняет развязку присоединенных к нему сегментов, то есть одновременно поддерживает несколько процессов обмена данными для каждой пары станций разных сегментов.

Маршрутизатор (router) — устройство, соединяющее сети одного или разных типов по одному протоколу обмена данными. Маршрутизатор анализирует адрес назначения и направляет данные по оптимально выбранному маршруту.

Шлюз (gateway) — это устройство, позволяющее организовать обмен данными между разными сетевыми объектами, использующими разные протоколы обмена данными.

10.1.2. Основные требования, предъявляемые к современным вычислительным сетям

Вычислительная сеть создается для обеспечения потенциального доступа к любому ресурсу сети для любого пользователя сети. Качество доступа к ресурсу как глобальная характеристика функционирования сети может быть описана многими показателями, выбор которых зависит от задач, стоящих перед вычислительной сетью. Среди основных показателей можно выделить следующие:

- производительность;
- надежность;
- управляемость;
- расширяемость;
- прозрачность.

Производительность

Производительность вычислительной сети может быть оценена с разных позиций. С точки зрения пользователя, важным числовым показателем производительности сети является *время реакции* системы, особенно в той части, которая относится к работе сети. Время реакции — это время между моментом возникновения запроса и моментом получения ответа. Время реакции зависит от многих факторов, таких как используемая служба сети, степень загруженности сети или отдельных сегментов и др. Поэтому при оценке производительности работы сети определяется среднее время реакции.

Пропускная способность сети определяется количеством информации, переданной через сеть или ее сегмент в единицу времени. Пропускная способность сети характеризует, насколько быстро сеть может выполнить свою основную задачу передачи информации. Пропускная способность определяется в битах в секунду.

Надежность

Надежность работы вычислительной сети определяется надежностью работы всех ее компонентов. Для повышения надежности работы аппаратных компонентов обычно используют дублирование, когда при отказе одного из элементов функционирование сети обеспечат другие.

При работе вычислительной сети должна обеспечиваться сохранность информации и защита ее от искажений. Как правило, информация в сети хранится в нескольких экземплярах (для повышения надежности). В этом случае необходимо обеспечить согласованность данных (например, идентичность копий при изменении информации).

Одной из функций вычислительной сети является передача информации (передача осуществляется порциями, которые называются пакетами), во время которой возможны ее потери и искажения. Для оценки надежности исполнения этой функции используются показатели вероятности потери пакета при его передаче, либо вероятности доставки пакета.

В современных вычислительных сетях важное значение имеет другая сторона надежности — *безопасность*. Это способность сети обеспечить защиту информации от несанкционированного доступа. Задачи обеспечения безопасности решаются применением как специального программного обеспечения, так и соответствующих аппаратных средств.

Управляемость

При работе вычислительной сети, которая в идеале объединяет отдельные компьютеры в единое целое, необходимы средства не только для наблюдения за работой сети, сбора разнообразной информации о функционировании сети, но и средства управления сетью. В общем случае система управления сетью должна предоставлять возможность воздействовать на работу любого элемента сети. Должна быть обеспечена возможность осуществлять мероприятия по управлению с любого элемента сети. Управлением сетью занимается администратор сети или пользователь, которому поручены эти функции. Обычный пользователь, как правило, не имеет административных прав.

Другими характеристиками *управляемости* являются возможность определения проблем в работе вычислительной сети или отдельных ее сегментов, выработка управленческих действий для решения выявленных проблем и возможность автоматизации этих процессов при решении похожих проблем в будущем.

Расширяемость

Любая вычислительная сеть является развивающимся объектом, и не только в плане модернизации ее элементов, но и в плане ее физического расширения, добавления новых элементов сети (пользователей, компьютеров, служб). Существование таких возможностей, трудоемкость их осуществления входят в понятие *расширяемости*. Другой похожей характеристикой является *масштабируемость* сети, которая определяет возможность расширения сети без существенного снижения ее производительности. Обычно одноранговые сети обладают хорошей

расширяемостью, но плохой масштабируемостью. В таких сетях легко добавить новый компьютер, используя дополнительный кабель и сетевой адаптер, но существуют ограничения на количество подключаемых компьютеров в связи с существенным падением производительности сети. В многосегментных сетях используются специальные коммуникационные устройства, которые позволяют подключать к сети значительное количество дополнительных компьютеров без снижения общей производительности сети.

Прозрачность

Прозрачность вычислительной сети является ее характеристикой с точки зрения пользователя. Эта важная характеристика должна оцениваться с разных сторон.

Прозрачность сети предполагает скрытие (невидимость) особенностей сети от конечного пользователя. Пользователь обращается к ресурсам сети как к обычным локальным ресурсам компьютера, на котором он работает.

Вычислительная сеть объединяет компьютеры разных типов с разными операционными системами. Пользователю, у которого установлена, например, Windows, прозрачная сеть должна обеспечивать доступ к необходимым ему при работе ресурсам компьютеров, на которых установлена, например, UNIX. Другой важной стороной прозрачности сети является возможность распараллеливания работы между разными элементами сети. Вопросы назначения отдельных параллельных заданий отдельным устройствам сети также должны быть скрытыми от пользователя и решаться в автоматическом режиме.

Интегрируемость

Интегрируемость означает возможность подключения к вычислительной сети разнообразного и разнотипного оборудования, программного обеспечения от разных производителей. Если такая неоднородная вычислительная сеть успешно выполняет свои функции, то можно говорить о том, что она обладает хорошей интегрируемостью.

Современная вычислительная сеть имеет дело с разнообразной информацией, процесс передачи которой сильно зависит от типа информации. Передача традиционных компьютерных данных характеризуется неравномерной интенсивностью. При этом нет жестких требований к синхронности передачи. При передаче мультимедийных данных качество передаваемой информации в существенной степени зависит от синхронизации передачи. Сосуществование двух типов данных с противоположными требованиями к процессу передачи является сложной задачей, решение которой является необходимым условием вычислительной сети с хорошей интегрируемостью.

Основным направлением развития интегрируемости вычислительных сетей является стандартизация сетей, их элементов и компонентов. Все стандарты можно разделить на следующие виды:

- стандарты отдельных фирм;
- стандарты специальных комитетов и объединений, создаваемых несколькими фирмами;

- стандарты национальных организаций по стандартизации;
- международные стандарты.

Работы по стандартизации вычислительных сетей ведутся большим количеством организаций. Среди них необходимо выделить те, которые давно и успешно работают в области стандартизации вычислительных сетей.

Международная организация по стандартизации (International Organization for Standardization — ISO). Эта организация известна разработкой модели взаимодействия открытых систем, которая в настоящее время является основной, своего рода «эталонной» моделью вычислительной сети. Эта модель является основой стандартизации в области вычислительных сетей.

Международный союз электросвязи (International Telecommunication Union, ITU) — организация при Организации Объединенных Наций, в которой существует телекоммуникационный сектор (ITU-T). ITU-T отвечает за разработку стандартов в области телекоммуникационного оборудования и услуг (телефонии, электронной почты, факсимильной связи, телетекста, телекса, передачи данных, аудио- и видеосигналов).

Институт инженеров по электротехнике и радиоэлектронике — Institute of Electrical and Electronic Engineers, IEEE — национальная организация США, определяющая стандарты электронных коммуникаций. Самыми известными его стандартами являются стандарты, разработанные группой 802 (802.1, 802.2, 802.3 и 802.5), которые описывают общие понятия, используемые в области локальных сетей.

Европейская ассоциация производителей компьютеров (ECMA) — некоммерческая организация, активно сотрудничающая с ITU-T и ISO. Она занимается разработкой стандартов и технических обзоров, относящихся к компьютерной и коммуникационной технологиям.

Американский национальный институт стандартов — American National Standarts Institute, ANSI. ANSI представляет США в международной организации ISO. Стандарт технологии FDDI является разработкой этого института.

10.1.3. Классификации вычислительных сетей

Классификация по территориальному признаку

Локальная сеть, Local Area Network (LAN), объединяет компьютеры, как правило, одной организации, которые располагаются компактно в одном или нескольких зданиях. Размер локальной сети не превышает нескольких километров. Небольшие расстояния между компьютерами экономически оправдывают прокладку новых высококачественных линий связи, которые позволяют использовать простые алгоритмы и процедуры передачи данных и относительно дешевые коммуникационные устройства. Пропускная способность современных локальных сетей достигает 1000 Мбит/с. Время обращения к сетевым ресурсам соизмеримо со временем обращения к локальным ресурсам рабочей станции.

Высокое качество передачи данных дает возможность сети предоставлять пользователю широкий спектр услуг: файловую службу, печать, факс, электронную почту,

сканер, базы данных и другие услуги, реализация которых отдельно на локальном компьютере непозволительно дорога. Каналы связи могут использоваться совместно сразу многими компьютерами сети. При этом возможно использование современных методов передачи данных, позволяющих сгладить неравномерность нагрузки на линии связи при передаче компьютерных данных (метод коммутации пакетов описывается далее).

Локальные сети обладают плохой масштабируемостью, так как используемые в них технологии накладывают жесткие ограничения на длину линий связи и количество подключаемых компьютеров.

Глобальные сети, Wide Area Network (WAN), объединяют компьютеры, которые могут располагаться на значительном расстоянии друг от друга. В общем случае компьютер может находиться в любой точке земного шара. Это обстоятельство делает экономически невозможным прокладку линий связи к каждому компьютеру. При организации WAN-сетей используются уже существующие линии связи, например телефонные линии. Эти линии прокладывались для целей, отличных от передачи компьютерных данных. Качество таких линий связи, как правило, очень низкое, что требует использования специальных сложных алгоритмов и процедур передачи данных и дорогой аппаратуры. Скорость обмена данных существенно ниже, чем в LAN-сетях. Количество предоставляемых услуг меньше. Линия связи, как правило, используется монопольно отдельной парой компьютеров на время их связи, что не способствует эффективному использованию каналов связи.

Глобальные сети обладают хорошей масштабируемостью. Подключение дополнительных компьютеров практически не влияют на общие показатели всей сети.

Особенности современных локальных и глобальных вычислительных сетей

Основное отличие локальных сетей от глобальных заключается в использовании качественных линий связи. Все остальные отличия являются производными. В настоящее время в глобальных сетях улучшается качество каналов связи. Ярким примером может служить применение оптоволоконной техники, стоимость которой приблизилась к стоимости традиционных линий связи. Скорость передачи данных существенно возросла и приблизилась к скорости передачи данных в локальных сетях. В результате службы, которые были прерогативой локальных сетей, стали активно применяться и в глобальных сетях.

Наблюдается и обратное проникновение технологий. В локальных сетях применяются транспортные технологии глобальных сетей. Наряду с разделяемыми линиями связи, стандарты локальных сетей поддерживают работу по индивидуальным линиям связи.

Разработанные для применения в глобальных открытых сетях методы защиты информации от несанкционированного доступа находят широкое применение в

локальных сетях. Эта необходимость вызвана тем обстоятельством, что обособленные ранее локальные сети объединяются, используя глобальные связи. Локальные сети становятся доступными практически для любого пользователя глобальной сети.

Процесс переноса технологий глобальных сетей в мир локальных сетей получил в последнее время самое широкое развитие. Появилось понятие *intranet-технология*, которое обозначает применение служб глобальных сетей для реализации целей, которые ставятся перед локальными сетями. Масштабы локальных сетей перестали определяться лишь территориальными признаками.

Классификация по масштабу сети

Локальные *сети рабочих групп* объединяют небольшое количество компьютеров, работающих, как правило, под управлением одной операционной среды. В сети выделен один компьютер, который выполняет сетевые службы, например файловый сервер, сервер печати, сервер факса.

Локальные *сети отделов* могут объединять компьютеры целого отдела. Количество компьютеров может быть в несколько раз больше, чем в сетях рабочих групп. Сетевые службы могут быть распределены между отдельными выделенными компьютерами-серверами.

Оба типа локальных сетей используют одну из базовых технологий. Эти сети можно назвать классическими локальными сетями. Территориально они могут занимать небольшую площадь, одно-два здания.

Сети кампусов преследуют цель объединения нескольких мелких сетей в одну большую сеть. Такие сети могут занимать большие территории. При объединении сетей в одну необходимо решать вопросы интеграции разнородных технологий, программного и аппаратного обеспечения. Значительный сетевой трафик локализован в рамках сетей отделов и рабочих групп. Сеть кампусов решает задачи взаимодействия отдельных сетей, обеспечивает доступ к общесетевым ресурсам, как правило, дорогостоящим. При объединении не используются глобальные связи.

Корпоративные сети объединяют компьютеры и сети в рамках одного предприятия или корпорации. Территориальный признак не имеет никакого значения. Такие сети могут охватывать любую часть земного шара. Для соединения удаленных локальных сетей и компьютеров используются телекоммуникационные средства, телефонные линии, спутниковая связь и другие современные средства.

Корпоративные сети обладают высокой степенью разнородности и разнотипности используемого оборудования, программ и технологий. При этом такие сети должны образовывать единое целое с возможностью доступа к любому ресурсу с приемлемым временем реакции.

На примере корпоративных сетей можно проследить процесс взаимного проникновения технологий локальных и глобальных сетей, рождение новой intranet-технологии.

10.1.4. Эталонная модель OSI

Для согласованной работы двух разных устройств необходимо иметь соглашение, требованиям которого будет удовлетворять работа каждого устройства. Соглашение, как правило, оформляется в виде стандарта.

Взаимодействие устройств в вычислительной сети является сложным процессом, реализация которого требует решения многих взаимосвязанных проблем и задач. Обычно сложная проблема разделяется на отдельные части. Решение каждой отдельной части представляет собой относительно простую задачу. Решение всех частей в сумме дает решение поставленной проблемы. Рассматриваемая ниже модель была разработана, исходя именно из этого принципа.

В начале 80-х годов международная организация по стандартизации ISO при поддержке других организаций по стандартизации разработала модель взаимодействия открытых систем (Open System Interconnection, OSI), *модель OSI*. Схема модели приведена на рис. 10.1.

Модель OSI очень быстро стала одной из основных моделей, описывающих процесс передачи данных между компьютерами.

Модель OSI описывает системные средства взаимодействия, реализуемые операционной системой, системными утилитами, системными аппаратными средствами. Модель не описывает взаимодействия приложений с конечным пользователем.

Модель OSI разделяет средства взаимодействия на семь уровней:

1. Прикладной уровень.
2. Представительский уровень.
3. Сеансовый уровень.
4. Транспортный уровень.
5. Сетевой уровень.
6. Канальный уровень.
7. Физический уровень.

Каждый уровень относительно независим. Модули реализации каждого уровня могут быть легко заменены без внесения изменений в модули других уровней.

Каждый уровень описывает строго определенные функции взаимодействия сетевых устройств. Все уровни образуют иерархическую систему, в которой запрос, вырабатываемый на каком-либо уровне, передается на исполнение нижележащему уровню. Результаты обработки запроса передаются на вышележащий уровень.

Прикладной уровень получает запрос от приложения, работающего на компьютере, в виде сообщения, которое нужно передать на другой компьютер. Физический уровень занимается собственно пересылкой подготовленных данных по физическим линиям. Для описания взаимосвязей между двумя уровнями устанавливаются правила или соглашения, которые называются *интерфейсом*. Интерфейс определяет набор сервисов, предоставляемый уровнем соседнему уровню.

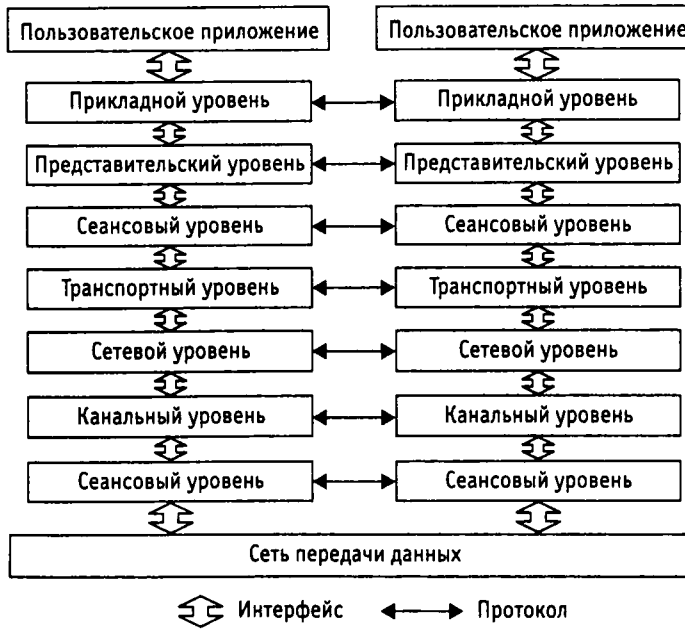


Рис. 10.1. Модель OSI

В процессе обмена данными участвуют два компьютера. Процесс взаимодействия компьютеров может быть представлен как набор взаимодействий одинаковых уровней. Правила, определяющие формат сообщений между одинаковыми уровнями, называются *протоколом*. В модели OSI различаются протоколы двух типов. *Протокол с установлением соединения* предполагает, что перед началом обмена данными между компьютерами должно быть установлена связь с определенными параметрами. По *дейтаграммному протоколу* сообщение передается в сеть без предварительного установления соединения.

Средства каждого уровня отрабатывают протокол своего уровня и интерфейсы с соседними уровнями. Набор протоколов, достаточный для организации взаимодействия в сети, называется *стеком коммуникационных протоколов.*

Блоки информации, передаваемые между уровнями, имеют стандартный формат: заголовок, служебная информация, данные, завершающая информация. Каждый уровень при передаче блока информации нижележащему уровню снабжает его своими заголовками. Заголовки вышележащих уровней воспринимаются нижележащими уровнями как данные блока информации. На каждом уровне информация обрабатывается в соответствии с назначенными функциями, например, данные шифруются.

При получении блока информации от нижележащего уровня заголовки и другая служебная информация текущего уровня отбрасываются. Данные обрабатываются, например, данные дешифруются. Данные, поступающие на вышележащий уровень, уже имеют его заголовки.

Физический уровень

На *физическом уровне* определяются характеристики электрических сигналов, передающих данные, физического канала связи, типы разъемов с назначением каждого контакта.

Физический уровень описывает передачу битов информации по физическим каналам связи.

Функции физического уровня на компьютере выполняются сетевым адаптером.

Канальный уровень

Канальный уровень решает две задачи. Первая задача — определение доступности среды передачи данных. Эта задача решается в сетях с разделяемой средой передачи данных, когда в конкретный момент времени канал связи занят только одной парой компьютеров. Вторая задача — определение механизма обнаружения и коррекции ошибок. Обмен данными осуществляется определенными порциями, которые называются *кадрами*. В каждый кадр добавляются определенные последовательности бит в начало и конец для выделения кадра, адрес компьютера-отправителя, адрес компьютера-получателя. Кроме того, в каждый кадр добавляется вычисляемая контрольная сумма, которая необходима для проверки корректности передачи кадра. Исправление обнаруженной ошибки возможно за счет повторной передачи кадра.

В компьютерах протоколы канального уровня реализуются сетевыми адаптерами и их драйверами.

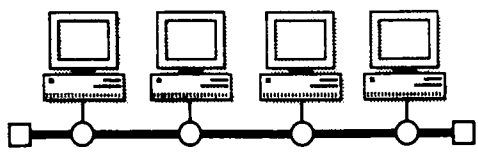


Рис. 10.2. Топология «общая шина»

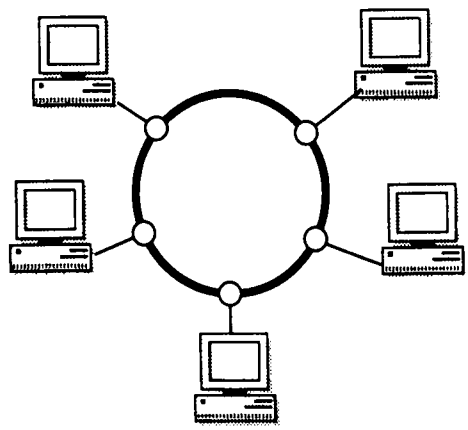


Рис. 10.3. Топология «кольцо»

Протоколы канального уровня, используемые в локальных сетях, разрабатываются для сетей с определенной топологией. *Топология* — это конфигурация графа, вершинами которого являются компьютеры сети или другие коммуникационные устройства, а ребрами — физические связи между ними. Компьютеры называются *узлами сети*. Протоколы канального уровня поддерживают топологии: общая шина (рис. 10.2), «звезда» (рис. 10.4), «кольцо» (рис. 10.3) и полученные на основе перечисленных топологий с использованием специального коммуникационного оборудования (мосты, коммуникаторы).

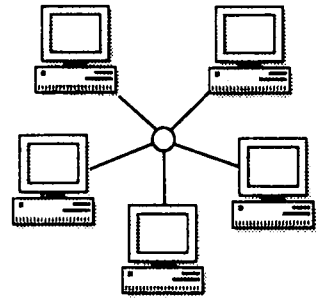


Рис. 10.4. Топология «звезда»

Использование протоколов канального уровня в глобальных сетях ограничено применением при обмене между двумя компьютерами, соединенными индивидуальной линией связи. Для обмена данными между конечными узлами разных сетей используются средства следующего, сетевого уровня.

Сетевой уровень

На *сетевом уровне* решаются вопросы объединения сетей с разными топологиями, с разными принципами передачи данных между конечными узлами для образования единой транспортной системы. Здесь сеть является не просто объединением компьютеров, но соединением по одной из типовых технологий, использующих для передачи данных один из протоколов канального уровня. Сетевой уровень решает вопросы обмена данными между сетями. Обмен данными осуществляется порциями, которые называются *пакетами*. Каждый пакет, кроме адреса компьютера, снабжается адресом сети как получателя, так и отправителя.

Для соединения сетей используется маршрутизатор, который собирает информацию о топологии межсетевых соединений. Для пути передачи пакета между конечными узлами, находящимися в разных сетях, возможно, находятся другие промежуточные сети, через которые необходимо сделать транзитные передачи. Таким образом, пакет проходит через несколько маршрутизаторов, которые образуют маршрут. Таких маршрутов может быть несколько. Проблема выбора наилучшего маршрута является главной задачей сетевого уровня, решение которой возлагается на маршрутазатор. Критерии выбора могут быть следующими: время передачи пакета, надежность передачи.

На сетевом уровне решаются вопросы согласования разных технологий, оптимизации информационных потоков между сетями.

На сетевом уровне выделяются два вида протоколов. Это *сетевые протоколы*, с помощью которых осуществляется продвижение пакетов через сеть. К ним можно отнести и так называемые *протоколы маршрутизации*, с помощью которых маршрутизаторы обмениваются маршрутной информацией. Второй вид протокола — *протокол разрешения адресов*, который отвечает за преобразование адреса узла, используемого на сетевом уровне, в локальный адрес сети.

Транспортный уровень

На *транспортном уровне* решаются вопросы обеспечения надежности передачи данных, обнаружения и исправления ошибок передачи (искажение, потеря и дублирование пакетов). Модель OSI определяет пять классов сервиса, которые определяются качеством предоставляемых услуг по надежности. Задача выбора класса сервиса решается не только приложениями и протоколами более высоких уровней, но и зависит от уровня надежности, который обеспечивается более низкими уровнями (сетевым, канальным, физическим). Если качество каналов связи, например, отечественных телефонных линий, оставляет желать лучшего, то разумно использовать более развитый сервис транспортного уровня по обеспечению надежности передачи данных.

Протоколы транспортного уровня и выше реализуются программными средствами узлов сети, компонентами сетевых систем.

Сеансовый уровень

На *сеансовом уровне* реализуются средства синхронизации, при помощи которых в длинных передачах устанавливаются специальные контрольные точки для возможного отката в случае сбоя не в начало, а на последнюю контрольную точку. Сеансовый уровень обеспечивает управление диалогом между конечными узлами.

Отдельные протоколы сеансового уровня обычно не используются. Его функции реализуются в протоколах прикладного уровня.

Представительский уровень

Функции *уровня представления* заключаются в преобразовании формы представления данных, полученных от прикладного уровня одной системы, в форму, необходимую для восприятия прикладным уровнем другой системы. На этом уровне преодолеваются синтаксические различия в представлении и кодировке данных.

На уровне представления также обеспечивается секретность обмена данными для всех служб прикладного уровня.

Прикладной уровень

Протоколы прикладного уровня обеспечивают доступ пользователей к разделяемым ресурсам сети (файлы, принтеры, факсы, сканеры, гипертекстовые страницы). К ним относятся протоколы электронной почты и другие протоколы совместной

работы. В качестве единицы информации протоколы этого уровня используют *сообщение*.

10.1.5. Методы передачи данных на физическом и канальном уровнях

На физическом уровне единицей информации является бит. Устройства физического уровня должны поддерживать *битовую синхронизацию* данных между передатчиком и приемником. Для работы на ненадежных линиях связи используется синхронизация на уровне байт. Каждый байт данных при передаче сопровождается специальными сигналами «старт» и «стоп». Эти сигналы позволяют синхронизировать передачу/прием байта данных. Такой режим называется *асинхронным*. Отдельный байт данных может быть смещен по времени от тактов других байтов данных.

На достаточно хороших линиях связи используется *синхронный режим* передачи данных, когда данные передаются в виде кадров, состоящих из последовательности байт. Перед каждым кадром передается специальный синхробайт.

При синхронном режиме скорость передачи данных выше, при асинхронном — меньше, но асинхронный режим можно применять на ненадежных линиях связи.

10.1.6. Методы коммутации

Для передачи данных между двумя компьютерами необходимо организовать физическую связь. Различают коммутируемые и некоммутируемые линии связи. *Некоммутируемая линия* связи соединяет два компьютера, которые владеют этой линией длительное время. Обеспечить такой линией связи любую пару компьютеров практически невозможно.

На практике используются *коммутируемые линии*, когда линия связи между парой компьютеров образуется только на время сеанса обмена данными между ними. В этом случае линия связи может быть использована многими компьютерами.

Различают три способа коммутации: коммутация каналов, коммутация пакетов, коммутация сообщений. Различают коммутацию динамическую и постоянную. При *динамической коммутации* соединение образуется на время обмена данными по инициативе одного из абонентов. *Постоянная коммутация* выполняется обслуживающим персоналом сети.

Коммутация каналов предполагает образование физической связи между абонентами путем соединения отдельных участков при помощи специальных устройств, которые называются *коммутаторами*. Перед началом передачи данных необходимо выполнить определенные действия по установлению соединения. Современные коммутаторы позволяют передавать данные по нескольким каналам, используя специальную технику мультиплексирования.

Коммутация пакетов используется при передаче компьютерных данных. Интенсивность обмена данными изменяется в широких пределах, что делает неэффективным использование коммутации каналов. Данные разбиваются на отдельные порции, обычно называемые *пакетами*. Каждый пакет является независимым

информационным блоком, который передается независимо от других пакетов. Одновременно по сети могут передаваться пакеты многих соединений. Конечно, коммутация каналов позволяет наиболее эффективно осуществлять обмен данными между двумя компьютерами. Но коммутация пакетов позволяет более эффективно использовать среду передачи данных. В отличие от коммутации каналов, при невозможности передачи пакета данных происходит его буферизация средствами коммутатора для последующей его передачи в следующий возможный момент. Это основное отличие рассмотренных методов.

При *коммутации сообщений* используется другой информационный блок. Размер сообщения определяется его содержанием. Сообщение, так же как и пакет данных, может временно храниться на транзитных устройствах сети при ее перегрузках. Коммутация сообщений обычно используется при некритичности к возможным задержкам в передаче.

10.1.7. Линии связи

В качестве линий связи могут использоваться разные физические среды. В настоящее время при построении вычислительных сетей используется радиосвязь в различных диапазонах и кабельная связь.

В локальных сетях широкое распространение получила именно кабельная связь. Кабель представляет собой проводник (или несколько проводников), помещенный в изолирующие материалы. Используются три вида кабелей: коаксиальный кабель, витая пара и волоконно-оптический кабель.



Рис. 10.5. Коаксиальный кабель

В центре коаксиального кабеля (рис. 10.5) находится жесткий медный проводник, окруженный толстым слоем изоляционного материала. Второй проводник сделан в виде оплетки поверх изоляции. Весь кабель помещается во внешнюю пластиковую оболочку. Коаксиальные кабели выпускаются разных типов. Различают «толстый» и «тонкий» коаксиальные кабели, название которых происходит от величины диаметра центрального проводника. «Толстый» коаксиальный кабель обладает лучшими характеристиками, чем «тонкий». Коаксиальный кабель используется в локальных сетях с пропускной способностью до 10 Мбит/с.

Кабель витой пары состоит из двух проводников, заключенных в оболочку. Для уменьшения влияния помех проводники скручиваются с определенным шагом

скрутки. Существуют неэкранированная (рис. 10.6) и экранированная (рис. 10.7) витые пары, которые различаются наличием дополнительного защитного экранного слоя. Кабель витой пары очень чувствителен к электромагнитным помехам. Кабели витой пары классифицируются по производительности передачи данных по уровням. Для построения вычислительных сетей используются кабели, начиная с уровня 3. Кабель уровня 3 предназначен для сетей с производительностью до 16 Мбит/с, уровня 4 — до 20 Мбит/с, уровня 5 — до 100 Мбит/с.

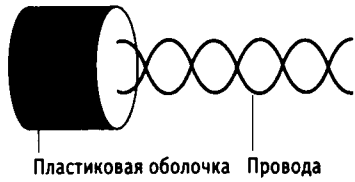


Рис. 10.6. Витая пара

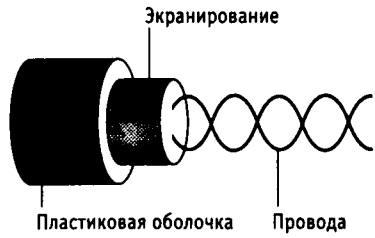


Рис. 10.7. Витая пара с экранированием

В волоконно-оптическом кабеле для передачи данных используются световые импульсы. Сердечник такого кабеля (рис. 10.8) изготовлен из стекла или пластика. Сердечник окружен слоем отражателя, который направляет световые импульсы вдоль кабеля. Такой кабель не подвержен воздействию электромагнитных помех. Производительность волоконно-оптического кабеля составляет до 10 Гбит/с. Различают одномодовые и многомодовые кабели. В одномодовом кабеле (рис. 10.9) используется очень тонкий центральный проводник. Луч света почти не отражается от внешнего отражателя. В многомодовом кабеле (рис. 10.10) используется более толстый центральный проводник, в котором одновременно существует несколько световых лучей с разными углами преломления-модами.



Рис. 10.8. Волоконно-оптический кабель



Рис. 10.9. Одномодовый волоконно-оптический кабель

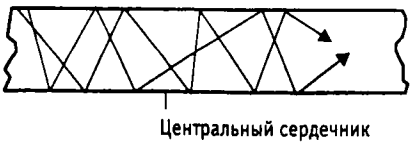


Рис. 10.10. Многомодовый волоконно-оптический кабель

10.2. Базовые технологии локальных сетей

В проектировании локальных сетей основная роль отводится протоколам физического и канального уровней модели OSI. Специфика локальных сетей, в которых используется разделяемая среда передачи данных, нашла свое отражение в разделении канального уровня на два подуровня: логической передачи данных (Logical Link Control), уровень LLC, и управления доступом к сети (Media Access Control), уровень MAC.

Уровень MAC обеспечивает корректное использование общей среды передачи данных, когда по определенному алгоритму любой узел получает возможность передачи своего кадра данных. В современных вычислительных сетях имеют распространение несколько протоколов уровня MAC: Ethernet, Fast Ethernet, Gigabit Ethernet, 100VG-AniLAN, Token Ring, FDDI.

Уровень LLC организует передачу кадров данных с различной степенью надежности.

10.2.1. Технология Ethernet

Фирменный сетевой стандарт *Ethernet* был разработан фирмой Xerox в 1975 году. В 1980 году фирмы DEC, Intel, Xerox разработали стандарт Ethernet DIX на основе коаксиального кабеля. Эта последняя версия фирменного стандарта послужила основой *стандарта IEEE 802.3*. Стандарт IEEE 802.3 имеет модификации, которые различаются типом используемой физической среды:

- 10Base-5 — «толстый» коаксиальный кабель диаметром 0,5". Позволяет создавать сегменты до 500 м;
- 10Base-2 (рис. 10.11) — «тонкий» коаксиальный кабель диаметром 0,25". Позволяет создавать сегменты до 185 м;
- 10Base-T (рис. 10.12) — неэкранированная витая пара. Позволяет создавать сеть по топологии «звезда». Расстояние от концентратора до конечного узла — не более 100 м;

- 10Base-F — волоконно-оптический кабель. Топология аналогична топологии предыдущей модификации. Расстояние от концентратора до конечного узла — от 1000 до 2000 м в зависимости от варианта спецификации.

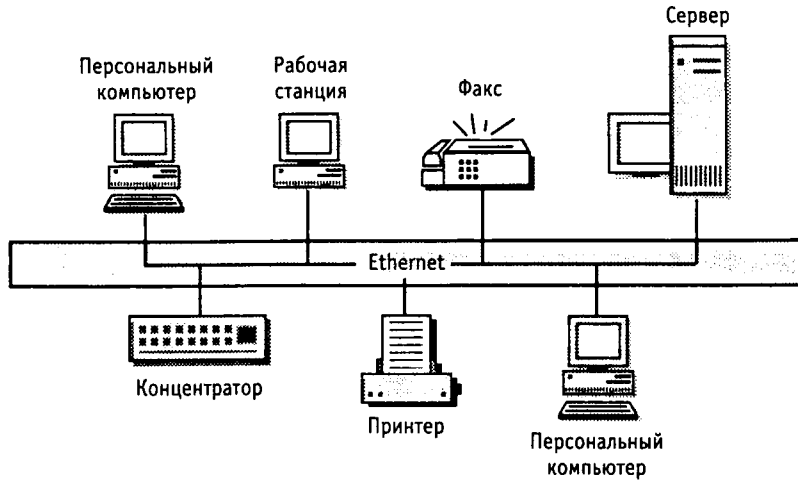


Рис. 10.11. Сеть Ethernet 10Base-2

Локальные сети, построенные по этому стандарту, обеспечивают пропускную способность до 10 Мбит/с. Используемая топология — общая шина, «звезда» и смешанные структуры.

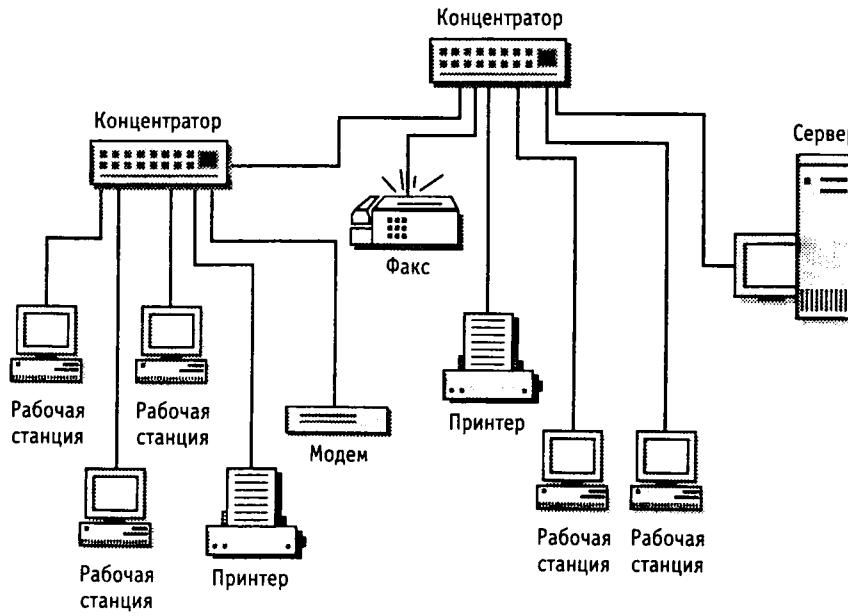


Рис. 10.12. Сеть Ethernet 10Base-T

Метод коллективного доступа

В стандарте 802.3, включая Fast Ethernet и Gigabit Ethernet, в качестве метода доступа к среде передачи данных используется метод коллективного доступа с опознаванием несущей и обнаружением коллизий (carrier-sense-miltiply-access with collision detection, CSMA/CD), *метод CSMA/CD*.

Этот метод используется в сетях, где все компьютеры имеют непосредственный доступ к общей шине и могут немедленно получить данные, которые посылаются любым компьютером. Простота этого метода позволила ему получить широкое распространение.

Данные передаются кадрами. Каждый кадр снабжается преамбулой (8 байт), которая позволяет синхронизировать работу приемника и передатчика. В заголовках кадра указывается адрес узла-получателя, который позволяет узлу-получателю распознать, что предаваемый кадр предназначен ему, и адрес узла-отправителя для отправки сообщения, подтверждающего факт получения кадра. Минимальная длина кадра — 64 байта, максимальная — 1518 байт. Минимальная длина кадра является одним из параметров, определяющих диаметр сети или максимальную длину сегмента сети. Чем меньше кадр, тем меньше диаметр сети.

Передача кадра возможна, когда никакой другой узел сети не передает свой кадр. Стандарт Ethernet не позволяет одновременную передачу/прием более одного кадра. На практике в сетях Ethernet возможны ситуации, когда два узла пытаются передать свои кадры. В таких случаях происходит искажение передаваемых данных, потому что методы стандарта Ethernet не позволяет выделять сигналы одного узла из общего сигнала и возникает так называемая *коллизия*[1]. Передающий узел, обнаруживший коллизию, прекращает передачу кадра, делает паузу случайной длины и повторяет попытку захвата передающей среды и передачи кадра. После 16 попыток передачи кадра кадр отбрасывается.

При увеличении количества коллизий, когда передающая среда заполняется повторными кадрами, реальная пропускная способность сети резко уменьшается. В этом случае необходимо уменьшить трафик сети любыми доступными методами (уменьшение количества узлов сети, использование приложений с меньшими затратами сетевых ресурсов, реструктуризация сети).

10.2.2. Технологии Fast Ethernet и 100VG-AnyLAN

Развитие локальных сетей, появление новых более быстрых компьютеров привело к необходимости совершенствования стандарта Ethernet с целью увеличения пропускной способности сети до 100 Мбит/с.

В 1995 году было принято два стандарта: IEEE 802.3u, *Fast Ethernet*, основанный на разработке некоммерческого объединения Fast Ethernet Alliance во главе компаний SynOptics, 3Com и др., и IEEE 802.3z, *100VG-AnyLAN*, основанный на разработке компаний Hewlett-Packard, AT&T и др.

Технология Fast Ethernet использует метод доступа CSMA/CD, такой же, как в технологии Ethernet, что обеспечивает согласованность технологий.

---

[1] Заметим, что сам термин «коллизия» не должен вводить читателя в заблуждение — в данном контексте коллизия является вполне обычным явлением.

Отличия Fast Ethernet от Ethernet наблюдаются только на физическом уровне. На канальном уровне изменений нет. Установлены три спецификации для физического уровня:

- 100Base-TX — неэкранированная витая пара (две пары в кабеле) и экранированная витая пара;
- 100Base-T4 — неэкранированная витая пара (четыре пары в кабеле);
- 100Base-FX — многомодовое оптоволокно (два волокна в кабеле).

Диаметр сети сократился до 200 метров, что связано с увеличением скорости передачи данных в 10 раз. Стандарты TX и FX могут работать как в *полудуплексном режиме* (передача ведется в двух направлениях, но попеременно во времени), так и в *полнодуплексном режиме* (передача ведется одновременно в двух направлениях) за счет использования двух витых пар или двух оптических волокон.

В технология 100VG-AnyLAN (рис. 10.13) для доступа к разделяемой среде используется приоритетный доступ по требованию *Demand Priority*, который может поддерживать кадры технологий как Ethernet, так и Token Ring. Сеть состоит из корневого концентратора и присоединенных к нему узлов и других концентраторов. Концентратор играет роль арбитра доступа к сети. Узел запрашивает у концентратора разрешение на передачу кадра. Если сеть свободна, концентратор отправляет кадр узлу назначения. Если сеть занята, то запрос ставится в очередь. Единовременно концентратор может хранить только один кадр.

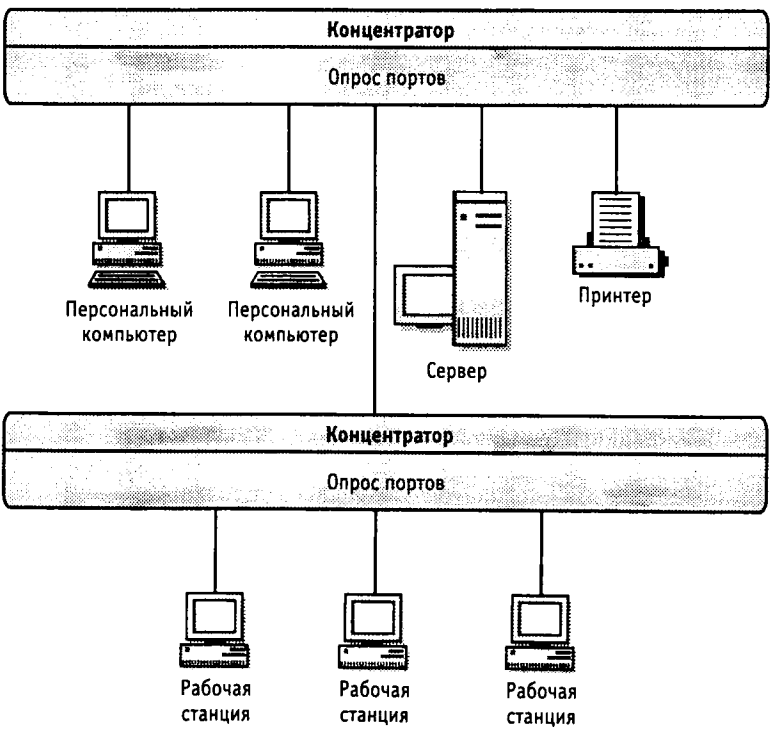


Рис. 10.13. Сеть 100VG-AnyLAN

Для определения занятости сети корневой концентратор циклически выполняет опрос портов. Если порт связан с другим концентратором, то опрос задерживается до момента завершения опроса портов концентратора нижнего уровня (возможны три уровня). Решение о предоставлении доступа принимается на основе опроса всеми концентраторами всех своих портов.

В сети поддерживаются два уровня приоритетов: высокий уровень соответствует кадрам данных, которые чувствительны к временным задержкам передачи, и низкий уровень — обычные данные. Узел, имеющий низкий приоритет, может получить высокий приоритет, если достаточно долго не может получить доступ к сети.

Концентратор отправляет кадр только узлу назначения, а не всем узлам. Адрес каждого узла сети ставится в соответствие с портом концентратора при установлении физического контакта и запоминается в специальной таблице концентратора.

10.2.3. Технология Gigabit Ethernet

Стандарт IEEE 802.3z *Gigabit Ethernet* был принят в 1998 году на основе согласованных усилий группы компаний, образовавших объединение Gigabit Ethernet Alliance. В качестве варианта физического уровня был принят физический уровень технологии Fiber Channel. Разработчики стандарта максимально сохранили преемственность предыдущих стандартов Ethernet: сохраняются все форматы кадров, полудуплексная и полнодуплексная версии протоколов, поддерживаются коаксиальный кабель, витая пара категории 5, волоконно-оптический кабель.

Поддержка полудуплексного режима метода доступа CSMA/CD сокращает диаметр сети до 25 м. Для увеличения диаметра сети до 200 м разработчики изменили размер минимального кадра с 64 до 512 байт. Для сокращения накладных расходов по передаче длинных кадров стандарт разрешает передавать несколько кадров подряд, не дополняя их до 512 байт и не передавая доступ к среде другому узлу.

10.2.4. Технология Token Ring

Технология *Token Ring* была разработана в 1984 году фирмой IBM, на основе которой в 1985 году был принят стандарт IEEE 802.5. Сеть Token Ring (рис. 10.14) так же, как и Ethernet, предполагает использование разделяемой среды передачи данных, которая образуется объединением всех узлов в кольцо. Каждый узел сети имеет связь с предшествующим и последующим узлом. Кадр данных передается от узла к узлу по кольцу в одном направлении. Такой режим называется *симплексным*.

Для доступа к среде передачи данных применяется *маркерный метод*. При использовании этого метода право доступа передается с помощью специального кадра, называемого *маркером*. Все узлы ретранслируют кадры как повторители. Маркер передается от узла к узлу. Каждый узел, получив маркер, определяет наличие у него данных для передачи. Если данных нет, то узел передает маркер следующему узлу. Если данные есть, то маркер изымается из сети. Узел посылает свой кадр данных по кольцу. Каждый кадр снабжается как адресом получателя, так и адресом отправителя. Узел, получивший кадр с адресом получателя, совпадающим с

его собственным адресом, копирует данные, вставляет в кадр признак подтверждения приема и оправляет кадр дальше. Получив обратно посланный кадр с подтверждением получения, узел-отправитель отправляет в сеть новую копию маркера для передачи доступа к сети. Время доступа к сети ограничивается временем удержания маркера, в течение которого узел может послать несколько кадров данных и после чего узел обязан передать маркер в сеть. Этот алгоритм маркерного доступа используется в сетях Token Ring, которые работают на скорости 4 Мбит/с.

В сетях Token Ring, которые работают на скорости 16 Мбит/с, используется алгоритм раннего освобождения маркера, суть которого заключается в отправке маркера сразу после передачи кадра данных. В этом случае по сети одновременно могут продвигаться кадры нескольких станций.

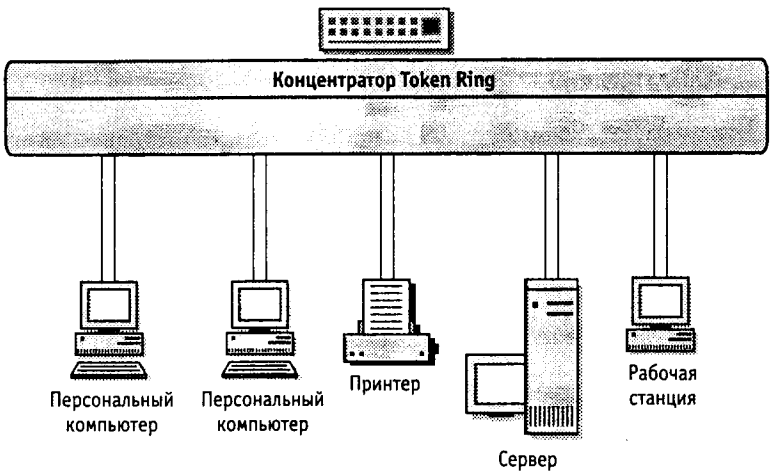


Рис. 10.14. Сеть Token Ring

Сеть Token Ring поддерживает 8 приоритетов кадров. Назначение приоритета кадра осуществляет передающий узел на верхнем, например, прикладном уровне. Маркер также имеет приоритет текущего значения. Узел получает доступ к сети только в том случае, если приоритет кадра данных для передачи в сеть не меньше приоритета маркера. В противном случае маркер передается следующему узлу. При этом передающий узел записывает приоритет своего кадра данных в резерв маркера, если записываемый приоритет не больше, чем приоритет в резерве. В случае реализации доступа к сети приоритет из резерва становится текущим значением приоритета новой копии маркера.

Контроль за работой сети, за наличием маркера в сети осуществляет активный монитор. Функции активного монитора выполняет один из узлов сети. В частности, в случае отсутствия маркера в сети в течение достаточно длительного времени активный монитор генерирует новую копию маркера. Одновременно в сети не может быть больше одной копии маркера.

Стандарт Token Ring поддерживает экранированную и неэкранированную витую пару, оптоволоконный кабель. Максимальная длина кольца 4000 м. Максимальное количество узлов 260.

Компания IBM предложила новую технологию High-Speed Token Ring, которая поддерживает скорости 100 и 155 Мбит/с и сохраняет основные особенности технологии Token Ring.

10.2.5. Технология FDDI

Технология FDDI (Fiber Distributed Data Interface) разрабатывается институтом ANSI, начиная с 80-х годов. В этой технологии в качестве физической среды передачи данных впервые предлагается оптоволоконный кабель. Имеется возможность использования неэкранированной витой пары.

Сеть FDDI (рис. 10.15) состоит из двух колец для повышения отказоустойчивости. Данные передаются по первичному кольцу сети в одном направлении, по вторичному кольцу — в противоположном. В обычном режиме используется только первичное кольцо. В случае отказа, когда часть первичного кольца не может передавать данные (например, обрыв кабеля или отказ узла), происходит процесс сворачивания колец, при котором первичное кольцо объединяется с вторичным, образуя новое кольцо. При множественных отказах сеть распадается на несколько колец. В стандарте FDDI предусмотрено одновременное подключение узлов к первичному и вторичному кольцам и подключение только к первичному кольцу. Первое называется двойным подключением, а второе — одиночным. При обрыве узла с двойным подключением происходит автоматическое сворачивание колец. Сеть продолжает нормально функционировать. При обрыве узла с одиночным подключением сеть продолжает работать, но узел будет отрезан от сети.

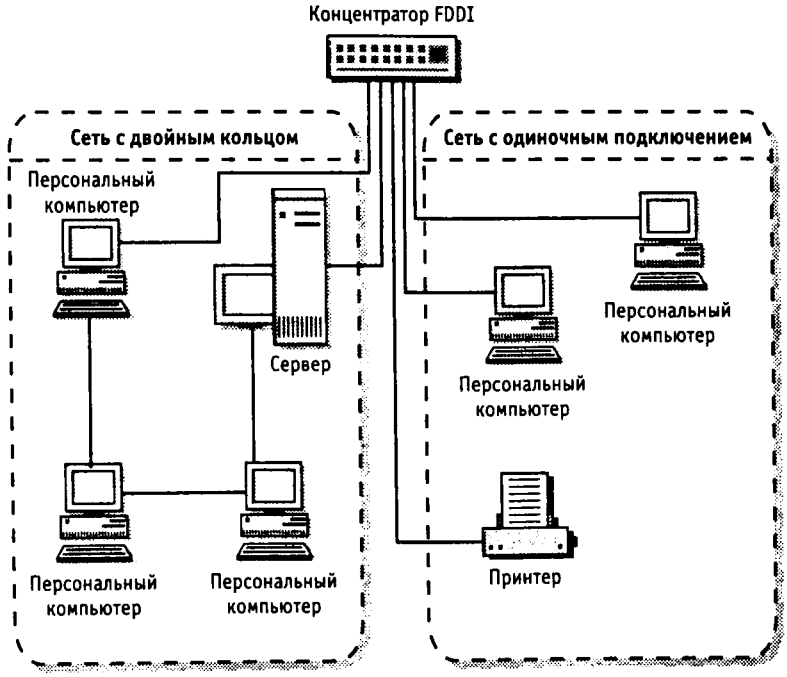


Рис. 10.15. Сеть FDDI

Кольца сети FDDI являются разделяемой средой передачи данных, для доступа к которой применяется маркерный метод, аналогичный используемому в сетях Token Ring. Различия в некоторых деталях. Время удержания маркера является переменной величиной и зависит от степени загрузки сети. При небольшой загрузке сети время удержания маркера больше, при большой загрузке — уменьшается.

Сеть FDDI поддерживает скорость 100 Мбит/с. Диаметр сети — 100 км. Максимальное количество узлов — 500. Однако стоимость реализации данной технологии значительна, поэтому область применения стандарта FDDI — магистрали сетей и крупные сети.

10.3. Построение локальных сетей

10.3.1. Структуризация локальных сетей средствами канального уровня

Принцип использования разделяемой среды передачи данных позволяет строить эффективные вычислительные сети. Простота используемых протоколов обеспечивает невысокую стоимость построения сети. Пропускная способность достигает 100 Мбит/с и даже 1000 Мбит/с. Благодаря простой топологии такие сети обладают хорошей расширяемостью. Но масштабируемость таких сетей оставляет желать лучшего. С увеличением количества подключаемых компьютеров полезная пропускная способность сети резко уменьшается. Существует ограничение на количество подключаемых узлов, превышение которого резко уменьшает эффективность работы сети.

Для построения больших вычислительных сетей, объединяющих сотни и тысячи узлов, используются *коммутаторы* (мосты). В последнее время производители пошли по пути объединения функций этих устройств в одном изделии. При небольшом увеличении стоимости потребитель получает больше возможностей при построении сетей. В отличие от концентраторов (хабов, повторителей), которые передают полученный кадр данных на все свои порты, коммутатор анализирует адрес, помещенный в кадр данных, и передает его только на порт, к которому подключен сегмент с узлом-получателем. Принцип работы коммутатора заключается в буферизации всех кадров, приходящих на его порты, и построении *адресной таблицы*, в которой адрес любого узла сети ставится в соответствие с портом коммутатора. Информация адресной таблицы динамически обновляется. При получении первого кадра от нового узла адрес этого узла записывается в адресную таблицу. Если в течение определенного времени от узла кадры не поступают, то запись в таблице об этом узле помечается как недействительная. Если адрес узла-получателя отсутствует в адресной таблице, то кадр пересылается на все порты, кроме порта, на который он пришел. Если узел-получатель есть в адресной таблице, то кадр передается на соответствующий порт. Операция продвижения кадра на другой порт сопровождается получением доступа к сегменту, подключенному к этому порту. Если узел-отправитель и узел-получатель находятся в одном сегменте, то кадр просто удаляется из буфера.

Сегментация сети уменьшает нагрузку на отдельные сегменты сети и увеличивает полезную пропускную способность как отдельных сегментов, так и сети в целом.

Сегментация, или разбиение сети на отдельные подсети, позволяет улучшить и другие свойства сети. При помощи мостов и коммутаторов можно управлять трафиком между подсетями. Улучшается управляемость сети, повышается безопасность данных.

10.3.2. Построение локальных сетей средствами сетевого уровня

Применение средств только канального уровня с использованием таких устройств, как концентраторы и коммутаторы, для построения больших вычислительных сетей имеет существенные ограничения и недостатки.

Использование в качестве адреса назначения плоского MAC-адреса, связанного с сетевым адаптером, позволяет использовать лишь одноуровневую систему адресации, которая обладает недостаточной гибкостью при построении больших систем.

Управление потоком информации при помощи коммутаторов на основе лишь данных кадра достаточно сложно.

Сегментация вычислительной сети при помощи коммутаторов изолирует подсети в недостаточной мере в смысле защиты от широковещательных сообщений (широковещательных штормов).

В разных сегментах вычислительной сети возможно использование разных базовых технологий. Но должна совпадать система адресации узлов. Например, Ethernet, Token Ring, FDDI, Fast Ethenet используют адресацию на основе MAC-адресов, а X.25, ATM, Frame Relay – другие системы адресации.

Другим ограничением применения коммутаторов для построения больших вычислительных сетей является обязательность отсутствия петель. Коммутатор может нормально работать только в случае, если между узлами есть только один маршрут. В то же время наличие избыточных связей является одним из основных способов для повышения надежности и улучшения трафика современной вычислительной сети.

Для построения больших неоднородных вычислительных сетей используются средства сетевого уровня, основным используемым устройством которого является маршрутизатор.

На сетевом уровне сеть называется *составной сетью*, состоящей из сетей, объединенных маршрутизаторами. Каждая сеть имеет свой номер. Каждый узел имеет номер внутри сети. Сетевой адрес узла состоит из этих номеров. Таким образом, в составных сетях каждый узел, кроме локального адреса, имеет дополнительный сетевой адрес.

Данные, передаваемые на сетевом уровне, называются *пакетами*. Заголовки пакета, которыми снабжаются данные, содержат необходимую информацию о сети с узлом-получателем, и их формат не зависит от технологий, используемых внутри участвующих сетей.

Пакет данных, который должен быть передан между узлами разных сетей, может транзитно передаваться через несколько сетей и связывающие их маршрутизаторы. Последовательность маршрутизаторов, которые проходит пакет данных, называется *маршрутом*. В общем случае в составной сети для передачи данных может и должно быть несколько маршрутов. Выбор оптимального маршрута является одной из основных задач маршрутизатора.

Выбор маршрута следования пакета осуществляется на основе маршрутной информации, которая собирается маршрутизатором и записывается в специальную *таблицу маршрутизации*. В таблице маршрутизации номеру сети назначения ставится в соответствие сетевой адрес выходного порта маршрутизатора и сетевой адрес следующего маршрутизатора. В таблицу маршрутизации помещается и другая информация, позволяющая выбрать оптимальный по какому-либо критерию маршрут, например количество транзитных маршрутизаторов, которые необходимо пройти пакету, прежде чем пакет попадет в нужную сеть.

Таблица маршрутизации создается автоматически путем периодического обмена между маршрутизаторами маршрутной информацией.

10.4. Стандартные стеки коммуникационных протоколов

10.4.1. Стек протоколов TCP/IP

Стек TCP/IP был разработан для работы в разнородных сетях. Этот стек получил большое распространение после своей реализации в ОС UNIX. В настоящее время он используется как для связи компьютеров в сети Интернет, так и в корпоративных сетях. На физическом и канальном уровнях стек поддерживает все базовые технологии локальных сетей.

Стек TCP/IP имеет четыре уровня:

- прикладной уровень;
- основной уровень;
- уровень межсетевых взаимодействий;
- уровень сетевых интерфейсов.

На *прикладном уровне* собраны службы, предоставляющие сетевой сервис пользовательским приложениям. Список служб постоянно изменяется. Среди основных служб можно выделить следующие: Telnet, FTP, TFTP, DNS, SNMP, HTTP. Прикладной уровень объединяет функции прикладного уровня и уровня представления модели OSI.

Основной уровень обеспечивает надежность доставки пакетов данных, их целостность и порядок доставки. На этом уровне передаваемые данные разбиваются на сегменты (пакеты) и передаются на нижний уровень. После передачи пакеты собираются и данные передаются на прикладной уровень. Основным протоколом

этого уровня является протокол управления передачей, *протокол TCP* (Transmission Control Protocol). Основной уровень выполняет функции сеансового и транспортного уровня модели OSI. Другое название этого уровня — транспортный.

Основным протоколом *уровня межсетевых взаимодействий* является протокол передачи пакетов, *протокол IP* (Internet Protocol). Этот протокол обеспечивает передачу пакетов данных в составной сети, где есть не только локальные, но и глобальные связи. На этом уровне для сбора маршрутной информации используются протоколы маршрутизации, такие как *протокол RIP* (Routing Internet Protocol), *протокол OSPF* (Open Shortest Path First). Этот уровень соответствует сетевому уровню модели OSI.

Уровень сетевых интерфейсов определяет интерфейсы к сетям, которые интегрируются в составную сеть. Включаемая сеть может использовать любую технологию. Для каждой сетевой технологии разрабатываются протоколы, по которым IP-пакеты сетевого уровня встраиваются в кадры используемых технологий. Этот уровень соответствует физическому и канальному уровням модели OSI.

Соответствие уровней стека TCP/IP уровням модели OSI достаточно условно. Это связано с тем, что стек TCP/IP был разработан более 20 лет назад.

10.4.2. Стек протоколов IPX/SPX

Стек IPX/SPX разработан фирмой Novell для сетевой операционной системы NetWare в начале 80-х годов. Протоколы разрабатывались с учетом применения в локальных сетях с небольшими сетевыми ресурсами, но с хорошими физическими коммуникационными средствами. Поэтому эти протоколы не очень хорошо работали в составных сетях с глобальными низкоскоростными связями. Но Novell постоянно вносит изменения в работу протоколов для лучшей адаптации к работе в крупных составных сетях. Пропускная способность современных глобальных сетей постоянно возрастает. Последние замечания позволяют стеку протоколов IPX/SPX конкурировать с другими стеками при создании корпоративных сетей.

Основным протоколом стека является протокол межсетевого обмена пакетами, *протокол IPX* (Internetwork Packet Exchange), который относится к сетевому уровню модели OSI. *Протокол SPX* (Sequenced Packet Exchange) обеспечивает надежность передачи данных. Использование протокола SPX протоколами верхнего уровня необязательно. Основной протокол верхнего уровня — это *протокол NCP* (NetWare Core Protocol), который обеспечивает работу основных служб сетевой операционной системы Novell NetWare и объединяет функции всех уровней от транспортного до прикладного модели OSI. Протокол SAP (Service Advertising Protocol) используется при широковещательных сообщениях, когда узел передает информацию о сетевых службах, которые он может предоставить. Здесь же указывается его сетевой адрес.

Протокол IPX использует адрес, состоящий из номера сети, номера узла и номера сокета. Номер сети назначается на сервере администратором. Номером узла является его аппаратный адрес (MAC-адрес сетевого адаптера или порта маршрутизатора). Номер сокета идентифицирует приложение, которое передает сообщение по протоколу IPX.

Конфигурирование клиентской части сети не требует значительных затрат. Адрес узла считывается автоматически из сетевого адаптера узла. Адрес сети узнается из серверных объявлений SAP. Для выяснения адреса маршрутизатора при старте системы клиента посылается сообщение SAP, в ответ на которое имеющиеся маршрутизаторы сообщают свои адреса.

10.5. Системное программное обеспечение вычислительных сетей

Сетевые операционные системы выполняют функции уровней, начиная с сетевого и выше, согласно модели OSI.

В общем случае сетевая ОС, установленная на отдельном компьютере, имеет определенную структуру и состоит из следующих частей.

Во-первых, любая сетевая ОС должна иметь средства управления локальными ресурсами компьютера, выполнять функции локальных ОС: функции распределения оперативной памяти между процессами, планирования и диспетчеризации процессов, управления процессорами в мультипроцессорных машинах, управления периферийными устройствами — и другие функции.

Во-вторых, сетевая ОС должна быть способной предоставлять собственные ресурсы и определенные услуги в общее пользование, то есть иметь серверную часть или сервер. К функциям сервера можно отнести, например, блокировку файлов и записей, что необходимо для их совместного использования, ведение справочников имен сетевых ресурсов, обработку запросов удаленного доступа к собственной файловой системе и базе данных, управление очередями запросов удаленных пользователей к своим периферийным устройствам.

В-третьих, сетевая ОС должна иметь клиентскую часть, или редиректор, которая обеспечивает доступ к удаленным ресурсам и услугам и их использование. Эта часть выполняет распознавание и перенаправление в сеть запросов к удаленным ресурсам от приложений и пользователей. Клиентская часть осуществляет прием ответов от серверов.

Коммуникационные средства ОС, с помощью которых происходит обмен сообщениями в сети, обеспечивают адресацию и буферизацию сообщений, выбор маршрута передачи сообщения по сети, надежность передачи и т. п., то есть являются средствами транспортировки сообщений.

В зависимости от задач, решаемых с помощью сетевого компьютера, на него устанавливается определенный набор модулей сетевой ОС. Сетевые компьютеры делятся на серверы и клиенты.

Клиент обычно не предоставляет пользователям сети свои локальные ресурсы. На компьютер-клиент не устанавливаются какие-либо сетевые службы. Это обычный персональный компьютер с установленной на нем локальной ОС и урезанным набором сетевых функций. Компьютер-клиент посылает запрос в сеть для получения доступа к сетевым ресурсам или услугам и обрабатывает полученный ответ (рис. 10.16).

Компьютер-сервер обеспечивает совместный доступ к своим ресурсам, обслуживает определенный сетевой сервис. При получении запросов от клиентов сервер обрабатывает их и отправляет ответы обратно.

На практике отдельные клиенты могут выполнять некоторые серверные функции (рис. 10.17), например предоставлять совместный доступ к своим дисковым ресурсам. Но полноценные серверные функции клиент никогда не выполняет. Компьютер-сервер может выполнять клиентские функции, то есть на этом компьютере могут выполняться локальные приложения, как на обычном персональном компьютере с генерацией запросов к сетевым ресурсам. В этом случае возможна смешанная схема взаимоотношений «клиент—сервер».

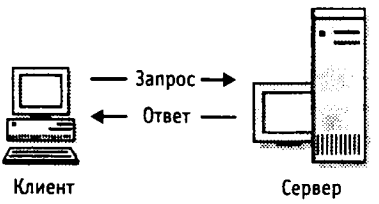


Рис. 10.16. «Чистые» взаимоотношения клиента и сервера

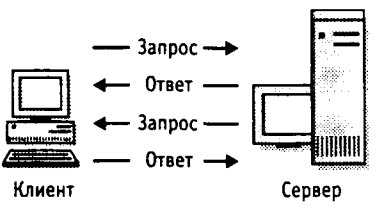


Рис. 10.17. «Смешанные» взаимоотношения клиент—сервер



Рис. 10.18. Сетевая ОС как сетевая оболочка

Сетевые модули ОС могут быть реализованы разными способами. В первом случае — в виде сетевой оболочки (рис. 10.18), которая является надстройкой над локальной ОС. Сетевая оболочка, редиректор, определяет запросы к удаленным ресурсам и направляет их по необходимому адресу. В противном случае запрос передается локальной части ОС.

Второй вариант, когда сетевые функции встраиваются в основные модули ОС, является более эффективным и производительным. По этому варианту построены почти все современные серверные варианты сетевых ОС и некоторые клиентские варианты.

В зависимости от распределения функций между компьютерами в сети можно выделить одноранговые сетевые ОС и сетевые ОС с выделенным сервером.

Одноранговые сетевые ОС используются для построения одноранговых сетей, где каждый компьютер может выполнять как функции клиента, так и функции сервера. К одноранговым ОС можно отнести практически все современные локальные ОС. В них, как правило, реализованы базовые сетевые функции для обеспечения сетевого взаимодействия. Основной среди этих функций является обеспечение доступа к дисковым устройствам, принтерам и другим локальным устройствам рабочих станций. Одноранговые сетевые ОС просты при инсталляции и эксплуатации. С другой стороны, они обладают низкой производительностью, имеют ограниченные возможности по обеспечению связи удаленных сегментов сети. Одноранговые сетевые ОС не обладают развитыми средствами управления сетью, не обеспечивают распределенный режим работы «клиент—сервер».

При построении сложных сетей, как правило, один или несколько компьютеров выделяют для выполнения отдельных сетевых функций. Такие сети называют сетями с выделенным сервером, а устанавливаемые на них сетевые ОС – *операционными системами с выделенным сервером*. В зависимости от исполняемых функций различают файл-серверы, принт-серверы, серверы приложений и т. д. На компьютеры-серверы устанавливаются специальные серверные варианты сетевых ОС. Такие ОС обладают мощными сетевыми возможностями. Их работа оптимизирована для выполнения возложенных на них функций. Обычно в серверах отсутствует клиентская часть, как в Novell NetWare. Но в ОС Windows NT for Server клиентская часть есть. Использование сервера в качестве локального клиентского компьютера должно быть ограничено, так как это может уменьшить производительность работы компьютера как сервера.

Сети с выделенным сервером или несколькими серверами по сравнению с одноранговыми сетями имеют ряд преимуществ, которые связаны не только с потенциальными возможностями используемых сетевых ОС, но и с применением более мощной техники серверов.

- высокая производительность сети;
- наличие развитых аппаратных и программных средств связи удаленных сегментов сети и рабочих станций;
- наличие развитых средств управления и администрирования в сети;
- распределенный режим работы «клиент—сервер».

К недостаткам можно отнести достаточную сложность как в освоении, так и в эксплуатации таких ОС и ограниченные возможности доступа к ресурсам клиентских машин.

Признанными лидерами сетевых ОС с выделенным сервером являются Windows NT и Novell NetWare.

10.5.1. Windows NT

Операционная система Windows NT разработана в двух версиях. Windows NT Workstation предназначена для установки на сетевых рабочих станциях. Эта опе-

рационная система позволяет использовать компьютер не только как клиент сети, но и как невыделенный сервер. Правда, возможности такого сервера сильно ограничены.

Для реализации полноценного сервера разработан второй вариант, Windows NT Server.

Windows NT Server может выступать как:

- файл-сервер;
- сервер печати;
- сервер приложений;
- контроллер домена;
- сервер удаленного доступа;
- сервер Интернета;
- сервер обеспечения безопасности данных;
- сервер резервирования данных;
- сервер связи;
- сервер вспомогательных служб.

Как файл-сервер Windows NT обеспечивает совместный доступ к дисковому пространству компьютера-сервера. Для организации такого доступа не требуется специальных мероприятий.

Как принт-сервер система обеспечивает доступ как к локальным, так и к сетевым принтерам. Сетевые принтеры могут быть подключены по разным протоколам. Количество принтеров не ограничено. Доступ к сетевым принтерам возможен с рабочих станций с разными локальными операционными системами. Дополнительная настройка принтеров на клиентских компьютерах не требуется, используются драйверы, установленные на сервере.

В операционную систему встроены возможность резервного копирования на магнитную ленту. Процедура резервного копирования может быть автоматизирована.

Windows NT изначально проектировалась для реализации модели «клиент—сервер», когда приложение, требующее для своего выполнения значительных вычислительных ресурсов, выполняется на мощном сервере по запросу клиентской машины. Результаты передаются на клиентскую машину. Windows NT оптимизирована для работы сервера приложений. В первую очередь к таким приложениям относятся:

- сервер баз данных Microsoft SQL Server;
- сервер управления системой Microsoft System Management Server;
- сервер информационного обмена Microsoft Exchange;
- сервер связи с мэйнфреймами SNA Server;
- сервер Интернета.

Подобные приложения разрабатываются и другими компаниями: Oracle, Informix, IBM, HP, DEC, Lotus и др.

Служба удаленного доступа используется для связи с территориально удаленными компьютерами. Набор поддерживаемых протоколов позволяет поддерживать связь в условиях разнородной сети. Windows NT поддерживает набор протоколов «точка–точка» (Point-to-Point Protocol, PPP), что позволяет осуществлять удаленный доступ через любой стандартный PPP-сервер. Windows NT поддерживает протоколы TCP/IP, IPX/SPX, NetBEUI.

Windows NT позволяет объединять разнородные сети. Поддержка набора протоколов IPX/SPX дает возможность сопряжения с сетями, обслуживаемыми ОС Novell NetWare. Поддержка протоколов TCP/IP выводит Windows NT в разряд систем, готовых к непосредственной работе в Интернете.

Система управления Windows NT имеет обширный набор функций и свойств. Она базируется на двух моделях построения сети:

- модели рабочих групп;
- модели доменов.

Модель рабочих групп имеет дело с организационной единицей *рабочая группа*, которая объединяет несколько компьютеров Windows NT. Система администрирования такой группы аналогична администрированию одного компьютера. Все выполняемые административные действия применяются только к одной рабочей станции.

Модель доменов используется для обеспечения надежной защиты и упрощения управления сетью при децентрализации баз учетных данных и разнообразных средств защиты. Организационной единицей этой модели является *домен* как совокупность компьютеров с общей базой учетных записей пользователей и единой политикой защиты. На каждого пользователя заводится только одна запись. Параметры пользователя распространяются на все компьютеры, входящие в домен. Компьютер, на котором находится база учетных записей, называется *первичным контроллером домена*. Для обеспечения надежности информация базы учетных записей пользователей дублируется на других компьютерах домена Windows NT. Эти компьютеры называются *резервными контроллерами домена*. Пользователи могут получить доступ ко всем компьютерам домена, выполнив процедуру регистрации всего лишь один раз.

Пользователи одного домена не имеют никаких прав в другом домене. Для объединения нескольких доменов назначаются *доверительные отношения* между доменами, которые позволяют получить информацию о ресурсах объединяемых доменов. Доверительные отношения обеспечивают доступ к ресурсам одного домена пользователям, имеющим учетные записи в другом домене. Доверительные отношения могут быть односторонними и двухсторонними. Односторонние доверительные отношения предоставляют права доступа пользователям одного домена к ресурсам другого. Двухсторонние доверительные отношения предоставляют пользователям обоих доменов доступ к ресурсам этих доменов.

С ростом числа доменов усложняется управление сетью. Для разных уровней организации сети Windows NT предлагает четыре типа доменной модели:

- однодоменная модель;
- модель с одним мастер-доменом;
- модель с несколькими мастер-доменами;
- полностью доверительная модель.

Сетью по однодоменной модели, где используется один домен с небольшим количеством компьютеров, ресурсов и пользователей, управляет администратор домена. Доверительных отношений нет.

Сеть по модели с одним мастер-доменом состоит из нескольких доменов, один из которых домен учетных записей пользователей, а другие — ресурсные домены. Ресурсные домены устанавливают с мастер-доменом односторонние доверительные отношения. Все пользователи мастер-домена получают доступ к ресурсам всех остальных доменов.

Модель с несколькими мастер-доменами используется для построения сетей, где есть домены с учетными записями пользователей и домены ресурсов. По необходимости организуются односторонние доверительные отношения между мастер-доменами с одной стороны и доменами ресурсов с другой. Между мастер-доменами возможны двухсторонние доверительные отношения.

Модель полностью доверительных отношений используется, когда нет деления доменов на ресурсные домены и домены учетных записей. Между доменами устанавливаются двухсторонние доверительные отношения.

10.5.2. Novell NetWare

NetWare предоставляет пользователям следующие возможности:

- поддерживает коллективное использование файлов;
- обеспечивает доступ к сетевым принтерам;
- предлагает средства для работы с электронной почтой;
- поддерживает работу СУБД различных типов;
- обеспечивает доступ к файловому серверу со стороны рабочих станций, функционирующих под управлением различных операционных систем;
- предлагает средства, позволяющие объединять удаленные сегменты сети;
- обеспечивает «прозрачность» доступа локальных и удаленных пользователей к ресурсам сети;
- предлагает средства для надежного хранения данных;
- обеспечивает защиту ресурсов сети от несанкционированного доступа;
- поддерживает динамически расширяемые многосегментные тома на нескольких дисках файлового сервера;

- предоставляет средства управления ресурсами корпоративных сетей: единый каталог сетевых ресурсов NDS;
- обеспечивает передачу и обработку данных с использованием разных протоколов: SPX/IPX, TCP/IP, NetBIOS, AppleTalk;
- поддерживает работу суперсерверов в симметричном режиме функционирования.

Файловый сервер

Сервер NetWare — это прежде всего *файловый сервер*, который обеспечивает многопользовательский доступ к общим файлам, хранящимся на его устройствах: на жестких дисках, оптических, оптомагнитных дисках, на стриммерах. Это, с одной стороны, позволяет не дублировать общие данные на рабочих станциях, а с другой стороны, обеспечивает взаимодействие пользователей через файловый сервер.

NetWare поддерживает несколько уровней системы многоуровневой защиты операций, System Fault Tolerance (SFT), обеспечивающих надежное хранение данных.

Система SFT имеет три уровня защиты.

Первый уровень защиты SFT-I включает следующие решения:

- дублирование таблиц DET и FAT;
- проверка записи на диск последующим чтением;
- динамическая переадресация блоков (Hot Fix).

Второй уровень надежности SFT-II включает следующие решения:

- зеркальное отображение дисков (Disk Mirroring);
- дублирование дисков (Disk Duplexing);
- система отслеживания транзакций TTS (Transaction Tracking System);
- использование устройств бесперебойного питания (UPS).

При зеркализации диска используются два жестких диска. Один и тот же блок данных записывается на оба диска. Считывание данных производится с основного диска. При сбое чтение автоматически осуществляется со второго диска.

Дублирование дисков — это процедура зеркалирования на дисках, подключенных к разным каналам передачи данных.

Система отслеживания транзакций TTS позволяет в случае сбоев в работе сервера вернуться к предыдущему состоянию системы.

Использование устройств бесперебойного питания позволяет нормально завершать работу сервера без потерь данных при отключении напряжения. Эта процедура осуществляется в автоматическом режиме после получения от устройства UPS соответствующего сигнала.

Третий уровень надежности SFT-III предполагает, что работают два файловых сервера, соединенных между собой при помощи специального кабеля. Один сервер функционирует в режиме нормального функционирования, второй — в режиме «горячего» резервирования. На резервный сервер передаются все изменения,

связанные с основным сервером. Резервный сервер постоянно определяет состояние основного сервера. При потере работоспособности основного сервера резервный сервер начинает выполнять функции основного до момента восстановления работоспособности основного сервера.

Сервер печати

Сервер печати NetWare позволяет сделать печатающие устройства разделяемыми, то есть доступными всем клиентам сети. Это очень удобное средство для случая, когда количество локальных принтеров меньше, чем пользовательских компьютеров. Пользователи могут получить доступ к единичному дорогостоящему оборудованию.

При использовании сетевой печати данные, направляемые на печать, помещаются в *очередь* в виде задания. Сервер печати периодически сканирует очереди и при наличии в них заданий на печать пересылает их на принтеры.

Принтеры в сетях NetWare могут подключаться тремя способами:

- к файловому серверу;
- к любой рабочей станции. В этом случае на этой станции требуется загружать необходимое программное обеспечение сетевого принтера;
- непосредственно к сетевой шине, если принтер снабжен специальной сетевой платой.

Управление принтерами включает:

- изменение приоритета очереди, обслуживаемой принтером;
- изменение списка пользователей, оповещаемых о проблемах, возникающих при работе принтера (отсутствие бумаги и т. д.);
- управление состоянием принтера: останов принтера, пауза принтера, старт принтера, сброс задания, прогон листа и др.

Управление очередью заданий на печать включает:

- управление заданиями в очереди: возможность задержки задания в очереди, возможность изменения номера обслуживания задания в очереди (то есть изменение приоритета задания в очереди);
- управление состоянием очереди: возможность постановки клиентом задания в очередь, возможность обслуживания очереди сервером печати, возможность подключения нового сервера печати к очереди.

Поддержка различных платформ операционных систем

NetWare обеспечивает возможность доступа к файловому серверу со стороны рабочих станций, функционирующих под управлением различных операционных систем, таких как DOS, OS/2, UNIX, Macintosh, Windows 9x, Windows NT и т. д. Разработаны средства взаимодействия NetWare с другими сетевыми операционными системами.

На каждой рабочей станции должно быть установлено специальное клиентское обеспечение. NetWare поддерживает работу различных файловых систем путем поддержания различных пространств имен, которые ссылаются на соответствующие FAT-записи.

Сервер электронной почты

Под управлением NetWare функционирует шлюз *электронной почты* MHS (Message Handling Service). MHS управляет сбором, маршрутизацией и доставкой сообщений через разнородные шлюзы электронной почты и линии связи. С MHS совместимы более 200 пакетов электронной почты, функционирующих на рабочих станциях.

Сервер баз данных

NetWare обеспечивает работу систем управления базами данных (СУБД), поддерживающих две технологии обработки запросов: клиент-файл и клиент-сервер[1]. В СУБД типа клиент-файл запросы прикладной программы к базе данных транслируются и выполняются на рабочей станции, на файловом сервере хранятся файлы БД, индексные и технологические файлы. В СУБД типа клиент-сервер запрос к БД передается серверу СУБД, функционирующему как совокупность NLM-модулей на файловом сервере. Здесь запрос транслируется и выполняется. На рабочую станцию обратно передаются только результаты выполнения запроса. Для СУБД этого типа характерно уменьшение сетевого трафика и увеличение нагрузочной способности сети.

10.5.3. Сетевая рабочая станция в локальных сетях

В локальной сети под управлением Novell NetWare или Windows NT могут работать рабочие станции с разными локальными ОС. На рабочей станции под управлением Windows 9x должен быть установлен соответствующий сетевой клиент, конфигурация которого может изменяться при помощи соответствующего пункта свойств сетевого окружения (рис. 10.19).

Рабочая станция в сети Windows

Для определения параметров рабочей станции в сети Windows NT используется соответствующий пункт в свойствах сетевого окружения (рис. 10.20), где указывается имя домена, на котором должна происходить регистрация, и дополнительная информация о необходимости проверки всех сетевых подключений.

Управлением сетью Windows и регистрацией занимается сетевой администратор, используя специальные утилиты администратора сервера и пользователей. Server Manager (рис. 10.21) позволяет управлять доменами, рабочими группами и компьютерами. User Manager (рис. 10.22) управляет регистрацией пользователей, правами и другими характеристиками, необходимыми для работы в сети Windows.

---

[1] Напоминаем, что более подробно данные вопросы были рассмотрены в главе 7.

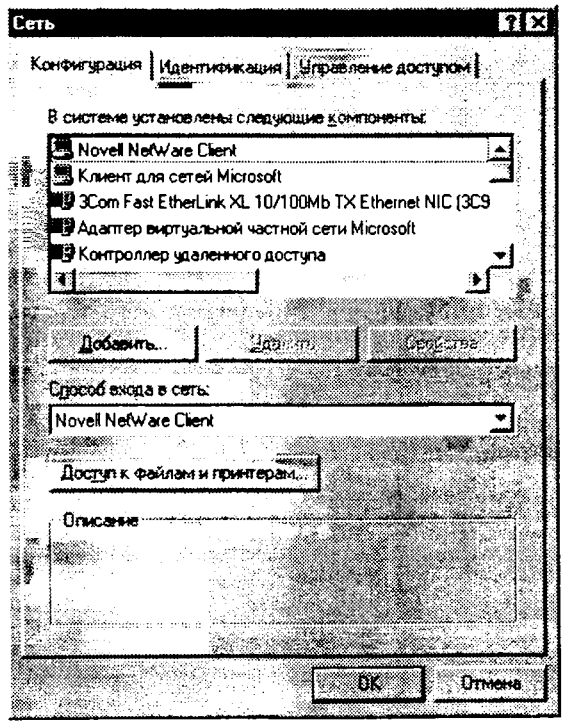


Рис. 10.19. Окно изменения настроек локальной сети

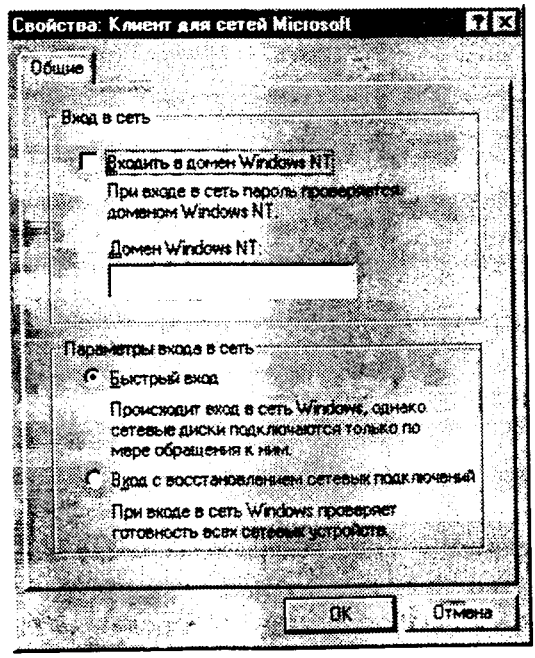


Рис. 10.20. Окно свойств клиента сетей Windows

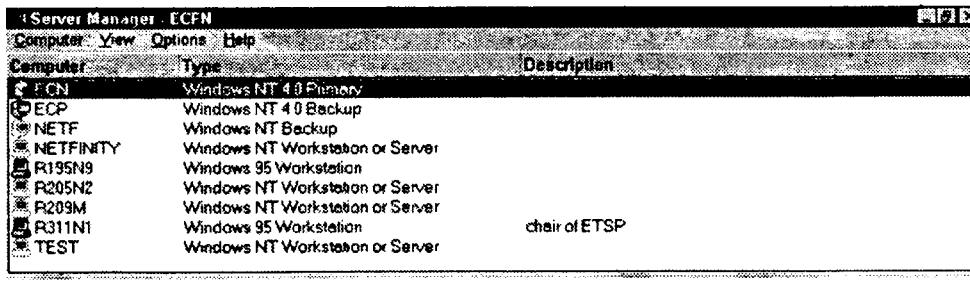


Рис. 10.21. Окно утилиты администратора сервера

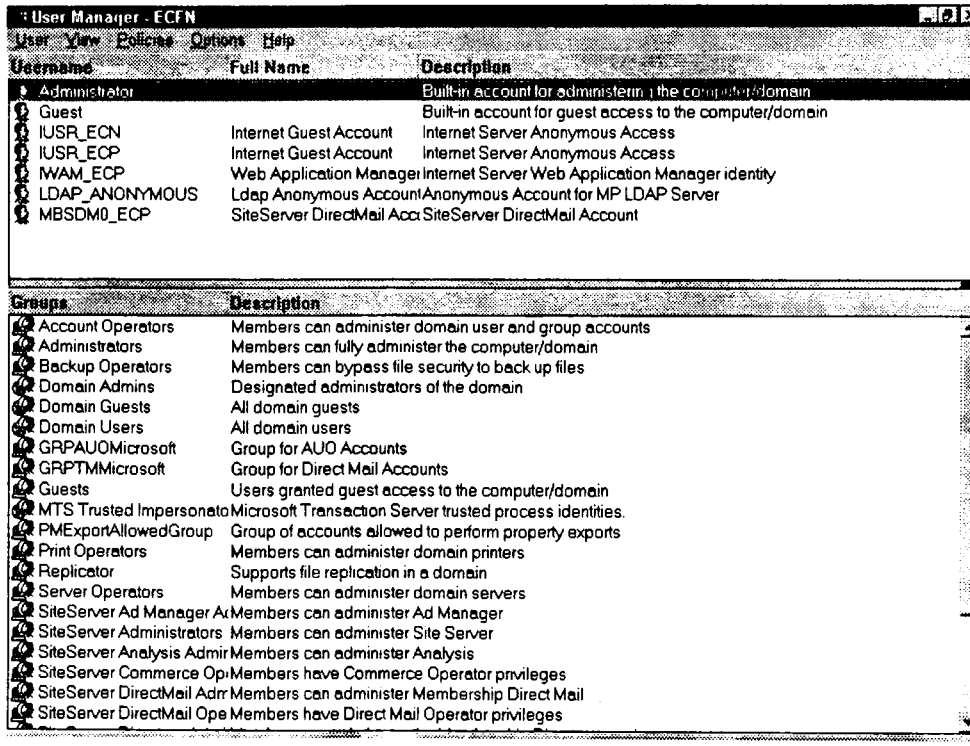


Рис. 10.22. Окно утилиты администратора пользователей

Рабочая станция в сети NetWare

Конфигурирование сетевой рабочей станции (Windows 9x) возможно при помощи настроек сетевого клиента NetWare (рис. 10.23).

Для получения доступа пользователя к локальной сети на рабочей станции необходимо зарегистрироваться. Для успешной регистрации необходимо указать имя пользователя, пароль и некоторую дополнительную информацию (рис. 10.24).

После успешной регистрации в сетевом окружении появятся дополнительные устройства. Возможно, через некоторые ярлыки на рабочем столе и в меню Пуск появится доступ к программам, расположенным на серверах локальной сети.

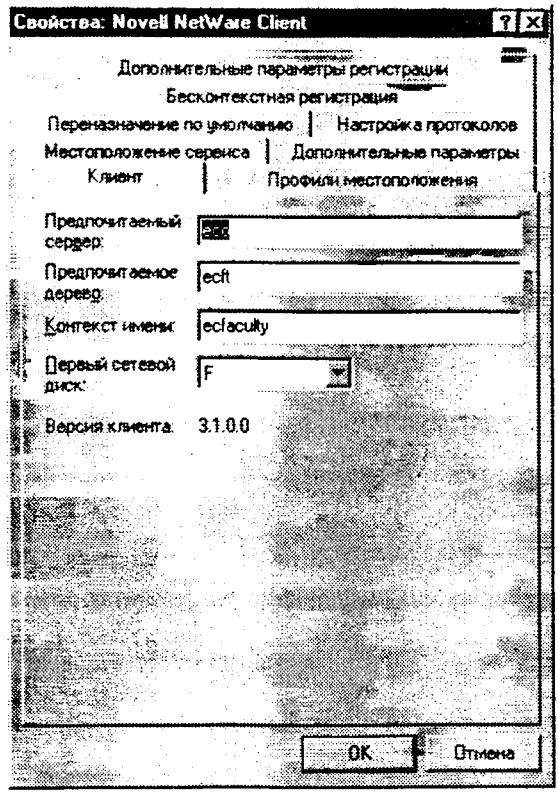


Рис. 10.23. Окно свойств клиента Netware

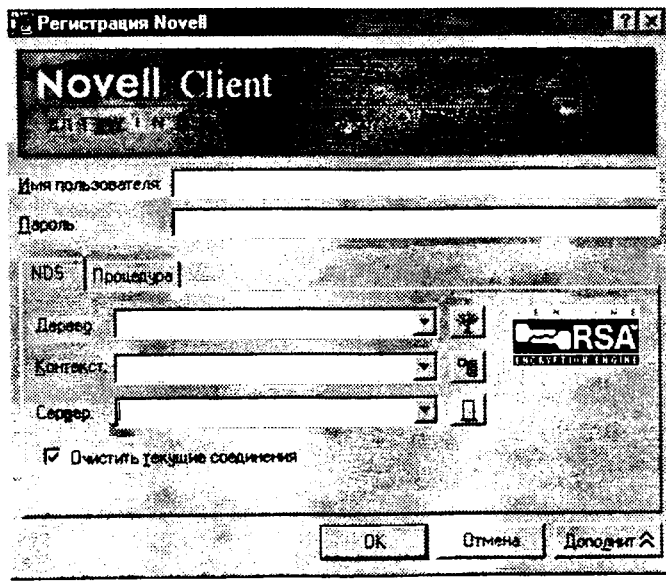


Рис. 10.24. Окно регистрации в локальной сети под управлением Novell NetWare

При установленном клиенте в правом нижнем углу есть значок **N**, через который можно получить доступ к свойствам сетевого подключения пользователя (рис. 10.25).

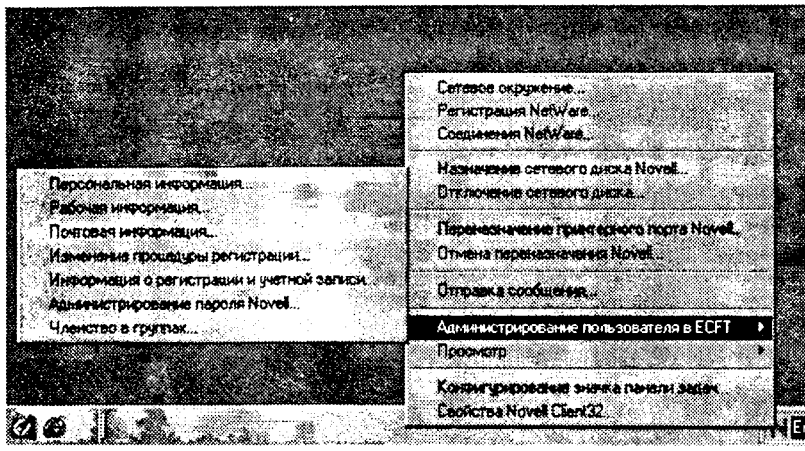


Рис. 10.25. Просмотр и изменение некоторых свойств сетевого подключения

Пользователь может ознакомиться с характеристиками своей регистрации и изменить некоторые параметры. Возможность изменения тех или параметров зависит от прав, которыми сетевой администратор наделяет пользователя.

Полный набор сервиса по изменению параметров сетевого подключения пользователя, и не только, предоставляет специальная утилита NetWare Administrator.

Ключевые понятия

- локальная сеть;
- модель OSI;
- протокол;
- интерфейс;
- сервер;
- сетевая операционная система;
- пакет данных;
- коммутация пакетов;
- стек протоколов;
- сегмент сети;
- маркерный доступ;
- маршрут;
- метод доступа.

Контрольные вопросы

1. Что такое локальная сеть?
2. Какие базовые технологии локальных сетей имеют наибольшее распространение?
3. Какие устройства используются для построения локальных сетей?
4. Какие типы линий связи используются для построения локальных сетей?
5. Какие различия в методе доступа к среде передачи данных в Ethernet и Token Ring?
6. Какие средства сетевого уровня используются при объединении локальных сетей?
7. Какие операционные системы используются при построении одноранговых локальных сетей и локальных сетей с выделенным сервером?
8. Какие функции выполняют выделенные серверы?

Литература

1. Лоренс Б. Novell NetWare 4.1 в подлиннике. СПб.: BHV, 1996.
2. Ресурсы Microsoft Windows NT Server 4.0. СПб.: BHV, 1997.
3. Сетевые средства Microsoft Windows NT Server 4.0. СПб.: BHV, 1997.
4. Олифер В. Г., Олифер Н. А. Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2000.

Источники в Интернете

Сервер Информационных Технологий: www.citforum.ru.

Глава 11. Глобальные компьютерные сети

Процессы взаимодействия вычислительных средств достаточно быстро переросли рамки отдельных фирм и организаций. Тенденции к интеграции и глобализации в современном мире получили адекватное отражение и в сфере компьютерных технологий. Совокупности вычислительных машин, объединенных коммуникационной средой, охватывающей значительные по расстоянию территории, получили название *глобальных компьютерных сетей*[1]. За последние два-три десятилетия все виды организационного, аппаратного и программного обеспечения этих сетей бурно развивались и претерпели массу метаморфоз. Среди сетей, получивших общемировую известность, могут быть названы SPRINT, некоммерческая компьютерная сеть FIDO, международная система расчетов S.W.I.F.T. Однако самой примечательной тенденцией в последние годы стало выделение из всего сообщества компьютерных сетей сети Интернет в качестве явного лидера как по размерам, так и по темпам прироста функциональных возможностей. Вопросы, связанные с возникновением, развитием, организацией и экономическими приложениями данной сети, рассматриваются в настоящей главе.

11.1. Основные принципы построения сети Интернет

Интернет — это глобальная информационная инфраструктура. Интернет является и механизмом распространения данных, и средой взаимодействия между пользователями и компьютерами вне зависимости от их географического положения.

Интернет представляет собой один из наиболее удачных примеров долгосрочных инвестиций в исследование и развитие информационных технологий. Первоначально целью создания Интернета являлось объединение компьютерных сетей различных типов. В настоящее время влияние Интернета распространяется не только на области, связанные с использованием компьютеров и телекоммуникаций, но и на общество в целом.

---

[1] Строго говоря, определение «охватывающей значительные по расстоянию территории» с учетом современного состояния глобальных сетей нуждается в уточнении, что и будет сделано в ходе последующего изложения.

В процессе создания сети Интернет были разработаны технические принципы функционирования и объединения компьютерных сетей, решена проблема управления глобальными информационными структурами, использованы новые принципы совместной работы над проектами и управления информационными потоками. И совсем недавно все это было очень успешно применено для ведения бизнеса.

11.1.1. История создания сети Интернет

Первые исследования в области соединения удаленных компьютеров были проведены в начале 60-х годов. В 1965 году компьютер, находящийся в Массачусетском технологическом институте, был подключен к компьютеру в Калифорнии по телефонной линии. Для соединения использовалась технология коммутации каналов, характерная для телефонных линий.

Технология коммутации каналов подразумевает создание непрерывной физической линии связи между двумя абонентами — канала. Соединить всех желающих абонентов друг с другом невозможно, поэтому используется коммутация, то есть возможность предоставления линии связи нескольким абонентам одновременно. Канал состоит из отдельных участков, которые соединяются между собой специальной аппаратурой — коммутаторами. Если абонент хочет установить соединение, то он обращается к ближайшему коммутатору, который, в свою очередь, по свободному каналу обращается к следующему и т. д. В конечном итоге устанавливается прямое соединение двух абонентов, и они могут обмениваться данными.

В результате эксперимента выяснилось, что коммутация каналов не подходит для создания компьютерных сетей. При использовании технологии коммутации каналов аппаратура абонентов должна работать с одинаковой скоростью, в то время как компьютеры обрабатывали данные с различной скоростью. Кроме того, неэффективно используется канал связи. Когда данные передаются, канал загружен, а когда обрабатываются — свободен. Однако физическое соединение и в том, и в другом случае сохраняется. Все это потребовало применения новой технологии передачи данных — коммутации пакетов.

При использовании этой технологии все передаваемые в сети сообщения разбиваются на небольшие части, которые называются пакетами. Каждый пакет снабжается заголовком, в котором указывается адрес назначения пакета. Коммутаторы, используя адрес, передают пакеты друг другу до тех пор, пока он не достигнет места назначения. Если какой-либо коммутатор слишком загружен и не может передать пакет в течение некоторого времени, он помещает его в очередь пакетов и передает позже.

Проект сети компьютеров на основе коммутации пакетов был разработан в Агентстве перспективных разработок Министерства обороны США (DARPA). Создаваемая сеть получила название ARPANET. В 1969 году началась реализация проекта, в сеть ARPANET были включены четыре удаленных компьютера.

Для включения в сеть большего числа компьютеров необходимо было выработать некоторый единый набор правил, определяющих способ взаимодействия узлов сети ARPANET (последовательность передачи, формат сообщений) — протокол.

В 1971–72 годах работа над единым протоколом для сети ARPANET была завершена. Этот протокол получил название Network Control Program (NCP). Создание протокола позволило начать разработку прикладных программ для использования в сети. Одной из первых таких программ стала программа электронной почты, которая позволила участникам проекта эффективнее обмениваться информацией между собой.

Возникновение Интернета базировалось на идее, предполагавшей возможность объединения нескольких независимых компьютерных сетей с различной архитектурой на основе ARPANET.

В протоколе NCP не предполагалось наличие какого-либо механизма взаимодействия с сетями другой, нежели ARPANET, архитектуры. Кроме того, объединяя сети, необходимо было учесть возможность временного нарушения связи или даже выхода части объединенной сети из строя. При этом оставшаяся часть сети должна продолжать нормально работать. Таким образом, были сформулированы основные принципы построения новой сети:

- для включения в Интернет отдельной сети не должно производиться никаких дополнительных изменений;
- пакеты в Интернете передаются на основе принципа негарантированной доставки, если пакет не смог достигнуть пункта назначения, то через короткое время он должен быть передан снова;
- для соединения сетей используются специальные устройства – маршрутизаторы, которые должны максимально упростить прохождение потока пакетов;
- не должно существовать единого, централизованного управления объединенной сетью.

Ключом к объединению сетей стал новый протокол, поддерживающий межсетевое взаимодействие. Первая версия нового протокола появилась в 1973 году. Протокол был назван TCP[1] (Transmission Control Protocol) и обеспечивал доставку сообщений в Интернете. Протокол TCP предоставляет достаточно широкий диапазон транспортных услуг.

TCP прекрасно работает при решении таких задач, как передача файлов или удаленная регистрация, но в некоторых случаях (например, при передаче речи) потери пакетов не могут быть исправлены только средствами TCP, а эти функции должны быть возложены на приложение. Этот факт привел к разделению TCP на два протокола: IP для адресации и передачи отдельных пакетов и TCP для разделения сообщений на пакеты, обеспечения целостности и восстановления потерянных пакетов. Для приложений, которые не хотели использовать TCP, был добавлен альтернативный протокол, названный UDP, который обеспечивал прямой доступ к IP. Объединенный протокол принято называть TCP/IP.

[1] Напомним, что вопросы, касающиеся организации и работы протоколов TCP/IP, уже затрагивались в главе 10.

Основной движущей силой при создании ARPANET и Интернета была необходимость обеспечить разделение ресурсов. Соединить два компьютера вместе оказалось гораздо экономичнее, чем копировать их функции. Однако передача файлов, удаленная регистрация и электронная почта произвели гораздо больший эффект, чем предполагалось в то время. Электронная почта, например, предоставила совершенно иной способ взаимодействия людей и изменила природу совместной работы над проектами.

Важно отметить, что одной из ключевых концепций Интернета было не создание одного или нескольких приложений для работы с сетевыми устройствами, а создание общей инфраструктуры, в которой могли бы существовать новые приложения.

Следующим сильным толчком к развитию Интернета было увеличение размеров сети и связанные с этим проблемы управления. Все большее количество людей использовало сеть Интернет, и все большее количество узлов поключалось к Интернету. В *протоколе IP* адрес узла представлен в виде набора цифр, разделенных точками. Более подробно его формат будет рассмотрен в 11.2.1.

Запомнить огромное множество комбинаций IP-адресов практически невозможно. Для того чтобы упростить использование сети людьми, с каждым узлом ассоциировались имена и, таким образом, не было необходимости запоминать числовые адреса. Первоначально существовало ограниченное число узлов, и соответствия адресов и имен можно было хранить в одной таблице. Появление большого количества независимо управляемых сетей привело к тому, что хранить адреса в одной таблице было невозможно, тогда появилась *система доменных имен* (DNS). DNS представляет собой масштабируемый, распределенный, иерархический механизм разрешения имен узлов и их адресов. К рассмотрению вопросов ее организации мы вернемся в 11.2.5.

Увеличение размеров Интернета потребовало изменения возможностей маршрутизаторов. Первоначально существовал единый алгоритм маршрутизации, который исполнялся всеми маршрутизаторами в Интернете. В то время как число хостов росло, возможности исходного алгоритма не могли быть адекватным образом расширены, и он был заменен иерархической моделью маршрутизации, включавшей протокол IGP, который использовался внутри регионов, и протокол EGP, который использовался для связи регионов вместе. Дизайн протоколов позволял использовать различные версии IGP в зависимости от требований к системе (стоимость, скорость конфигурации, чувствительность к сбоям, масштабируемость). Требования к маршрутизаторам определялись не только алгоритмами маршрутизации, но и размерами таблиц маршрутизации. Создание иерархической модели маршрутизации определило современную структуру сети Интернет.

В 1980 году протокол TCP/IP был принят в качестве стандарта для сети ARPANET. Переход на использование TCP/IP позволил разделить сеть ARPANET на две различные сети: MILNET, предназначенную для военных целей, и ARPANET — для использования исследовательскими и научными организациями. В 1986 году Национальный научный фонд США (NSF) начал создание собственной сети, объединяющей крупные научные суперкомпьютерные центры США. В качестве

основы для новой сети был выбран протокол TCP/IP и другие технологии, опробованные в ARPANET. В дальнейшем эта сеть, получившая название NSFNET, стала основной магистралью сети Интернет. В 1988 году было принято решение предоставлять доступ к сети NSFNET не только научным и образовательным организациям, но и коммерческим фирмам. В 1994 году финансирование основной магистрали сети Интернет было полностью передано от NSF различным государственным и коммерческим организациям.

11.1.2. Роль документации в развитии Интернета

Важным фактором в быстром росте сети Интернет стал открытый и свободный доступ к документации, в особенности к спецификациям протоколов. Обычный, традиционный способ публикации результатов исследований был слишком медленным для динамичного обмена идеями в сообществе разработчиков Интернета. Поэтому в 1969 году была создана система документов, названная RFC (Request For Comment). Задачей RFC было быстрое распространение идей и результатов в Интернете. Первоначально RFC распространялись с помощью электронной почты.

Целью RFC было создание цикла обратной связи между разработчиками, чтобы идеи и предложения, представленные в одном из RFC, давали начало другим идеям в следующих документах и т. д. Когда будет найден некоторый последовательный ряд идей, они объединяются для создания спецификации. Подобная спецификация может быть использована другими разработчиками.

Впоследствии в RFC авторы стали уделять больше внимания стандартам протоколов («официальным» спецификациям), хотя существуют документы, в которых рассматриваются различные (дублирующие) подходы или особенности того или иного протокола, а также другие вопросы. Сейчас RFC рассматриваются скорее как «зарегистрированные» в сообществе Интернет документы.

Открытый и бесплатный доступ к документам RFC способствует росту Интернета, так как позволяет использовать действующие спецификации в качестве примера для обучения и разработки новых систем. Ранние версии RFC представляли собой обращение одного автора ко всему сообществу Интернета. Сейчас документы RFC являются совместным творчеством нескольких разработчиков.

11.1.3. Организационная структура Интернета

В конце 70-х годов стало понятно, что рост размеров Интернета сопровождается ростом числа заинтересованных групп и, таким образом, необходима координация. Было сформировано несколько координирующих органов. IAB (Internet Architecture Board) занимается координацией деятельности в области развития структуры сети Интернет. IETF (Internet Engineering Task Force) состоит из нескольких рабочих групп, разрабатывающих и утверждающих стандарты для сети Интернет. В настоящее время в IETF существует 75 рабочих групп, изучающих различные аспекты развития Интернета.

Впоследствии основным органом, осуществляющим информационную поддержку и регулирование в Интернете, стал Internet Society (ISOC). ISOC является общественной организацией, базирующейся на взносах и пожертвованиях спонсоров. ISOC проводит ежегодные конференции, выпускает информационные материалы, поддерживает информационные серверы.

11.1.4. Современная структура сети Интернет

В настоящее время основу сети Интернет составляют высокоскоростные *магистральные сети*. Независимые сети подключаются к магистральной сети через точки сетевого доступа NAP (Network Access Point). Независимые сети рассматриваются как *автономные системы*, то есть каждая из них имеет собственное административное управление и собственные протоколы маршрутизации. Изменение протоколов маршрутизации внутри автономной системы не влияет на работу остальных систем. Деление сети Интернет на автономные системы позволяет распределить информацию о топологии всей сети и существенно упростить маршрутизацию.

Автономная система должна состоять не менее чем из 32 меньших по размеру сетей. Обычно в качестве автономных систем выступают крупные, независимые, национальные сети. Примерами подобных сетей являются сеть EUNet, охватывающая страны центральной Европы, сеть RUNet, объединяющая университеты России. Автономные сети могут образовывать компании, специализирующиеся на предоставлении услуг доступа в сеть Интернет, — *провайдеры*. Такими провайдерами, например, являются компания UUNET в США и компания Relcom в России.

Внутри автономной системы данные предаются от одной сети к другой, пока не достигнут точки сопряжения с другой автономной системой. Обмен данными возможен только в том случае, если между автономными системами существуют соглашения о предоставлении транзита. По этой причине для пользователей разных автономных систем время доступа к одному и тому же ресурсу может существенно различаться.

Сети, включенные в автономные системы, представляют собой региональные сети, сети университетов, исследовательских центров и коммерческих фирм, а также сети более мелких региональных провайдеров.

Важным параметром, определяющим качество работы в сети Интернет, является скорость доступа к ресурсам сети. Скорость доступа определяется пропускной способностью канала связи внутри автономной системы и между автономными системами. Для модемного соединения, которое используют большинство домашних пользователей персональных компьютеров, пропускная способность канала невелика — от 19,2 до 57,6 Кбит/с; для выделенных телефонных линий, часто используемых для подключения к сети Интернет небольших локальных компьютерных сетей, — от 64 Кбит/с до 2 Мбит/с; для спутниковых и оптоволоконных каналов связи, которые в основном используются для создания автономных сетей, — от 2 Мбит/с и выше.

11.2. Основные протоколы сети Интернет

В основном в сети Интернет используется семейство протоколов TCP/IP. Рассмотрим более подробно структуру протоколов TCP/IP с точки зрения модели OSI[1].

На канальном и физическом уровнях модели OSI TCP/IP поддерживает многие из существующих стандартов, определяющих среду передачи данных. Это могут быть, например, технологии Ethernet и FDDI для локальных компьютерных сетей или X.25 и ISDN для организации крупных территориальных сетей. На этом уровне также могут использоваться протоколы PPP и SLIP, предназначенные для установления соединения с использованием аналоговых линий связи.

Основой семейства протоколов TCP/IP является сетевой уровень, представленный протоколом IP, а также различными протоколами маршрутизации. Этот уровень предоставляет адресное пространство, обеспечивает перемещение пакетов в сети, а также управляет их маршрутизацией.

Размеры пакета, параметры передачи, контроль целостности осуществляются на транспортном уровне протоколом TCP. Протокол UDP работает на том же уровне, но применяется в том случае, когда требования к надежности передачи данных менее жесткие.

Прикладной уровень объединяет все службы, которые система предоставляет пользователю. К наиболее важным прикладным протоколам относятся протокол удаленного управления telnet, протокол передачи файлов FTP, протокол передачи гипертекста HTTP, протоколы для работы с электронной почтой: SMTP, POP, IMAP и MIME. На этом уровне работает система доменных имен DNS, отвечающая за преобразование числовых IP-адресов в имена. Кроме того, следует отметить протокол SNMP, предназначенный для управления сетевыми устройствами.

11.2.1. Адресация в сети Интернет

Каждый компьютер, включенный в сеть Интернет, имеет уникальный *IP-адрес*, на основании которого протокол IP передает пакеты в сети. IP-адрес состоит из четырех байтов и записывается в виде четырех десятичных чисел, разделенных точками, например:

194.85.120.66

IP-адрес состоит из двух логических частей: номера сети и номера узла в сети. Если сеть, в которую включен компьютер пользователя, является частью Интернета, то номер сети выдает специальное подразделение Интернета — InterNIC (Internet Network Information Center) или его представители. Номер узла определяет администратор сети.

В зависимости от того, какое количество байтов в IP-адресе выделяется для номера сети и номера узла, выделяют несколько классов IP-адресов.

---

[1] Более подробно структура модели OSI была рассмотрена в главе 10.

Если номер сети занимает один байт, а номер узла три байта, то такой адрес относится к *классу A*. Количество узлов в сети класса A может достигать 2^{24}, или 16 777 216. Номер сети класса A меняется в диапазоне от 1.0.0.0 до 126.0.0.0.

Если под номер сети и номер узла отводится по два байта, то адрес принадлежит к *классу B*. Количество возможных узлов в сети класса B составляет 2^{16}, или 65 536 узлов. Номер сети класса B меняется от 128.0.0.0 до 191.255.0.0.

Если под номер сети отводится три байта, то адрес принадлежит к *классу C*. Количество узлов в сети класса C ограничено 2^8, или 256. Номер сети меняется от 192.0.1.0 до 223.255.255.0.

Например, в IP-адресе 194.85.120.66, 0.0.0.66 — это номер узла в сети класса C с номером 194.85.120.0.

Существует несколько специальных IP-адресов. Так, например, адрес 127.0.0.1 определяет локальную машину пользователя и используется для тестирования различных программ. При этом данные по сети не передаются.

11.2.2. Протокол IP

Протокол IP представляет собой основу протоколов TCP/IP. Протокол IP относится к типу протоколов без установления соединения, то есть — никакой управляющей информации кроме той, что содержится в самом IP-пакете, по сети не передается. Кроме того, протокол IP не гарантирует надежной доставки сообщений.

Поток данных протокол IP разбивает на определенные части — дейтаграммы и рассматривает каждую дейтаграмму как независимую единицу, не имеющую связи с другими дейтаграммами. *Дейтаграмма* — общее название единицы данных, которыми оперируют протоколы без установления соединения. Основной задачей протокола IP является передача дейтаграмм между сетями. Часто дейтаграммы, передаваемые с помощью протокола IP, называют IP-пакетами.

11.2.3. Протокол TCP/IP

Так как протокол IP не гарантирует надежную доставку сообщений, эту задачу решает протокол TCP. В отличие от протокола IP, протокол TCP устанавливает логическое соединение между взаимодействующими процессами. Перед передачей данных посылается запрос на начало сеанса передачи, получателем посылается подтверждение.

Надежность протокола TCP заключается в том, что источник данных повторяет их посылку в том случае, если не получит в определенный промежуток времени от адресата подтверждения их успешного получения. Части, на которые протокол TCP разбивает поток данных, принято называть *сегментами*.

Каждый сегмент предваряется заголовком. В заголовке сегмента существует поле контрольной суммы. Если при пересылке данные повреждены, то по контрольной сумме протокол TCP может это определить. Поврежденный сегмент уничтожается, а источнику ничего не посылается. Если данные не были повреждены, то они

пропускаются на сборку сообщения приложения, а источнику отправляется подтверждение.

Для транспортировки сегментов протокол TCP использует протокол IP. Перед отправкой протокол TCP помещает сегменты в оболочку IP-пакета.

11.2.4. Порты и соединения

Задача протокола TCP заключается в передаче данных между любыми прикладными процессами, выполняющимися на компьютерах в сети. На каждом компьютере может выполняться одновременно несколько процессов. Для того чтобы доставить сообщение определенному процессу, необходимо каким-то образом идентифицировать его среди других. Идентификатор процесса носит название *номера порта*. Номер порта и IP-адрес компьютера однозначно определяют процесс, работающий в сети. Набор этих параметров называется *сокет*.

За некоторыми процессами номера портов закреплены. Так, например, порт 21 закреплен за службой удаленного доступа к файлам FTP, порт 23 — за службой удаленного управления telnet.

Для организации надежной передачи данных предусматривается установление логического соединения между прикладными процессами, которое определяется парой сокетов взаимодействующих процессов. В процессе соединения осуществляется подтверждение правильности приема сообщений и при необходимости выполняется повторная передача.

11.2.5. Система доменных имен DNS

Человеку крайне неудобно использовать числовые IP-адреса, поэтому логичным представляется создание механизма, позволяющего ставить в соответствие IP-адресам символьные имена. В сети Интернет для этой цели используется система доменных имен (DNS), которая имеет иерархическую структуру. Младшая часть доменного имени соответствует конечному узлу сети. Составные части отделяются друг от друга точкой. Например, mail.econ.pu.ru. У одного узла может быть несколько имен.

Совокупность имен, у которых несколько старших частей доменного имени совпадают, называется *доменом*. Например, имена mail.econ.pu.ru и www.econ.pu.ru принадлежат домену econ.pu.ru.

Самым главным является корневой домен. Далее следуют домены первого, второго и третьего уровней. Корневой домен управляется InterNIC. Домены первого уровня назначаются для каждой страны, при этом принято использовать трехбуквенные и двухбуквенные аббревиатуры. Так, например, для России домен первого уровня — ru, для США — us. Кроме того, несколько имен доменов первого уровня закреплено для различных типов организаций:

- com — коммерческие организации (например, ibm.com);
- edu — образовательные организации (например, spb.edu);

- gov — правительственные организации (например, loc.gov);
- org — некоммерческие организации (например, w3.org);
- net — организации, поддерживающие сети (например, ripn.net);

Для каждого имени домена создается свой *DNS-сервер*, который хранит базу данных соответствий IP-адресов и доменных имен, расположенных в данном домене, а также содержит ссылки на DNS-серверы доменов нижнего уровня. Таким образом, для того чтобы получить адрес компьютера по его доменному имени, приложению достаточно обратиться к DNS-серверу корневого домена, а тот, в свою очередь, перешлет запрос DNS-серверу домена нижнего уровня. Благодаря такой организации системы доменных имен нагрузка по разрешению имен равномерно распределяется среди DNS-серверов.

11.3. Электронная почта

Система *электронной почты* позволяет доставить сообщение на любой компьютер, включенный в сеть Интернет. Сообщение может содержать текст или файл практически любого формата — графику, музыку и т. д.

Все пользователи электронной почты имеют уникальные адреса. Адрес пользователя зарегистрирован в определенном домене Интернета. С каждым доменом связан почтовый сервер, управляющий адресами пользователей.

Пользователь набирает текст письма в специальной программе, которая называется почтовым клиентом. Эта программа, в зависимости от возможностей, позволяет создавать и редактировать новые письма, обрабатывать пришедшие, хранить и систематизировать переписку и т. д.

Почтовый клиент помещает письмо в «почтовый ящик» пользователя, расположенный на почтовом сервере. Сервер, в свою очередь, передает письмо на почтовый сервер адресата.

11.3.1. Адрес электронной почты

В Интернете принята система адресов, которая базируется на доменном адресе машины, подключенной к сети. Адрес пользователя состоит из двух частей, разделенных символом «@»:

Например: `Jones@Registry.org`

В данном случае `Jones` — это имя пользователя. А `Registry.org` — адрес, доменное имя почтового сервера.

11.3.2. Формат сообщения электронной почты

Почтовое сообщение состоит из трех частей: конверта, заголовка и тела сообщения. Пользователь видит только заголовок и тело сообщения. Конверт используется только программами доставки. Заголовок всегда находится перед телом сообщения и отделен от него пустой строкой. RFC-822 регламентирует содержание

заголовка сообщения. Заголовок состоит из полей. Поля состоят из имени поля и содержания поля. Имя поля отделено от содержания символом ":". Минимально необходимыми являются поля Date, From, cc или To, например:

```
Date: 26 Aug 76 1429 EDT
From: Jones@Registry.org
```

или

```
Date: 26 Aug 76 1429 EDT
From: Jones@Registry.org
To: Smith@Registry.org
```

Поле Date определяет дату отправки сообщения, поле From — отправителя, а поля cc и To — получателя или нескольких получателей. Чаще заголовок содержит дополнительные поля:

```
Date: 26 Aug 76 1429 EDT
From: George Jones
Sender: Secy@SHOST
To: Smith@Registry.org
Message — ID: <4231.629.XYzi - What@Registry.org>
```

В данном случае поле Sender указывает, что George Jones не является автором сообщения. Он только переслал сообщение, которое получил из Secy@SHOST. Поле Message — ID содержит уникальный идентификатор сообщения и используется программами доставки почты. Следующее сообщение демонстрирует все возможные поля заголовка:

```
Date: 27 Aug 76 0932
From: Ken Davis
Subject: Re: The Syntax in the RFC
Sender: KSecy@Other - host
Reply to: Sam.Irving@Reg.Organization
To: George Jones
cc: Important folks: Tom Softwood , "Sam Irving"@Other - Host;
Standard Distribution: /main/davis/people/standard@Other - Host
Comment: Sam is away on bisiness.
In reply to: George's message
X special action: This is a sample of user - defined field - names. Message - ID:
<4331.629.XYzi - What@Other - Host
```

Поле Subject определяет тему сообщения, Reply to — пользователя, которому отвечают, Comment — комментарий, In reply to — показывает, что сообщение относится

к типу «В ответ на Ваше сообщение, отвечающее на сообщение, отвечающее ...», X special action — поле, определенное пользователем, которое не определено в стандарте.

Следует сказать, что формат сообщения постоянно дополняется и совершенствуется. Так в RFC-1327 введены дополнительные поля для совместимости с почтой X.400. Кроме того, следует обратить внимание на поля некоторых, довольно часто встречающихся заголовков, которые не регламентированы в RFC-822. Так, первое предложение заголовка, которое начинается со слова «From», содержит UUCP — путь сообщения, по которому можно определить, через какие машины сообщение «пробиралось». Поле Received: содержит транзитные адреса почтовых серверов с датой и временем прохождения сообщения. Вся эта информация полезна при разборе трудностей с доставкой почты.

Возможности электронной почты не ограничиваются только пересылкой корреспонденции. По почте можно получить доступ ко многим ресурсам Интернета, которые используют почтовых роботов, отвечающих на запросы пользователей. Поэтому имеет смысл более детально изучить программное обеспечение, поддерживающее электронную почту. Стандарт MIME (или, в нотации Интернета, RFC-1341) предназначен для описания тела почтового сообщения Интернета. Предшественником MIME является Стандарт почтового сообщения ARPA (RFC-822). Стандарт RFC-822 был разработан для обмена текстовыми сообщениями. С момента опубликования стандарта возможности аппаратных средств и телекоммуникаций ушли далеко вперед и стало ясно, что многие типы информации, которые широко используются в сети, невозможно передать по почте без специальных преобразований. Так, в тело сообщения нельзя включить графику, аудио, видео и другие типы информации. RFC-822 не дает возможностей для передачи даже текстовой информации, которую нельзя реализовать 7-битовой кодировкой ASCII. Естественно, что при использовании RFC-822 не может быть и речи о передаче размеченного текста для отображения его различными стилями. Ограничения RFC-822 становятся еще более очевидными, когда речь заходит об обмене сообщениями в разных почтовых системах. Например, для приема/передачи сообщений из/в X.400 (новый стандарт ISO), который позволяет иметь двоичные данные в теле сообщения, ограничения старого стандарта могут стать фатальными, так как в этом случае не спасает старый испытанный способ кодировки информации процедурой uuencode, так как эти данные могут быть интерпретированы различным образом в X.400 и в программе рассылки почты в Интернете (mail-agent).

В некотором смысле стандарт MIME ортогонален стандарту RFC-822. Если последний подробно описывает в заголовке почтового сообщения текстовое тело письма и механизм его рассылки, то MIME ориентирован главным образом на описание в заголовке письма структуры тела почтового сообщения и возможности составления письма из информационных единиц различных типов. В стандарте зарезервировано несколько способов представления разнородной инфор-

мации. Для этого используются специальные поля заголовка почтового сообщения:

- поле версии MIME, которое используется для идентификации сообщения, подготовленного в новом стандарте;
- поле описания типа информации в теле сообщения, которое позволяет обеспечить правильную интерпретацию данных;
- поле типа кодировки информации в теле сообщения, указывающее на тип процедуры декодирования;
- два дополнительных поля, зарезервированных для более детального описания тела сообщения.

Стандарт MIME разработан как расширяемая спецификация, в которой подразумевается, что число типов данных будет расти по мере развития форм представления данных. При этом следует учитывать, что анархия типов (безграничное их увеличение) тоже недопустима. Каждый новый тип в обязательном порядке должен быть зарегистрирован в IANA (Internet Assigned Numbers Authority). Остановимся подробнее на форме и назначении полей, определяемых стандартом.

Поле версии MIME (MIME – Version) указывается в заголовке почтового сообщения и позволяет программе рассылки почты определить, что сообщение подготовлено в стандарте MIME. Формат поля выглядит так:

MIME – Version: 1.0

Поле версии указывается в общем заголовке почтового сообщения и относится ко всему сообщению целиком. Здесь уместно отметить, что в отличие от стандарта RFC-822 стандарт MIME позволяет перемешивать поля заголовка сообщения с телом сообщения. Поэтому все поля делятся на два класса: общие поля заголовка, которые записываются в начале почтового сообщения, и частные поля заголовка, которые относятся только к отдельным частям составного сообщения и записываются перед ними.

Поле типа содержания тела почтового сообщения (Content type) используется для описания типа данных, которые содержатся в теле почтового сообщения. Это поле сообщает программе чтения почты, какого сорта преобразования необходимы для того, чтобы сообщение правильно проинтерпретировать. Эта же информация используется и программой рассылки при кодировании/декодировании почты. Стандарт MIME определяет семь типов данных, которые можно передавать в теле письма:

- текст (text);
- смешанный тип (multipart);
- почтовое сообщение (message);
- графический образ (image);
- аудиоинформация (audio);

- фильм или видео (video);
- приложение (application).

Остановимся подробнее на каждом из типов, разрешенных стандартом MIME.

Text. Этот тип указывает на то, что в теле сообщения содержится текст. Основным подтипом типа text является plain, что соответствует так называемому планарному тексту. Понятие планарного текста появилось в связи с тем, что существует еще размеченный текст, то есть текст со встроенными в него символами управления отображением, и гипертекст, то есть текст, который можно просматривать не последовательно, а произвольно, следуя гипертекстовым ссылкам. Для обозначения размеченного текста используют подтип richtext, а для обозначения гипертекста — подтип html. Вообще говоря, это специальный вид размеченного текста, который используется для представления гипертекстовой информации в системе World Wide Web, которая получила широкое распространение в Интернете. Понятие размеченного текста требует более подробного обсуждения, так как его передача и интерпретация являются одной из причин появления стандарта MIME. Richtext определяет текст со встроенными в него специальными управляющими последовательностями, называемыми тегами, в соответствии со стандартом языка разметки документов SGML (Standard Generalized Markup Language). Теги представляют собой последовательность символов типа <строка символов>.

Разметка гипертекста строится по тому же принципу, что и в тексте типа richtext. При этом могут применяться теги, позволяющие описать гипертекстовые ссылки. К таким тегам относятся <A HREF=".....">.....</A>.

Multipart. Этот тип почтового сообщения определяет смешанный документ, который может состоять из данных разного типа. Тип multipart имеет ряд подтипов. Подтип mixed может создавать сообщения, состоящие из нескольких фрагментов, которые разделены между собой границей, задаваемой в качестве параметра подтипа.

Другим подтипом может быть подтип alternative. Данный подтип позволяет организовать просмотр почтового сообщения с возможностью выбора в зависимости от типа программы просмотра.

Подтип digest предназначен для многоцелевого почтового сообщения, когда различным частям хотят приписать более детальную информацию, чем просто тип.

Подтип parallel предназначен для составления такого почтового сообщения, части которого должны отображаться одновременно, что предполагает запуск сразу нескольких программ просмотра. Синтаксис такого сообщения аналогичен рассмотренным ранее.

Тип message предназначен для работы с обычными почтовыми сообщениями, которые, однако, не могут быть переданы по разного рода причинам, которые объясняются подтипами данного типа.

Подтип partial предназначен для передачи одного большого сообщения по частям. Атрибуты подтипа определяют идентификатор сообщения (id), номер порции (number) и общее число порций (total). Каждая часть имеет свое поле Content type. Это означает, что все сообщение может состоять из частей разных типов.

Другим подтипом является External body, который позволяет ссылаться на внешние относительно сообщения информационные источники.

Стандартным подтипом типа message является rfc822. Данный подтип определяет типы описания нетекстовой информации по стандарту RFC-822. Таких типов имеется четыре:

- image — для описания графических образов. Наиболее часто используются файлы форматов GIF и JPEG;
- audio — для описания аудиоинформации. Для воспроизведения сообщения данного типа требуется специальное оборудование;
- video — для передачи видеоизображений. Наиболее популярным является формат MPEG;
- application — для передачи данных любого другого формата. Обычно используется для передачи двоичных данных с последующим промежуточным преобразованием. Так, если на машине стоит видеокарта с 512 Кбайт памяти, а графика подготовлена в 256 цветах, то сначала ее следует преобразовать, и здесь может помочь тип application. Основной подтип данного типа — octet stream, но существуют ODA и postscript;
- поле типа кодирования почтового сообщения (Content-Transfer-Encoding)

< pclass=just>. Многие данные передаются по почте в их исходном виде. Это могут быть 7-битные символы, 8-битные символы, 64-base символы и т. п. Однако при работе в разнородных почтовых средах необходимо определить механизм их представления в стандартном виде — US — ASCII. Для этого существуют процедуры кодирования такого рода данных. Наиболее широко применяемая — uuencode. Для того чтобы при получении данные были правильно распакованы, в стандарт введено поле Content-Transfer-Encoding. Синтаксис этого поля следующий:

```
Content-Transfer-Encoding := "BASE64" / "QUOTED-PRINTABLE" / "8BIT" / "7BIT"
/ "BINARY" / x-token
```

Каждая из альтернатив применяется в подходящем случае. Альтернативы 8bit, 7bit, BINARY реально никакого преобразования не требуют, так как почта передается байтами и SMTP не делает различия между ними. Однако они введены для строгости описания типов. BASE64 обычно используется в связке с типом text/ ISO — 8859 — 1. Элемент x-token позволяет пользователю описать свою процедуру преобразования.

Дополнительные необязательные поля: как уже говорилось ранее, стандарт определяет еще два дополнительных поля: Content-ID и Content-Description. Первое поле определяет уникальный идентификатор содержания, а второе служит для комментария. Ни то, ни другое программами просмотра обычно не отображаются.

Подводя итоги обсуждения, еще раз следует отметить, что стандарт MIME позволяет расширить область применения электронной почты, обеспечить доступ к другим информационным ресурсам сети в стандартных форматах.

11.4. Система World Wide Web

В 1989 году гипертекст представлял новую многообещающую технологию, которая имела относительно большое число реализаций, с одной стороны, а с другой стороны, делались попытки построить формальные модели гипертекстовых систем, которые носили скорее описательный характер и были навеяны успехом реляционного подхода описания данных. Идея создания системы WWW заключалась в том, чтобы применить гипертекстовую модель к информационным ресурсам, распределенным в сети, и сделать это максимально простым способом. Он заложил три краеугольных камня системы из четырех существующих ныне, разработав:

- язык гипертекстовой разметки документов HTML (HyperText Markup Language);
- универсальный способ адресации ресурсов в сети URL (Universal Resource Locator);
- протокол обмена гипертекстовой информацией HTTP (HyperText Transfer Protocol).

Позже команда NCSA добавила к этим трем компонентам четвертый: универсальный интерфейс шлюзов CGI (Common Gateway Interface).

Язык программирования Java не включается в этот список намеренно, так как область применения этого языка гораздо шире, чем простое «оживление» World Wide Web.

Идея HTML — пример чрезвычайно удачного решения проблемы построения гипертекстовой системы при помощи специального средства управления отображением. На разработку языка гипертекстовой разметки существенное влияние оказали два фактора: исследования в области интерфейсов гипертекстовых систем и желание обеспечить простой и быстрый способ создания гипертекстовой базы данных, распределенной в сети.

В 1989 году активно обсуждалась проблема интерфейса гипертекстовых систем, то есть способов отображения гипертекстовой информации и навигации в гипертекстовой сети. Значение гипертекстовой технологии сравнивали со значением книгопечатания. Утверждалось, что лист бумаги и компьютерные средства отображения/воспроизведения серьезно отличаются друг от друга, и поэтому форма представления информации тоже должна отличаться. Наиболее эффективной формой организации гипертекста были признаны контекстные гипертекстовые ссылки, а кроме того, было признано деление на ссылки, ассоциированные со всем документом в целом и с отдельными его частями.

Обычно гипертекстовые системы имеют специальные программные средства построения гипертекстовых связей. Сами гипертекстовые ссылки хранятся в специальных форматах или даже составляют специальные файлы. Такой подход хорош для локальной системы, но не для распределенной на множестве различных компьютерных платформ. В HTML гипертекстовые ссылки встроены в тело документа и хранятся как его часть. Часто в системах применяют специальные форматы хранения данных для повышения эффективности доступа. В WWW-документах это

обычные ASCII-файлы, которые можно подготовить в любом текстовом редакторе. Таким образом, проблема создания гипертекстовой базы данных была решена чрезвычайно просто.

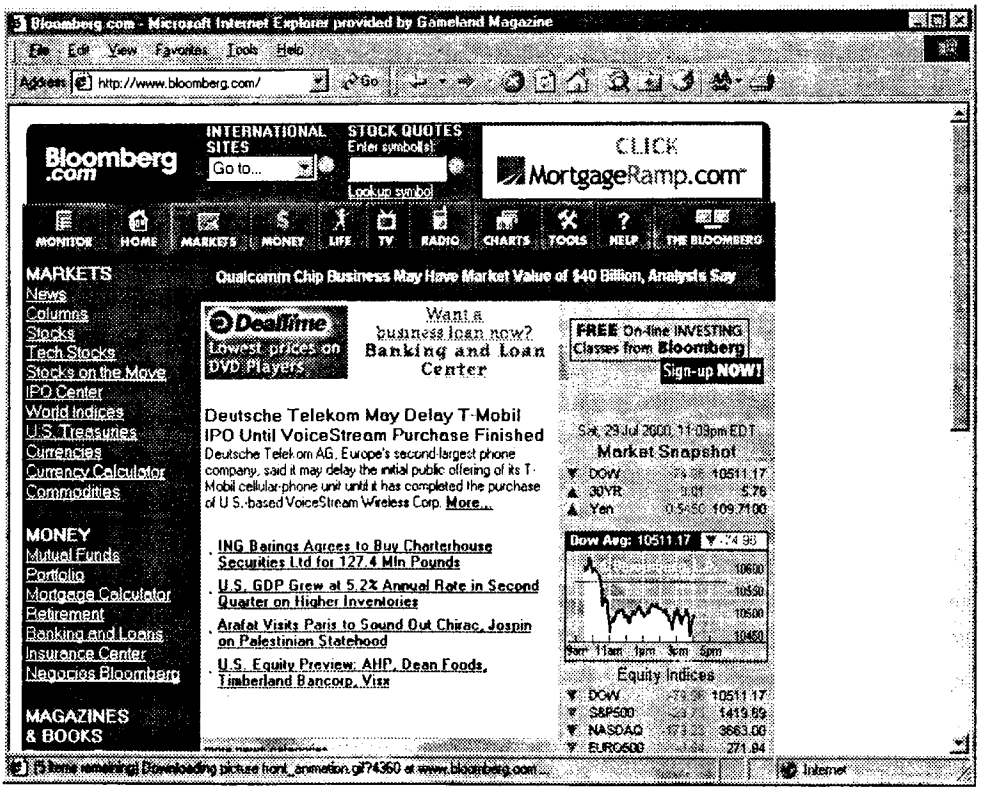


Рис. 11.1. Финансовые индикаторы и новости финансового рынка на сервере информационного агентства Bloomberg

С момента разработки первой версии языка (HTML 1.0) прошло уже пять лет. За это время произошло довольно серьезное развитие языка. Почти вдвое увеличилось число элементов разметки, оформление документов все больше приближается к оформлению качественных печатных изданий, развиваются средства описания нетекстовых информационных ресурсов и способы взаимодействия с прикладным программным обеспечением. Совершенствуется механизм разработки типовых стилей. Фактически, в настоящее время HTML развивается в сторону создания стандартного языка разработки интерфейсов как локальных, так и распределенных систем.

Вторым краеугольным камнем WWW стала универсальная форма адресации информационных ресурсов (Universal Resource Identification, URI), представляющая собой довольно стройную систему, учитывающую опыт адресации и идентификации E-mail, Gopher, WAIS, Telnet, FTP и т. п. Но реально из всего, что описано в URI, для организации баз данных в WWW требуется только Universal Resource Locator (URL). Без наличия этой спецификации вся мощь HTML оказа-

лась бы бесполезной. URL используется в гипертекстовых ссылках и обеспечивает доступ к распределенным ресурсам сети. В URL можно адресовать как другие гипертекстовые документы формата HTML, так и ресурсы E-mail, Telnet, FTP, Gopher, WAIS. Различные программы различным образом осуществляют доступ к этим ресурсам. Следует отметить, что программы обработки электронной почты в формате MIME также имеют возможность отображать документы, представленные в формате HTML. Для этой цели в MIME зарезервирован тип text/html.

Третьим в нашем списке стоит протокол обмена данными в World Wide Web — HTTP (HyperText Transfer Protocol). Данный протокол предназначен для обмена гипертекстовыми документами и учитывает специфику такого обмена. Так, в процессе взаимодействия клиент может получить новый адрес ресурса сети, запросить встроенную графику, принять и передать параметры и т. п. Управление в HTTP реализовано в виде ASCII-команд. Реально разработчик гипертекстовой базы данных сталкивается с элементами протокола только при использовании внешних программ или при доступе к внешним относительно WWW информационным ресурсам, например базам данных.

Последняя составляющая технологии WWW — это спецификация CGI (Common Gateway Interface). CGI была специально разработана для расширения возможностей WWW за счет подключения внешнего программного обеспечения. Эта технология соответствовала принципам простоты разработки, доступности и наращивания возможностей WWW. Предложенный и описанный в CGI способ подключения не требовал дополнительных библиотек и буквально ошеломлял своей простотой. Сервер взаимодействовал с программами через стандартные потоки ввода/вывода, что упрощает программирование до предела. При реализации CGI чрезвычайно важное место заняли методы доступа, описанные в HTTP. И хотя реально используются только два из них (GET и POST), опыт развития HTML показывает, что сообщество WWW ждет развития и CGI по мере усложнения задач, в которых будет использоваться WWW-технология.

11.4.1. Принципы работы сервера и клиента

Архитектура построения системы. От описания основных компонентов перейдем к архитектуре взаимодействия программного обеспечения в системе World Wide Web. WWW построена по хорошо известной схеме «клиент—сервер». Программа-клиент выполняет функции интерфейса пользователя и обеспечивает доступ практически ко всем информационным ресурсам Интернета. В этом смысле она выходит за обычные рамки работы клиента только с сервером определенного протокола. Клиент — это интерпретатор HTML. И, как типичный интерпретатор, клиент в зависимости от команд (разметки) выполняет различные функции. В круг этих функций входит не только размещение текста на экране, но и обмен информацией с сервером по мере анализа полученного HTML-текста, что наиболее наглядно происходит при отображении встроенных в текст графических изображений. При анализе URL-спецификации или по командам сервера клиент запускает дополнительные внешние программы для работы с документами в форматах, отличных от HTML, например GIF, JPEG, MPEG, Postscript и т. п.

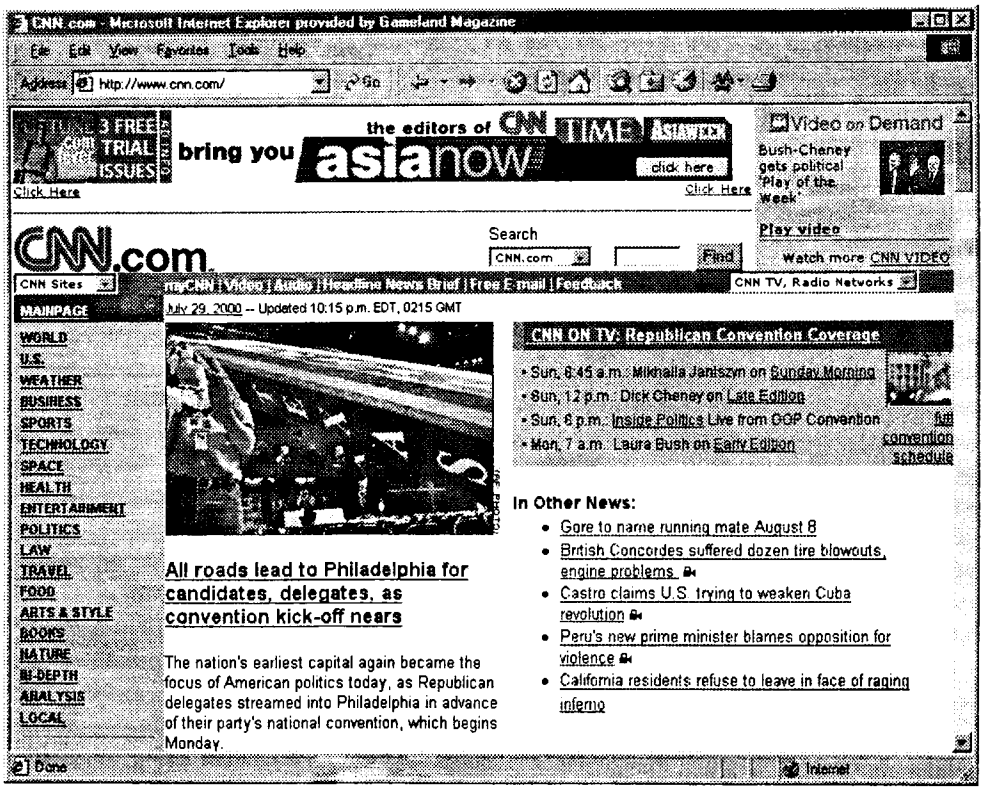


Рис. 11.2. Страница финансовых новостей на сервере компании CNN в окне броузера Microsoft Internet Explorer

Другую часть программного комплекса WWW составляет сервер протокола HTTP, базы данных документов в формате HTML, управляемые сервером, и программное обеспечение, разработанное в стандарте спецификации CGI. До самого последнего времени реально использовалось два HTTP-сервера: сервер CERN и сервер NCSA. Но в настоящее время число базовых серверов расширилось.

База данных HTML-документов — это часть файловой системы, которая содержит файлы в формате HTML и связанные с ними графику и другие ресурсы.

Среди прикладного программного обеспечения, работающего с сервером, можно выделить программы-шлюзы. *Шлюзы* — это программы, обеспечивающие взаимодействие серверов различных протоколов, например, через FTP. Другие прикладные программы принимают данные от сервера и выполняют какие-либо действия: получение текущей даты, реализацию графических ссылок, доступ к локальным базам данных или просто расчеты.

Все, что было сказано до этого момента, можно отнести к классической схеме World Wide Web. В настоящее время следует говорить о ее изменении.

К середине 1996 года произошли некоторые изменения в архитектуре сервиса World Wide Web.

Произошел возврат к модульной структуре сервера World Wide Web. Этот возврат был реализован в виде спецификации API. API — это спецификация разработки прикладных модулей, которые редактируются совместно с модулями сервера.

В дополнение к HTML активно стал применяться еще один язык разметки — VRML (Virtual Reality Modeling Language), а также язык XML.



Рис. 11.3. Новости бизнеса на сервере электронной газеты Gazeta.Ru

Изменения коснулись и клиентской части технологии. В настоящее время происходит постепенный переход от простой классической архитектуры клиент—сервер к архитектуре с сервером приложений, в роли которого выступает программа-клиент. В частности, NCSA опубликовала спецификацию CCI (Common Client Interface) для разработки приложений. Завершая обсуждение архитектуры World Wide Web, хотелось бы еще раз подчеркнуть, что ее компоненты существуют практически для всех типов компьютерных платформ и свободно доступны в сети. Любой, кто имеет доступ в Интернет, может создать свой WWW-сервер или, по крайней мере, посмотреть информацию с других серверов.

11.4.2. Универсальный адрес ресурса

Для того чтобы получить информацию из Интернета, необходимо знать адрес, по которому она расположена. Универсальный адрес ресурса (URL) — это адрес в системе World Wide Web, с помощью которого однозначно определяется любой документ. В общем случае универсальный адрес ресурса имеет следующий формат:

```
протокол://компьютер/путь
```

Протокол является набором правил, согласно которому должна происходить передача данных. Основным протоколом в системе World Wide Web является протокол HTTP — протокол передачи гипертекста, поэтому большая часть адресов начинается следующим образом:

```
http://
```

Тем не менее, могут быть использованы и другие протоколы передачи данных, например протокол передачи файлов — FTP или протокол передачи данных в формате Gopher. Тогда на первое место в универсальном адресе ресурса ставится название используемого протокола. Например:

```
ftp://
```

или

```
gopher://
```

Компьютер — это адрес сервера, с которым необходимо установить соединение. Может использоваться как IP-адрес, так и имя сервера в доменной системе имен. Например:

```
http://www.econ.pu.ru
```

или

```
ftp://194.85.120.66
```

Адреса большей части серверов в системе World Wide Web начинаются с префикса www. Этот префикс используется просто как удобное обозначение того, что на данном компьютере запущен Web-сервер.

Путь представляет собой точное указание месторасположения документа на Web-сервере. Это может быть название директории и файла, как в следующем примере:

```
http://www.econ.pu.ru/info/history/jubilee.htm
```

Если ввести в строке «адрес» броузера данный адрес, броузер установит связь с компьютером www.econ.pu.ru по протоколу HTTP и запросит у него документ с названием jubilee.htm из каталога /info/history.

Последняя часть универсального адреса ресурса может включать дополнительную информацию, которую обычно используют для того, чтобы передать Web-серверу параметры запроса пользователя в интерактивных страницах, а также

путь и имя той программы на сервере, которая этот запрос будет обрабатывать. Например:

```
http://www.econ.pu.ru/sf/cgi - bin/main.bat?object=teachers&id=1
```

Получив такой запрос, Web-сервер попытается найти программу main.bat в каталоге /sf/cgi-bin/, запустить ее и передать ей параметры object и id с соответствующими значениями.

В современных версиях броузеров нет необходимости указывать имя протокола в начале каждого адреса ресурса. Если имя протокола не указано, то броузер попытается самостоятельно определить, какой протокол необходимо использовать.

Если не указано имя файла, а только каталог, в котором он должен находиться, то пользователю будет передан файл, который администратор Web-сервера определил как файл, передаваемый по умолчанию. Обычно таковым является файл с названием index.htm (index.html) или default.htm (default.html). Если в каталоге нет файла по умолчанию, то будет выдано сообщение об ошибке.

Для указания документов на одном и том же сервере в HTML-документах часто используется сокращенное обозначение, называемое относительным адресом. Перед отправкой запроса на Web-сервер броузер преобразует относительный адрес в полный. Например, если документ по адресу http://www.econ.pu.ru/info/index.htm содержит ссылку на документ history/jublilee.htm, то броузер преобразует эту ссылку в http://www.econ.pu.ru/info/history/jubilee.htm.

11.4.3. Язык разметки гипертекста (HTML)

Большая часть документов в системе World Wide Web хранится в формате HTML. HTML – это язык гипертекстовой разметки, используемый для кодирования документов. Язык HTML представляет собой набор команд, в соответствии с которыми броузер отображает содержимое документа, команды HTML не отображаются. В языке HTML реализован механизм гипертекстовых ссылок, который обеспечивает связь одного документа с другими. Эти документы могут находиться на том же сервере, что и страница, с которой на них делается ссылка, а могут быть размещены на другом сервере.

Команды в тексте HTML-документа называются тегами.

HTML-тег может содержать список атрибутов. Текст тега заключается в угловые скобки (< и >). В простейшем случае тег представляет собой имя, заключенное в угловые скобки, например <TITLE> или <B>. Для более сложных тегов характерно наличие атрибутов, которые могут иметь конкретные значения, определяющие функции тега. Например:

```
<P ALIGN="JUSTIFY">
```

В данном случае <P> – это тег, ALIGN – один из его атрибутов, а "JUSTIFY" – значение этого атрибута. (Данный тег определяет начало абзаца, атрибут ALIGN определяет способ выравнивания текста в абзаце, значение "JUSTIFY" задает выравнивание по ширине.)

Регистр символов в именах тегов не учитывается. Теги <P> и <p> будут интерпретироваться одинаково. Однако в некоторых случаях важно использовать определенный регистр символов в значениях атрибутов.

Большинство HTML-тегов имеют парный закрывающий тег. Текст, к которому применяется форматирование, заключается между открывающим и закрывающим тегами. Закрывающий тег имеет имя, идентичное открывающему тегу, но перед именем закрывающего тега ставится косая черта. Например, чтобы отобразить текст курсивом, его нужно заключить в теги <I> и </I>:

```
<I>Этот текст будет отображен курсивом</I>
```

11.4.4. Протокол передачи гипертекста (HTTP)

Протокол передачи гипертекста (HTTP) — это стандартный протокол для передачи документов между серверами и броузерами в системе World Wide Web. Протокол HTTP позволяет установить соединение между клиентом и сервером. Соединение сохраняется только на время обработки сервером запросов клиента. Запрос клиента и ответ сервера образуют так называемую транзакцию.

Обмен данными по протоколу HTTP происходит следующим образом. Клиент устанавливает соединение с сервером по указанному номеру порта. Если в качестве клиента выступает броузер, то номер порта указывается в URL-запросе. Если номер не указан, то по умолчанию используется порт 80. Затем клиент посылает запрос на документ, указывая HTTP-команду, адрес документа и номер версии HTTP. Например:

```
GET /index.html HTTP/1.0
```

Команды HTTP-клиента принято называть методами. Метод сообщает серверу о цели запроса. В данном примере используется метод GET, который запрашивает файл index.html, расположенный в корневом каталоге сервера, используя протокол HTTP версии 1.0. Другими часто используемыми методами являются методы HEAD и POST. Метод HEAD аналогичен GET, но запрашивает не содержимое файла, а информацию о нем. Метод POST позволяет разместить файл на сервере.

Кроме того, клиент может послать информацию, называемую заголовком, чтобы сообщить серверу дополнительную информацию о себе. В качестве такой информации может выступать имя и номер версии клиента, информация о типах данных, которые предпочтительны для клиента, и др. Например:

```
User - Agent: Mozilla/4.6 [en] (Win98;I)
Accept: image/gif, image/x-xbitmap, image/jpeg, */*
```

Заголовок завершается пустой строкой.

11.4.5. Поиск информации в Интернете

Количество информации, размещенной в Интернете, постоянно увеличивается. В связи с этим возникает проблема поиска данных.

В Интернете существует два типа поисковых систем — *классификаторы* и *поисковые машины*. В зависимости от информации, которую необходимо найти, удобнее воспользоваться поисковой системой того или иного типа.

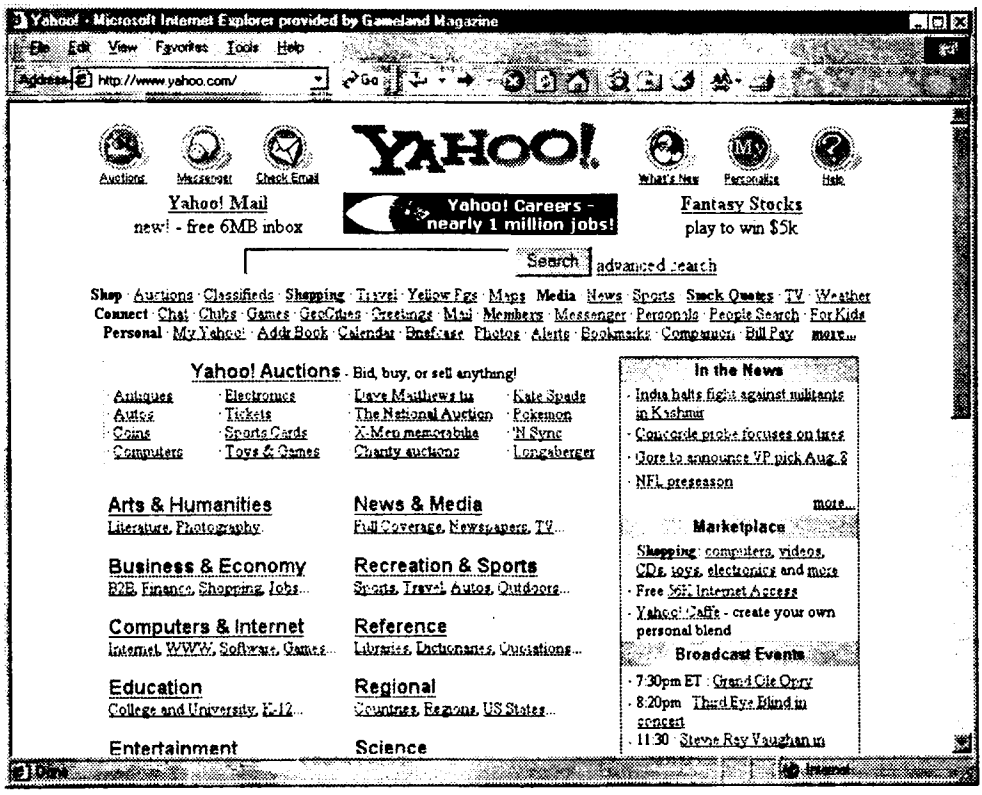


Рис. 11.4. Домашняя страница одного из самых первых классификаторов Yahoo!

Классификаторы хранят упорядоченные списки ссылок на Web-узлы. Обычно каждой ссылке сопутствует краткое описание. Списки упорядочены по тематическим разделам на поисковом сервере. Эти списки образуют иерархическую древовидную структуру. Спускаясь по дереву каталогов, можно последовательно ограничить область поиска и, в конечном итоге, получить список ссылок на Web-узлы, связанные с той темой, которая интересна пользователю.

Использовать классификаторы удобно в том случае, когда тема для поиска достаточно общая. Например, «Вузы Санкт-Петербурга» или «Авиационная промышленность». В первом случае искомый ресурс, скорее всего, будет размещен в разделе «Образование: Вузы: Вузы Санкт-Петербурга», а во втором — в разделе Промышленность: Машиностроение: Авиационная промышленность.

При использовании для поиска информации классификаторов необходимо достаточно четко представлять, к какой категории эта информация относится. Кроме того, классификация ресурсов проводится людьми, поэтому часто бывает достаточно субъективна.

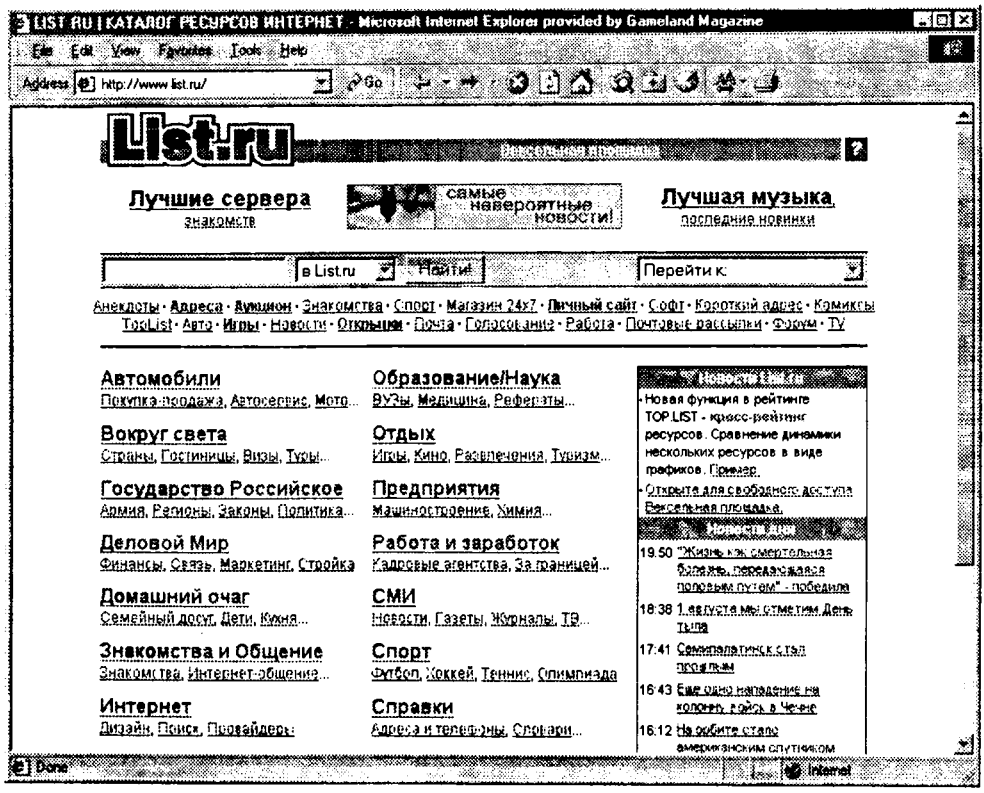


Рис. 11.5. Популярный российский классификатор List.ru

Этими недостатками не обладает другой тип поисковых систем — *поисковые машины*. Поисковые машины просматривают страницы, размещенные в Интернете, и составляют индексы используемых слов. Пользователь вводит искомое слово, набор слов или логическое выражение, и по его запросу поисковая машина выдает список ссылок на страницы, в которых это слово используется.

Одним из основных элементов поисковых машин является *индексатор* (иногда используется термин «паук») — программный модуль, периодически сканирующий Интернет для сбора данных о состоянии информационных ресурсов.

Эти данные используются для формирования и обновления *индекса* — массива данных поисковой машины, служащего для поиска адреса информационного ресурса. Основные составляющие индекса — это прямой и инвертированный списки, устанавливающие соответствие между поисковыми терминами и содержащими их документами.

Третий элемент поисковой машины — *аппарат поиска*, непосредственно обеспечивающий работу пользователя с индексом. Под этим термином подразумевают информационно-поисковый язык системы, пользовательский интерфейс и механизмы осуществления запросов в индексной базе.



Рис. 11.6. Поисковая система Altavista

Для того чтобы не увеличивать размеры словарей и индексов, введено такое понятие, как вес термина. Он определяется в процессе индексирования и зависит от метода индексирования, который используется данной поисковой машиной.

Методы индексирования делятся на статистические, теоретико-информационные и вероятностные.

В статистических методах документы рассматриваются как точки в информационном пространстве. Чем ближе группы терминов, составляющих документы, тем ближе находятся отображающие их точки. В качестве терминов индексации выбираются понижающие плотность пространства документов.

Теоретико-информационные методы основаны на предположении о том, что наибольшую информационную ценность имеют наименее часто встречающиеся слова. Для оценки полезности термина применяются концепции теории информации.

Вероятностные методы предполагают наличие обучающего множества документов для оценки релевантности результатов обработки запроса. Обучающее множество применяется для вычисления весовых коэффициентов, получаемых путем

оценки условной вероятности вхождения термина в данный документ в случае его релевантности (или нерелевантности). На основе этих коэффициентов определяется вес термина.

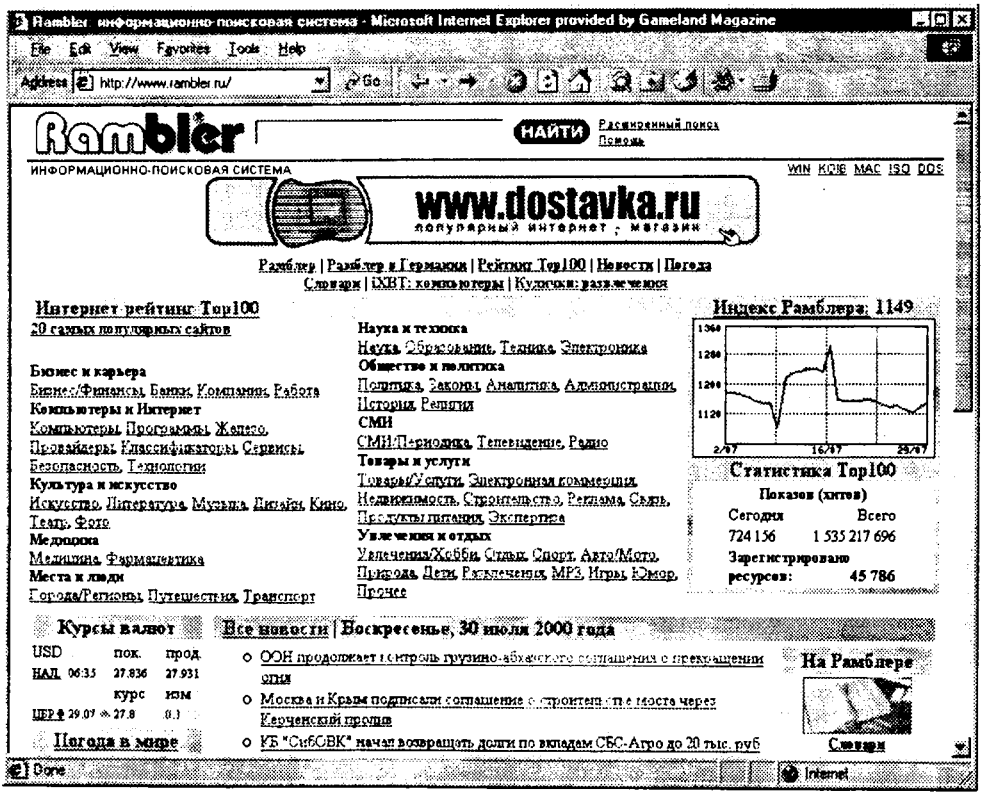


Рис. 11.7. Форма для запроса на российском поисковом сервере Rambler

При построении индекса реальные документы заменяются поисковыми образами документов. При индексирования нетекстовой информации в поисковые образы входят главным образом универсальные адреса ресурсов, в случае новостей и почтовых списков — поля Subject и Keywords. Из составляющих HTML-документы слов в поисковые образы обычно входят имеющие наибольший вес.

Формальную релевантность вычисляет система, на основании чего ранжируется выборка найденных документов. *Реальная релевантность* — это оценка самим пользователем ценности найденных документов.

Некоторые поисковые машины показывают дату, когда был проиндексирован тот или иной документ. Это помогает пользователю понять, насколько актуальным является ресурс, ссылку на который выдает поисковая машина.

Часто поисковые машины не включают определенные слова в свои индексы или могут не включать эти слова в запросы пользователей. Такими словами обычно считаются предлоги или просто очень часто использующиеся слова. Не включают их ради экономии места на носителях.



Рис. 11.8. Форма для запроса на российском поисковом сервере Яndex

В последнее время Web-серверы, предназначенные для поиска информации в Интернете, сочетают в себе возможности классификаторов и поисковых машин.

11.5. Перспективные технологии на основе Интернета

11.5.1. IP-телефония

Под *IP-телефонией* понимается технология, позволяющая использовать Интернет или любую другую IP-сеть в качестве средства организации и ведения телефонных разговоров и передачи факсов в режиме реального времени. Существует техническая возможность оцифровать звук или факсимильное сообщение и переслать его аналогично тому, как пересылаются цифровые данные. И в этом смысле IP-телефония использует Интернет (или любую другую IP-сеть) для пересылки голосовых или факсимильных сообщений между двумя пользователями компьютера в режиме реального времени.

Общий принцип действия телефонных серверов IP-телефонии таков: с одной стороны, сервер связан с телефонными линиями и может соединиться с любым

телефоном мира. С другой стороны, сервер связан с Интернетом и может связаться с любым компьютером в мире. Сервер принимает стандартный телефонный сигнал, оцифровывает его (если он исходно не цифровой), значительно сжимает, разбивает на пакеты и отправляет через Интернет по назначению с использованием протокола TCP/IP. Для пакетов, приходящих из Сети на телефонный сервер и уходящих в телефонную линию, операция происходит в обратном порядке. Обе составляющие операции (вход сигнала в телефонную сеть и его выход из телефонной сети) происходят в режиме реального времени.

Для того чтобы осуществить связь с помощью телефонных серверов, организация или оператор услуги должны иметь серверы в тех местах, куда и откуда планируются звонки. Стоимость такой связи на порядок меньше стоимости телефонного звонка по обычным телефонным линиям. Особенно велика эта разница для международных переговоров.

11.6. Электронная коммерция

Электронная коммерция подразумевает использование технологий глобальных компьютерных сетей для ведения бизнеса. Благодаря широкому распространению технологии World Wide Web в течение последних лет сеть Интернет из академической сети превратилась в популярную среду для общения, рекламы и бизнеса. Хотя электронная коммерция существовала и ранее, но именно популярность и доступность Интернета сделала возможным широкое использование электронной коммерции.

Электронная коммерция позволяет изменить практически все процессы, происходящие в современном бизнесе, интегрируя их в единое целое. Потребители могут искать, заказывать и оплачивать товары, используя Интернет, обмениваться информацией о товарах и услугах с другими пользователями. Правительственные организации могут использовать Интернет для сбора налоговых деклараций и распространения официальной информации.

11.6.1. Общие сведения об электронной коммерции

Первоначально под термином электронная коммерция понимались продажи товаров и переводы денежных средств с помощью компьютерных сетей. Всякий раз, когда мы используем кредитную или телефонную карту, мы принимаем участие в электронной коммерции. Если банки осуществляют перевод денежных средств с помощью систем электронных платежей, они также используют электронную коммерцию. Однако банковские системы платежей являются закрытыми. Круг действия подобной системы ограничен банками, принимающими в ней участие, или клиентами конкретного банка. Между тем Интернет является открытой системой и представляет собой совершенной новый тип взаи-

модействия с пользователем. Поэтому сейчас термин «электронная коммерция» охватывает практически все аспекты ведения бизнеса, которые возможны с использованием Интернета. Можно выделить следующие два аспекта электронной коммерции:

- **Электронная коммерция как торговля в сети Интернет.** Электронная торговля подразумевает продажу товаров и услуг с использованием Интернета.

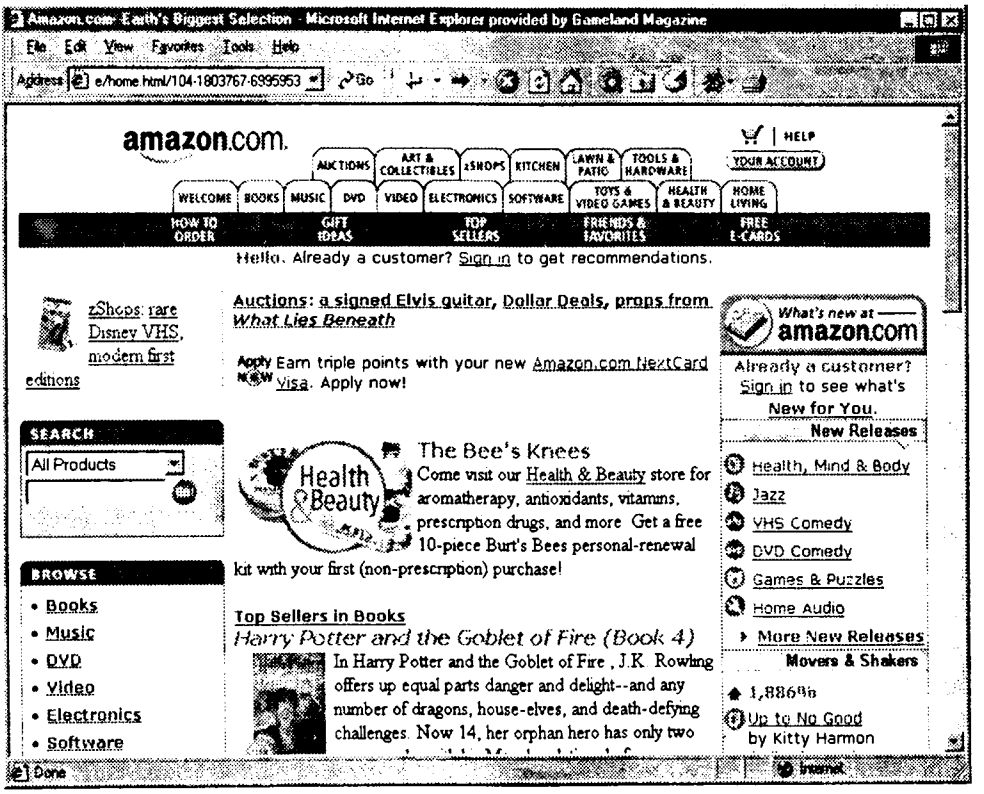


Рис. 11.9. Популярный американский Интернет-магазин Amazon.com

- **Электронная коммерция как рынок.** Электронная коммерция не ограничивается покупкой и продажей товаров в Интернете. Например, виртуальный магазин может не только предлагать свою продукцию пользователям Интернета, но и искать поставщиков продукции, заключать контракты с ними, оплачивать счета, нанимать сотрудников и проводить маркетинговые акции через Интернет. При этом компании, вовлеченные в процесс электронной торговли, могут даже не подозревать об этом. Электронная коммерция оказывает значительное влияние на процессы производства, распространения и обмена товарами, а также на то, каким образом потребитель получает информацию о товаре и производит торговую сделку.

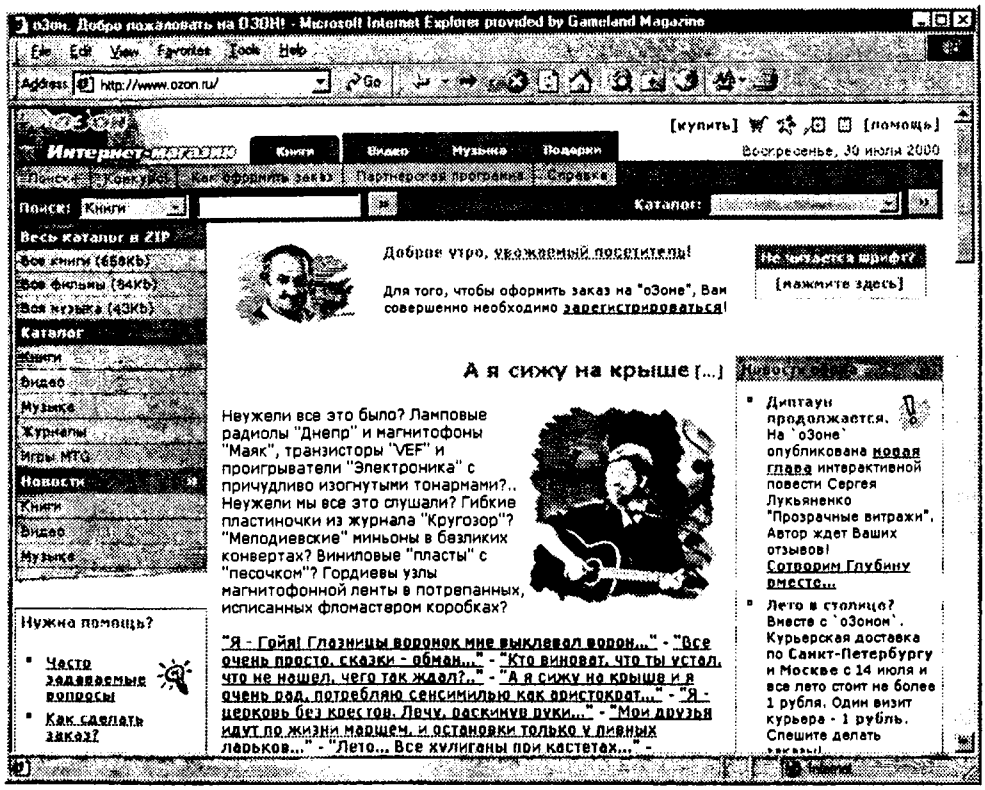


Рис. 11.10. Российский Интернет-магазин Ozon

11.6.2. Электронный рынок и роль информации в электронном бизнесе

В работе электронного рынка принимают участие компании-производители и потребители товаров и услуг. Изменения в области коммуникаций влияют на способ, которым потребители получают информацию о товарах и услугах, делают заказы, оплачивают продукцию и т. д.

Электронный рынок по многим параметрам похож на обычный рынок, основными компонентами которого являются:

- игроки (компании, поставщики, брокеры, магазины и покупатели);
- продукты (товары и услуги);
- процессы (поставка, производство, маркетинг, конкуренция, распространение, потребление и др.).

Разница заключается в том, что элементы электронного рынка (по крайней мере, часть из них) являются виртуальными (электронными). Необходимо отметить,

что несмотря на очевидные преимущества, связанные с возможностью охвата огромных масс потенциальных покупателей и продавцов, электронный рынок не способен решить все проблемы, присущие традиционным рынкам. Более того, он порождает собственные проблемы. Прежде всего, это появление новых видов трансакционных издержек, снижающих эффективность электронных сделок.

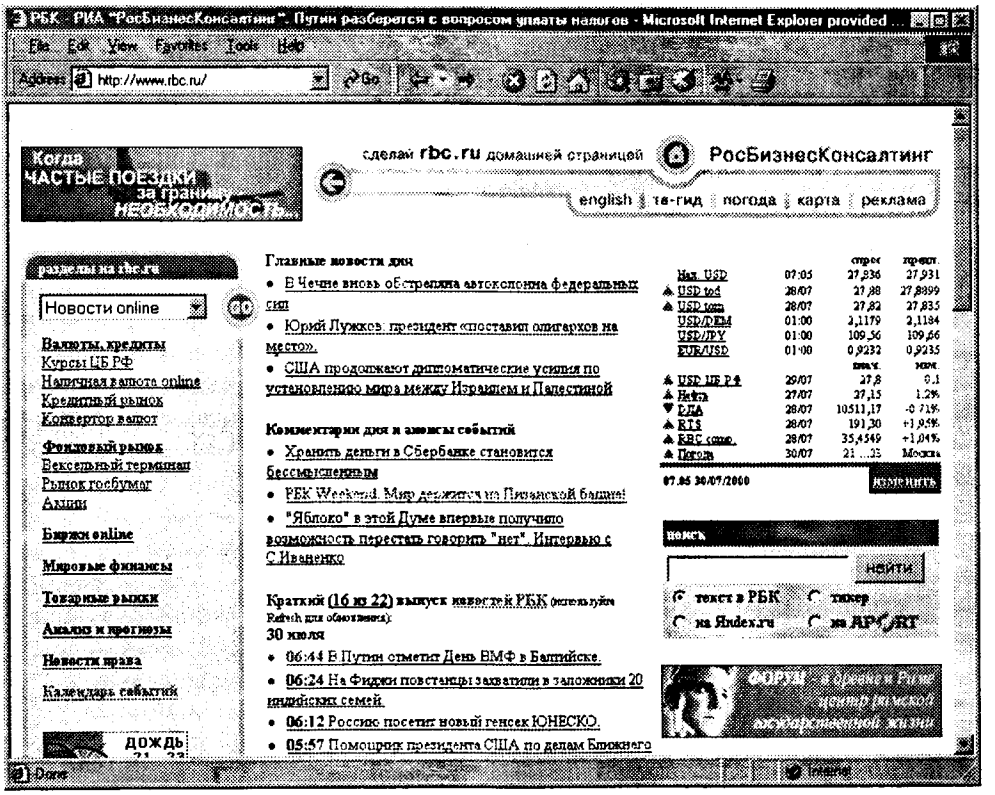


Рис. 11.11. Российское агентство финансовой информации «РосБизнесКонсалтинг»

Электронные (цифровые) продукты имеют высокую степень настраиваемости. Информация о вкусах покупателя позволяет существенно расширить диапазон предлагаемых товаров. На таком достаточно дифференцированном и сегментированном рынке цены будут стремиться к максимальным, которые покупатель согласен заплатить.

11.6.3. Системы электронных платежей

Существуют две основные области применения систем электронных платежей — это обмен данными (заказы, счета и др.) и электронный перевод денежных средств, который осуществляется между банками и имеет достаточно большой объем и относительно небольшое количество передаваемых полей, а также долгосрочные связи.

11.6.4. Электронные платежные средства

Существует три типа электронных платежных средств.

Первый тип представляет собой комбинацию обычных и электронных платежей. Например, платеж осуществляется традиционным способом, а подтверждение владельцу высылается по электронной почте. Или, напротив, платеж делается электронным способом, а подтверждение приходит по обычной почте.

Второй тип является расширением традиционного способа передачи денежных средств. Сюда входит передача номеров кредитных карточек электронным способом и использование так называемых смарт-карт, которые могут хранить и передавать всю информацию о своем владельце. В данном случае вся операция может происходить в электронном виде.

Третий тип включает различные виды цифровой наличности и электронных денег. Разница между первыми двумя типами и третьим заключается в том, что в последнем случае действительно переводятся деньги, а не только информация о них. Например, если передается только номер кредитной карточки — это платеж второго типа, а если сообщение само несет в себе некую сумму денег — это платеж третьего типа.

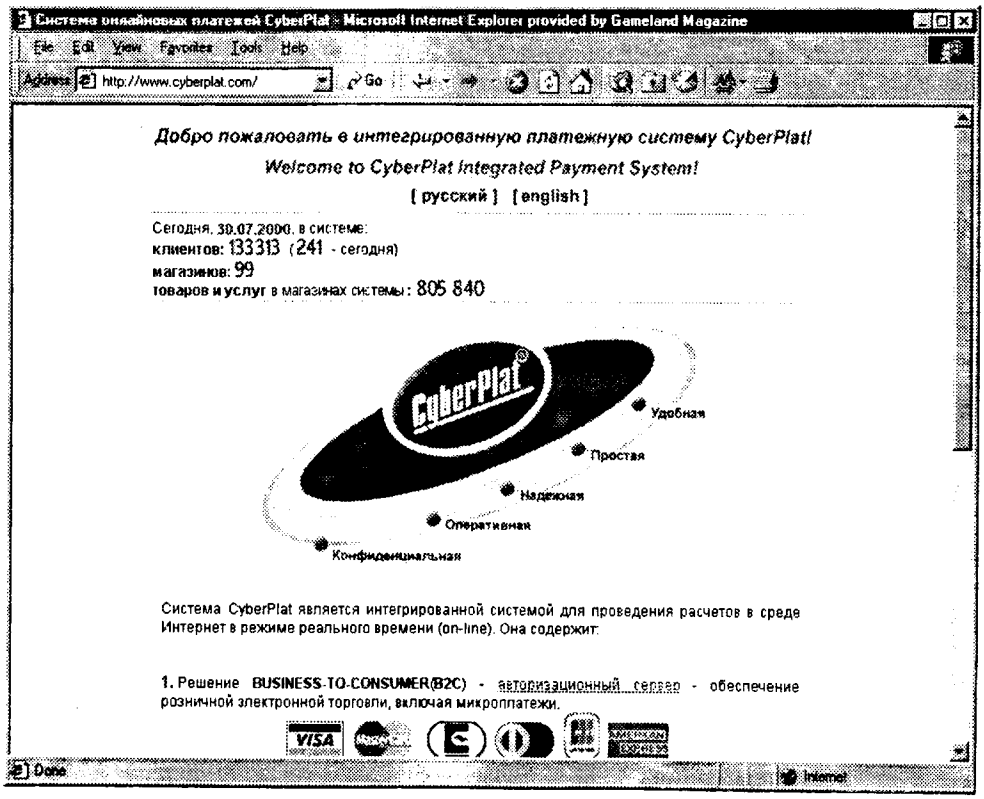


Рис. 11.12. Система электронных платежей через Интернет CyberPlat

Электронные деньги — это некоторые зашифрованные серийные номера, представляющие реальную сумму денег. В то же время они являются вполне полноценными денежными средствами в том смысле, что могут быть обменены на обычные деньги.

Смарт-карты являются небольшими устройствами, которые могут хранить информацию, необходимую для совершения транзакций в рамках электронной коммерции. Смарт-карты могут хранить электронную наличность, информацию о владельце, электронные ключи и другую информацию.

Микроплатежи — это особый вид электронных денег, необходимый для оплаты микротранзакций. Стоимость одной транзакции может быть очень мала, и вследствие этого ее может быть невозможно выразить в обычных денежных единицах, поэтому возникла необходимость в микроплатежах. Важно отметить, что микроплатежи являются особенностью электронной коммерции.

11.7. Защита информации в Интернете

Проведение финансовых операций с использованием Интернета, заказ товаров и услуг, использование кредитных карточек, доступ к закрытым информационным ресурсам, передача телефонных разговоров требуют обеспечения соответствующего уровня безопасности.

Конфиденциальная информация, которая передается по сети Интернет, проходит через определенное количество маршрутизаторов и серверов, прежде чем достигнет пункта назначения. Обычно маршрутизаторы не отслеживают проходящие сквозь них потоки информации, но возможность того, что информация может быть перехвачена, существует. Более того, информация может быть изменена и передана адресату в измененном виде. К сожалению, сама архитектура сети Интернет всегда оставляет возможность для недобросовестного пользователя осуществить подобные действия.

Всегда существует проблема выбора между необходимым уровнем защиты и эффективностью работы в сети. В некоторых случаях пользователями или потребителями меры по обеспечению безопасности могут быть расценены как меры по ограничению доступа и эффективности. Однако такие средства, как, например, криптография, позволяют значительно усилить степень защиты, не ограничивая доступ пользователей к данным.

11.7.1. Принципы защиты информации

Проблемы, возникающие с безопасностью передачи информации при работе в компьютерных сетях, можно разделить на три основных типа:

- перехват информации — целостность информации сохраняется, но ее конфиденциальность нарушена;
- модификация информации — исходное сообщение изменяется либо полностью подменяется другим и отсылается адресату;

- подмена авторства информации. Данная проблема может иметь серьезные последствия. Например, кто-то может послать письмо от вашего имени (этот вид обмана принято называть спуфингом) или Web-сервер может притворяться электронным магазином, принимать заказы, номера кредитных карт, но не высылать никаких товаров.

В соответствии с перечисленными проблемами при обсуждении вопросов безопасности под самим термином «безопасность» подразумевается совокупность трех различных характеристик обеспечивающей безопасность системы:

1. Аутентификация — это процесс распознавания пользователя системы и предоставления ему определенных прав и полномочий. Каждый раз, когда заходит речь о степени или качестве аутентификации, под этим следует понимать степень защищенности системы от посягательств сторонних лиц на эти полномочия.
2. Целостность — состояние данных, при котором они сохраняют свое информационное содержание и однозначность интерпретации в условиях различных воздействий. В частности, в случае передачи данных под целостностью понимается идентичность отправленного и принятого.
3. Секретность — предотвращение несанкционированного доступа к информации. В случае передачи данных под этим термином обычно понимают предотвращение перехвата информации.

11.7.2. Криптография

Для обеспечения секретности применяется *шифрование*, или *криптография*, позволяющая трансформировать данные в зашифрованную форму, из которой извлечь исходную информацию можно только при наличии ключа.

В основе шифрования лежат два основных понятия: алгоритм и ключ. *Алгоритм* — это способ закодировать исходный текст, в результате чего получается зашифрованное послание. Зашифрованное послание может быть интерпретировано только с помощью *ключа*.

Очевидно, чтобы зашифровать послание, достаточно алгоритма. Однако использование ключа при шифровании предоставляет два существенных преимущества. Во-первых, можно использовать один алгоритм с разными ключами для отправки посланий разным адресатам. Во-вторых, если секретность ключа будет нарушена, его можно легко заменить, не меняя при этом алгоритм шифрования. Таким образом, безопасность систем шифрования зависит от секретности используемого ключа, а не от секретности алгоритма шифрования. Многие алгоритмы шифрования являются общедоступными.

Количество возможных ключей для данного алгоритма зависит от числа бит в ключе. Например, 8-битный ключ допускает 256 (2^8) комбинаций ключей. Чем больше возможных комбинаций ключей, тем труднее подобрать ключ, тем надежнее зашифровано послание. Так, например, если использовать 128-битный ключ, то необходимо будет перебрать 2^{128} ключей, что в настоящее время не под силу

даже самым мощным компьютерам. Важно отметить, что возрастающая производительность техники приводит к уменьшению времени, требующегося для вскрытия ключей, и системам обеспечения безопасности приходится использовать все более длинные ключи, что, в свою очередь, ведет к увеличению затрат на шифрование.

Поскольку столь важное место в системах шифрования уделяется секретности ключа, то основной проблемой подобных систем является генерация и передача ключа. Существуют две основные схемы шифрования: *симметричное шифрование* (его также иногда называют традиционным или шифрованием с секретным ключом) и *шифрование с открытым ключом* (иногда этот тип шифрования называют асимметричным).

При *симметричном шифровании* отправитель и получатель владеют одним и тем же ключом (секретным), с помощью которого они могут зашифровывать и расшифровывать данные. При симметричном шифровании используются ключи небольшой длины, поэтому можно быстро шифровать большие объемы данных. Симметричное шифрование используется, например, некоторыми банками в сетях банкоматов. Однако симметричное шифрование обладает несколькими недостатками. Во-первых, очень сложно найти безопасный механизм, при помощи которого отправитель и получатель смогут тайно от других выбрать ключ. Возникает проблема безопасного распространения секретных ключей. Во-вторых, для каждого адресата необходимо хранить отдельный секретный ключ. В третьих, в схеме симметричного шифрования невозможно гарантировать личность отправителя, поскольку два пользователя владеют одним ключом.

В схеме *шифрования с открытым ключом* для шифрования послания используются два различных ключа. При помощи одного из них послание зашифровывается, а при помощи второго — расшифровывается. Таким образом, требуемой безопасности можно добиться, сделав первый ключ общедоступным (открытым), а второй ключ хранить только у получателя (закрытый, личный ключ). В таком случае любой пользователь может зашифровать послание при помощи открытого ключа, но расшифровать послание способен только обладатель личного ключа. При этом нет необходимости заботиться о безопасности передачи открытого ключа, а для того чтобы пользователи могли обмениваться секретными сообщениями, достаточно наличия у них открытых ключей друг друга.

Недостатком асимметричного шифрования является необходимость использования более длинных, чем при симметричном шифровании, ключей для обеспечения эквивалентного уровня безопасности, что сказывается на вычислительных ресурсах, требуемых для организации процесса шифрования.

11.7.3. Электронная подпись

Даже если послание, безопасность которого мы хотим обеспечить, должным образом зашифровано, все равно остается возможность модификации исходного сообщения или подмены этого сообщения другим. Одним из путей решения этой проблемы является передача пользователем получателю краткого представления пере-

даваемого сообщения. Подобное краткое представление называют контрольной суммой, или дайджестом сообщения.

Контрольные суммы используются при создании резюме фиксированной длины для представления длинных сообщений. Алгоритмы расчета контрольных сумм разработаны так, чтобы они были по возможности уникальны для каждого сообщения. Таким образом, устраняется возможность подмены одного сообщения другим с сохранением того же самого значения контрольной суммы.

Однако при использовании контрольных сумм возникает проблема передачи их получателю. Одним из возможных путей ее решения является включение контрольной суммы в так называемую *электронную подпись*.

При помощи электронной подписи получатель может убедиться в том, что полученное им сообщение послано не сторонним лицом, а имеющим определенные права отправителем. Электронные подписи создаются шифрованием контрольной суммы и дополнительной информации при помощи личного ключа отправителя. Таким образом, кто угодно может расшифровать подпись, используя открытый ключ, но корректно создать подпись может только владелец личного ключа. Для защиты от перехвата и повторного использования подпись включает в себя уникальное число — порядковый номер.

11.7.4. Аутентификация

Аутентификация является одним из самых важных компонентов организации защиты информации в сети. Прежде чем пользователю будет предоставлено право получить тот или иной ресурс, необходимо убедиться, что он действительно тот, за кого себя выдает.

При получении запроса на использование ресурса от имени какого-либо пользователя сервер, предоставляющий данный ресурс, передает управление серверу аутентификации. После получения положительного ответа сервера аутентификации пользователю предоставляется запрашиваемый ресурс.

При аутентификации используется, как правило, принцип, получивший название «что он знает», — пользователь знает некоторое секретное слово, которое он посылает серверу аутентификации в ответ на его запрос. Одной из схем аутентификации является использование стандартных паролей. Эта схема является наиболее уязвимой с точки зрения безопасности — пароль может быть перехвачен и использован другим лицом. Чаще всего используются схемы с применением одноразовых паролей. Даже будучи перехваченным, этот пароль будет бесполезен при следующей регистрации, а получить следующий пароль из предыдущего является крайне трудной задачей. Для генерации одноразовых паролей используются как программные, так и аппаратные генераторы, представляющие собой устройства, вставляемые в слот компьютера. Знание секретного слова необходимо пользователю для приведения этого устройства в действие.

Одной из наиболее простых систем, не требующих дополнительных затрат на оборудование, но в то же время обеспечивающих хороший уровень защиты, явля-

ется S/Key, на примере которой можно продемонстрировать порядок представления одноразовых паролей.

В процессе аутентификации с использованием S/Key участвуют две стороны — клиент и сервер. При регистрации в системе, использующей схему аутентификации S/Key, сервер присылает на клиентскую машину приглашение, содержащее зерно, передаваемое по сети в открытом виде, текущее значение счетчика итераций и запрос на ввод одноразового пароля, который должен соответствовать текущему значению счетчика итерации. Получив ответ, сервер проверяет его и передает управление серверу требуемого пользователем сервиса.

11.7.5. Защита сетей

В последнее время корпоративные сети все чаще включаются в Интернет или даже используют его в качестве своей основы. Учитывая то, какой урон может принести незаконное вторжение в корпоративную сеть, необходимо выработать методы защиты. Для защиты корпоративных информационных сетей используются брандмауэры. *Брандмауэр* — это система или комбинация систем, позволяющие разделить сеть на две или более частей и реализовать набор правил, определяющих условия прохождения пакетов из одной части в другую. Как правило, эта граница проводится между локальной сетью предприятия и Интернетом, хотя ее можно провести и внутри. Однако защищать отдельные компьютеры невыгодно, поэтому обычно защищают всю сеть.

Брандмауэр пропускает через себя весь трафик и для каждого проходящего пакета принимает решение — пропускать его или отбросить. Для того чтобы брандмауэр мог принимать эти решения, для него определяется набор правил.

Брандмауэр может быть реализован как аппаратными средствами (то есть как отдельное физическое устройство), так и в виде специальной программы, запущенной на компьютере.

Как правило, в операционную систему, под управлением которой работает брандмауэр, вносятся изменения, цель которых — повышение защиты самого брандмауэра. Эти изменения затрагивают как ядро ОС, так и соответствующие файлы конфигурации. На самом брандмауэре не разрешается иметь разделов пользователей, а следовательно, и потенциальных дыр — только раздел администратора. Некоторые брандмауэры работают только в однопользовательском режиме, а многие имеют систему проверки целостности программных кодов.

Брандмауэр обычно состоит из нескольких различных компонентов, включая фильтры или экраны, которые блокируют передачу части трафика.

Все брандмауэры можно разделить на два типа:

- пакетные фильтры, которые осуществляют фильтрацию IP-пакетов средствами фильтрующих маршрутизаторов;
- серверы прикладного уровня, которые блокируют доступ к определенным сервисам в сети.

Таким образом, брандмауэр можно определить как набор компонентов или систему, которая располагается между двумя сетями и обладает следующими свойствами:

- весь трафик из внутренней сети во внешнюю и из внешней сети во внутреннюю должен пройти через эту систему;
- только трафик, определенный локальной стратегией защиты, может пройти через эту систему;
- система надежно защищена от проникновения.

Ключевые понятия

- глобальная компьютерная сеть;
- Интернет (Internet);
- IP-адрес и IP-протоколы;
- подсеть;
- домен;
- протоколы TCP, TCP/IP и UDP;
- провайдер;
- информационные сервисы Интернета: FTP, WWW и др.;
- URL;
- HTML;
- HTTP;
- MIME;
- электронная подпись;
- аутентификация.

Контрольные вопросы

1. Сформулируйте основные отличия глобальных вычислительных сетей от локальных.
2. Сформулируйте базовые принципы построения сети Интернет.
3. Что представляет собой современная структура Интернета?
4. Какие информационные сервисы Интернета вы можете назвать?
5. Дайте характеристику формату сообщения электронной почты.
6. В чем содержание стандарта MIME?
7. Какие основные средства поиска существуют в Интернете?
8. Охарактеризуйте основные направления развития электронной коммерции.
9. Приведите классификацию проблем, возникающих при передаче информации в глобальных компьютерных сетях.

10. Что такое электронная подпись и для чего она служит?
11. Перечислите основные схемы шифрования.
12. Перечислите основные меры, используемые для организации защиты сетей.

Литература

1. Бабушкин М., Иваненко С., Коростелев В. Web-сервер в действии. СПб.: Питер, 1997.
2. Кенцл Т. Форматы файлов Internet. СПб.: Питер, 1997.
3. Олифер В. Г., Олифер Н. А. Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2000.
4. Павлов А. CGI-программирование: учебный курс. СПб.: Питер, 2000.
5. Холзнер С. Perl: Специальный справочник. СПб.: Питер, 2000.
6. Хеслоп Б., Бадник Л. HTML с самого начала. СПб.: Питер, 1997.

Глава 12. Информационные базы данных и электронные библиотеки

Рассмотренные в предыдущих главах средства взаимодействия компьютерных систем обеспечивают лишь «технические» возможности доступа к информации. Однако нетрудно заметить, что осуществление поиска необходимых данных требует решения обширного круга задач логического плана. Формы и методы их решения описываются в этой главе.

12.1. Общие принципы построения информационно-поисковых систем

12.1.1. Модели организации хранения и поиска документов

Строго говоря, проблема поиска документа, отвечающего тем или иным критериям, возникает в любом хранилище данных, содержащем более одного документа. Очевидно, что решение этой проблемы так или иначе замыкается на те способы, которые применяются при создании систем хранения. Можно указать два основных способа:

- использование иерархической модели;
- использование гипертекстовой модели.

Использование *иерархической модели* подразумевает многоуровневую рубрикацию информационных ресурсов. Для выбора пути к нужному документу используются описания, составленные службой поддержки данной системы.

Гипертекстовая модель позволяет связывать документы ссылками, которые располагаются непосредственно в тексте.

Эти две модели имеют очевидные недостатки. Так как и многоуровневая рубрикация, и простановка ссылок выполняется высококвалифицированными специалистами, объем обработанных таким образом документов не может быть очень

большим. По этой же причине страдает актуальность описания массива документов. Помимо этого связанные документы ограничены какой-либо одной предметной областью, о которой, к тому же, у пользователя системы может быть иное представление, чем у составителя рубрикатора. И наконец, для нахождения необходимого документа пользователю таких систем потребуется просмотреть множество документов, полезной информацией в которых будут только ссылки на другие ресурсы.

Эти проблемы становятся особенно острыми при больших объемах информации, высокой скорости их обновления и высокой разнородности потребностей пользователей. Помочь в решении этих проблем призваны *информационно-поисковые системы* (*ИПС*). Такие системы, однажды созданные, могут работать автономно. Принцип их взаимодействия с пользователем заключается в выдаче списка указателей на документы, удовлетворяющие запросу. Этот список может быть отсортирован по релевантности (степени соответствия документа запросу). Таким образом, ИПС может обеспечить очень быстрый поиск необходимого документа — при том, что от пользователя требуется лишь ввести запрос.

Первые информационно-поисковые системы были созданы достаточно давно. Большинство открытий в этой области приходится на 70-е и 80-е годы. Сейчас, с развитием Интернета, количество пользователей этих систем исчисляется миллионами, а в скором будущем будет исчисляться миллиардами. Так же стремительно растет количество документов, хранящихся в Интернете, что ставит все более сложные задачи перед разработчиками ИПС.

12.1.2. Основные принципы информационного поиска

Основные принципы информационного поиска были сформулированы еще в первой половине этого века. Между 1939 и 1945 годами У. Е. Баттеном была разработана система для отыскания патентов. Каждый патент классифицировался в соответствии с понятиями, к которым он имел отношение. Для каждого понятия, использовавшегося в системе, была создана 800-позиционная перфокарта. При регистрации в системе нового патента находились карты, соответствующие тем понятиям, которые в нем рассматриваются, и в позиции пробивались номера патента. Чтобы найти патент, в котором рассматривается одновременно несколько понятий, необходимо было совместить карты, соответствующие этим понятиям. Номер нужного патента определялся из позиции просвета.

Основные принципы информационного поиска с тех пор не изменились. На примере уже этой ИПС видно, как происходит процесс поиска. Во-первых, должен быть создан массив указателей на информационные ресурсы. *Указатель* (index) содержит в себе некое свойство документа и ссылки на документы, этим свойством обладающие. Указатели могут быть различных видов. Широко распространен, например, *авторский указатель*. Такой указатель позволяет получить ссылки на работы интересующего нас автора. Также указатели могут быть составлены и по другим атрибутам документа. В системе Баттена использовался *предмет-*

ный указатель, то есть документы классифицировались по понятиям (предметам), которые в них затрагиваются.

Процесс создания указателей на документы называется *индексированием*, а термины, использующиеся для индексирования, называются *терминами индексирования*. В случае с авторским указателем роль терминов индексирования будут выполнять фамилии авторов хранящихся в фонде работ. Совокупность используемых терминов индексирования называется *словарем*.

Массив указателей, полученный после индексации информационных ресурсов, называется *индексом* (Index database).

После создания индекса к нему обращаются посредством запросов. Так как процесс поиска заключается в сопоставлении запроса пользователя с имеющимися данными, полученный запрос также должен быть переведен на язык индексирования. В индексе выполняется поиск соответствующих запросу документов, пользователю выдается список ссылок на подходящие ресурсы.

Для повышения скорости индексирования и поиска словарь и индекс должны быть упорядочены по системе, наиболее отвечающей задачам поиска в данной предметной области.

12.1.3. Предметное индексирование и механизм поиска

Когда говорят об информационно-поисковой системе, подразумевают, что она использует *предметный указатель*. Предметный указатель позволяет отыскивать документы, касающиеся некоего «предмета». Для составления предметного указателя анализируется содержание документа и определяется «предмет» или «предметы», о которых в документе идет речь. Затем названия этих предметов переводятся на *информационно-поисковый язык* (ИПЯ). Таким образом, мы получаем *поисковый образ документа* (ПОД). Проиндексировав (создав поисковые образы) все информационные ресурсы, мы получаем то, что принято называть индексом (index database) — основной массив данных ИПС.

Так как процесс поиска заключается в сопоставлении запроса пользователя с имеющимися данными, полученный запрос также должен быть переведен на ИПЯ. После сопоставления переведенного на ИПЯ запроса и поисковых образов документов пользователь получает список ссылок на документы, которые соответствуют, по мнению системы, его запросу.

Типовая схема ИПС, использующей предметное индексирование, представлена на рис. 12.1.

Как видно, поиск происходит не по тексту документов, а по их поисковым образам, составленным на ИПЯ. Поэтому ИПЯ — основная часть информационно-поисковой системы, от которой в первую очередь зависит качество системы.

В состав информационно-поискового языка входят:

1. Словарь индексационных терминов — множество терминов индексирования.
2. Кодовый словарь — множество кодовых терминов.

3. Словарь входов — множество входных терминов.
4. Вспомогательные средства языка индексирования — средства, используемые совместно с индексационными терминами для расширения или сужения определенных понятий.
5. Правила использования языка индексирования.

Для повышения эффективности поиска словарь, используемый системой, должен быть *контролируемым*, то есть он должен быть организован таким образом, чтобы полнота и точность поиска была оптимальной. Очевидно, что организация словаря зависит от многих факторов — предметной области, в которой будет использоваться ИПС, характера интересов пользователей, степени их подготовки и т. д.

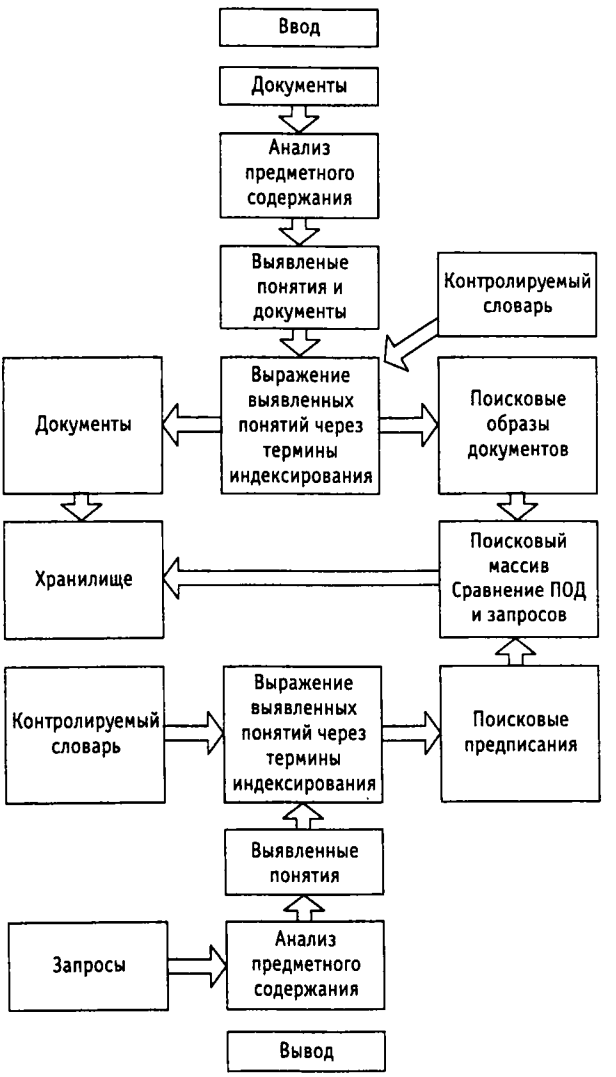


Рис. 12.1. Типовая схема ИПС

Для улучшения результатов поиска необходимо определить степень специфичности терминов, используемых при индексации. Принято использовать два принципа — использование наиболее специфического термина, соответствующего объему и содержанию отражаемого понятия, и избыточное индексирование.

Под *избыточным индексированием* понимается дополнение поискового образа терминами, связанными с основным. При этом могут использоваться термины, связанные как с основным отношением обобщения или спецификации, так и ассоциативной связью. Дополнение поискового образа терминами с ассоциативной связью может увеличить полноту поиска, но неизбежно понижает его точность.

Недостатком избыточного индексирования является также увеличение объема поисковых образов. Для решения этой проблемы во многих ИПС используется избыточное индексирование не документов, а запросов.

Использование предметного индексирования не исключает использования при создании поискового образа атрибутов документа. Это могут быть такие атрибуты, как данные об авторе, дата публикации, язык публикации и т. д.

12.1.4. Стратегии поиска

Точность и полнота поиска зависят не только от характеристик самой ИПС, но и от того, как создается запрос. Идеальный запрос может быть составлен пользователем, в полном объеме знакомым с той предметной областью, которая его интересует, а также с используемой ИПС. Но такому пользователю ИПС, очевидно, не нужна.

Остальные же пользователи вынуждены довольствоваться или низкой точностью поиска, или низкой полнотой. Для повышения качества поиска можно использовать различные методы. Наиболее употребляемый из них — использование логических операторов AND, OR, NOT.

Использование логических операторов — довольно простой способ повысить релевантность выдаваемых документов, но он имеет и свои недостатки. Главный из них — плохая масштабируемость. Применение оператора AND может сильно сузить выдачу, а оператора OR — сильно расширить.

Степень точности и полноты поиска зависит от того, насколько общие термины использовались при формулировке запроса. Может быть неверным использование как наиболее общих терминов (возрастает уровень информационного шума), так и слишком специфичных терминов (снижается полнота поиска). Использование слишком специфичных терминов может быть чревато еще и тем, что в словаре ИПС этого термина может не оказаться.

В общем виде процедура поиска является процедурой итеративной, то есть за этапом выдачи результатов поиска следует коррекция запроса, поиск по этому запросу и т. д. Схематично такая процедура показана на рис. 12.2.

Коррекция запроса происходит исходя из количества полученных документов и их релевантности, и может выполняться как пользователем, так и самой информационно-поисковой системой.

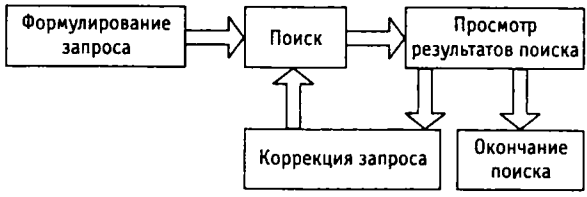


Рис. 12.2. Процедура поиска

В зависимости от соотношения полноты и точности найденных документов пользователь может сузить или расширить область поиска, перейдя к более общим или, наоборот, более специфичным терминам, а также использовав родственные понятия. В случае поиска по нескольким терминам такая коррекция области поиска может происходить по одному из нескольких терминов, что позволяет изменять эту область достаточно плавно.

Может оказаться полезным знание пользователя о наличии определенно релевантных документов. Не найдя их в списке найденных документов, область поиска надо расширить.

Коррекция запроса системой информационного поиска происходит на основании анализа документов, помеченных пользователем как наиболее точно отвечающих его потребности. В таком случае при следующем поиске система ищет те документы, в которых, помимо заданных в первоначальном запросе, содержатся термины, встречающиеся в документах, отмеченных пользователем.

Улучшить результаты поиска можно различными способами, если функции для этого предоставляются интерфейсом информационно-поисковой системы.

12.1.5. Интерфейс системы

Важным фактором, во многом определяющим эффективность поиска, может быть вид представления информации в программе, то есть ее интерфейс. По форме диалога, способу задания условия отбора и механизму поиска рассматриваемые программные средства можно разделить на два класса:

- системы *рубрикационного типа*;
- *структурно-логические* системы.

Первые реализуются интерфейсом в виде иерархических последовательно раскрывающихся списков, через которые обеспечивается доступ к тематически связанным группам документов. Раскрывая очередную рубрику и перемещаясь, таким образом, по тематической иерархии, пользователь уточняет предметную область и увеличивает (усредненно) степень точности соответствия выдаваемых документов и информационной потребности. При таком решении предопределенность соотнесения документов с отдельными рубриками компенсируется логичностью естественно-научной классификационной схемы, заменяющей пользователю путеводитель.

Структурно-логические методы формирования запроса обычно используются для работы с базами данных структурированной информации, когда каждый документ состоит из многих информационных полей, возможно, разного типа. Критерий отбора в этом случае строится как логическая комбинация простых, сводящихся к проверке условия присутствия или отсутствия в документе, слов (имен собственных или имен понятий, определяющих предмет поиска).

При составлении запроса к системе используют либо «меню-ориентированный» подход, либо командную строку. Первый позволяет ввести список терминов, обычно разделяемых пробелом, и выбрать тип логической связи между ними. Логическая связь распространяется на все термины. Многие ИПС позволяют сохранять запросы пользователя — в большинстве систем это просто фраза на ИПЯ, которую можно расширить за счет добавления новых терминов и логических операторов. Но это только один способ использования сохраненных запросов, называемый расширением или уточнением запроса. Для выполнения этой операции традиционная ИПС хранит не запрос как таковой, а результат поиска — список идентификаторов документов, который объединяется или пересекается со списком, полученным при поиске документов по новым терминам.

12.1.6. ИПС глобальных сетей

ИПС глобальной сети имеет отличия, обусловленные как характером сети, так и особенностями работы пользователей такой системы. Рассмотрим основные особенности использования ИПС в глобальной сети на примере сети Интернет.

Схематично ИПС для Интернета выглядит так, как показано на рис. 12.3:

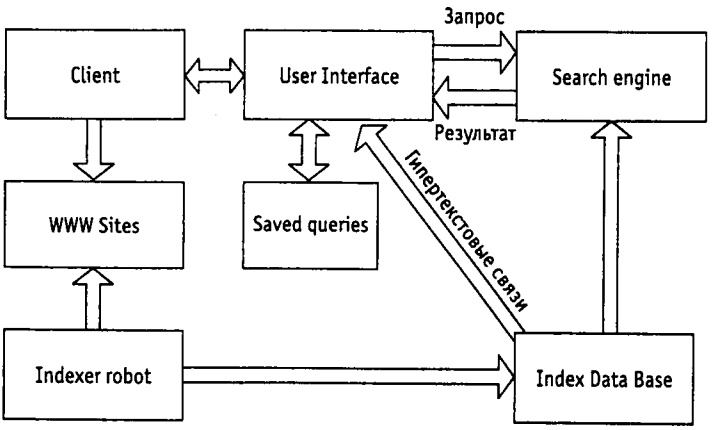


Рис. 12.3. ИПС для Интернета

Client (клиент) на этой схеме — это программа просмотра конкретного информационного ресурса. Такая программа обеспечивает просмотр документов WWW, Gopher, Wais, FTP-архивов, почтовых списков рассылки и групп новостей Usenet.

В свою очередь, все эти информационные ресурсы являются объектом поиска информационно-поисковой системы.

User interface (пользовательский интерфейс) — способ общения пользователя с поисковым аппаратом: системой формирования запросов и просмотра результатов поиска.

Search engine (поисковая машина) — служит для трансляции запроса на информационно-поисковом языке, в формальный запрос системы, поиска ссылок на информационные ресурсы Сети и выдачи результатов этого поиска пользователю.

Index database (индекс базы данных) — индекс, который является основным массивом данных ИПС и служит для поиска адреса информационного ресурса.

Queries (запросы пользователя) — сохраняются в его (пользователя) личной базе данных. На отладку каждого запроса уходит достаточно много времени, и поэтому чрезвычайно важно запоминать запросы, на которые система дает хорошие ответы.

Index robot (робот индексирования) — служит для просмотра данных в Интернете и поддержания базы данных индекса в актуальном состоянии. Эта программа является основным источником информации о состоянии информационных ресурсов сети.

WWW sites — это весь Интернет, или, точнее, информационные ресурсы, просмотр которых обеспечивается программами просмотра.

Как мы видим, источником информации о состоянии информационных ресурсов сети является робот-индексировщик. Это программа, которая по определенному алгоритму «заходит» на различные страницы, «читает» их и индексирует.

Индекс поисковых систем Интернета обновляется с периодичностью около недели. Отсюда видно, что в индекс поисковой системы не могут попасть материалы, например, периодических изданий, так как выходят они заведомо чаще, чем обновляется индекс.

Еще одна проблема заключается в том, что не все документы хранятся в виде файлов HTML, с которыми роботу работать легче всего. Если информация хранится в другом формате, может сложиться ситуация, когда адрес страницы, выдаваемой пользователю, содержит параметры, которые робот не знает, и, следовательно, он не может эти данные проиндексировать

Объем информации, опубликованной в Интернете, приводит также к ограничению количества терминов, которыми индексируется документ. Современные ИПС в Интернете используют порядка 100 терминов для индексации документа. Выбор терминов, используемых для индексации, зависит от реализации данной системы. Чаще всего первым критерием является отношение частоты употребления термина в документе к частоте употребления этого термина во всех ранее проиндексированных документах. То есть наибольший вес присваивается тем терминам, которые наиболее часто встречаются в данном документе и наиболее редко — во всех остальных проиндексированных документах. Термины, которые исполь-

зуются в очень большом количестве документов, при индексировании не используются совсем.

Для определения терминов индексирования, используемых для создания поискового образа, робот может также использовать разметку индексируемой страницы. И в индексе присваивать наибольший вес термину, используемому, например, в заголовке. Автор информационного ресурса также может повлиять на индексацию собственной страницы, указав роботу, какие термины надо использовать для индексирования. Но многие поисковые системы отказались от использования описаний ресурсов, представленных авторами. Это было сделано по причине недобросовестности некоторых авторов, которые использовали для описания своих страниц термины, наиболее часто встречающиеся в запросах.

Так как на запрос могут быть выданы ссылки на сотни ресурсов, необходимо предоставить пользователю отсортированный список. Наиболее часто используется сортировка по релевантности. Она происходит по тем же принципам, что и отбор терминов, применяющихся при индексировании.

Как уже отмечалась ранее, произвести точный поиск тем сложнее, чем шире круг потребностей пользователей системы. В глобальной сети эта проблема принимает глобальный же характер.

Очень сильно усложняется поиск по причине непрофессионализма как пользователя, формулирующего запрос, так и автора информационного ресурса. И если непрофессионализм пользователя мешает лишь ему самому (если не считать непроизводительной загрузки поискового сервера), то непрофессионализм автора ресурса стоит гораздо больше. Многие отмечают все время растущий уровень шума в результатах, выдаваемых на запрос.

Для уменьшения уровня этого шума может использоваться платная регистрация ресурса, которая подразумевает, что автор ответственно относится к его содержимому. Существует, например, система платной регистрации RealNames. База данных этой службы используется некоторыми поисковыми системами. Ресурсы, зарегистрированные в базе RealNames, будут помещаться в начало списка найденных документов.

12.2. Справочно-правовые системы

12.2.1. Понятие, назначение и преимущества

Принятие обоснованных решений в сфере как экономики, так и политики невозможно без обладания достаточным объемом правовой информации. Особенно остро эта необходимость ощущается в период реформирования экономического и политического устройства. Задачу удовлетворения потребности в своевременном предоставлении необходимого объема правовой информации решают различные средства массовой информации (СМИ).

В этой области конкурируют как традиционные СМИ, так и справочно-правовые системы (СПС). По-настоящему эффективная СПС может быть создана только с

применением современных информационных технологий. Созданная таким образом СПС называется компьютерной.

Компьютерная *справочно-правовая система* — это программный комплекс, включающий в себя массив правовой информации и инструменты для работы с ним. Эти инструменты могут позволять производить поиск документов, формировать подборки документов, выводить документы или их фрагменты на печать.

Преимущества компьютерных СПС очевидны. Это и доступность информации, и удобство работы с нею. Проблема же, присущая таким системам, — недостаточная оперативность — может быть решена с помощью глобальной сети Интернет.

12.2.2. Обзор рынка СПС в России

На рынке справочно-правовых систем в России работает большое количество фирм как разрабатывающих собственные программные комплексы, так и обслуживающих существующие. Наиболее известны следующие продукты таких фирм (по данным АО «Консультант Плюс»):

- «Консультант Плюс» (АО «Консультант Плюс»);
- «ГАРАНТ» (НПП «Гарант-Сервис»);
- «Кодекс» (Центр компьютерных разработок).

Системы, созданные государственными предприятиями для обеспечения потребностей в правовой информации государственных ведомств:

- «Эталон» (НЦПИ при Министерстве юстиции РФ);
- «Система» (НТЦ «Система» при ФАПСИ).

Кроме того, на российском рынке представлены такие системы, как:

- «ЮСИС» (фирма «Инталекс»);
- «Референт» (ЗАО «Референт-Сервис»);
- «Юридический мир» (издательство «Дело и право»);
- «Ваше право» и «Юрисконсульт» (фирма «Информационные системы и технологии»);
- «1С: Кодекс», «1С: Гарант», «1С: Эталон» (компания «1С»);
- «Законодательство России» (Ассоциация развития банковских технологий) и некоторые другие.

Различные продукты могут существенно различаться не только по задачам, решаемым с их помощью, но и по качеству.

Качество СПС зависит как от качества предоставляемой информации, так и от качества инструментов, используемых для работы с ней. Применение самых последних компьютерных технологий не поможет, если в СПС не содержится полной правовой информации или если информация обновляется с недостаточной периодичностью. И наоборот, СПС, содержащая даже самую полную и оперативно обновляющуюся информацию, не будет достаточно эффективна, если не предоставлены качественные инструменты для обработки этой информации.

Таким образом, основными параметрами, позволяющими определить качество содержания информационной базы, являются:

- полнота информации;
- достоверность информации;
- оперативность обновления информации.

Параметры, характеризующие качество программной оболочки:

- поисковые возможности системы;
- средства актуализации информации;
- дополнительные сервисные функции.

12.2.3. Основные свойства информационных банков СПС

Оценка полноты, достоверности и оперативности обновления информации основывается на количественных показателях. Оценка же качества юридической обработки поступающих в информационный банк документов достаточно субъективна.

Без юридической обработки СПС является всего лишь электронным аналогом бумажных изданий. Ее цель — систематизация документов для повышения эффективности их дальнейшего использования.

Юридическая обработка обычно состоит из следующих основных этапов:

- классификации документов;
- выявления взаимосвязей между различными документами;
- составления примечаний к документу.

Классификация документов предназначена для последующего их поиска по некоторым признакам. Это могут быть как формальные признаки, такие как выходные данные, так и неформальные — темы, которые рассматриваются в этих документах. Классификация производится на основании классификатора данной системы. В общем случае классификатор — это иерархическая структура, содержащая все понятия, используемые для описания документов, входящих в информационную базу.

Качество последующей юридической обработки документа, а также эффективность его поиска напрямую зависят от корректности классификации документа в системе.

Выявление взаимосвязей документа позволяет создать список документов, его дополняющих. Таким образом, пользователю предоставляется возможность максимально полно ознакомиться с интересующей его темой, начав работу всего лишь с одним документом.

В качестве примечаний к документу обычно используются ссылки на связанные с ним информационные ресурсы, данные о текущем статусе документа, комментарии юриста.

12.2.4. Основные возможности программных технологий СПС

Хранение и обработка больших объемов правовой информации

Необходимость хранения и регулярного обновления больших объемов информации накладывает на СПС ряд требований. Эти требования связаны с потребностями как пользователей системы, так и разработчиков, производящих обновление и юридическую обработку массива правовой информации. Объем ежемесячно вводимой и обновляемой информации может достигать нескольких тысяч документов, что вынуждает разработчиков с самого начала создания системы приспосабливать ее к таким условиям поддержки актуальности информационного банка СПС.

Основные поисковые и сервисные возможности

Поиск документов можно условно разделить на два вида:

- поиск по реквизитам документов;
- поиск по тексту документов.

Поиск по реквизитам документа позволяет найти документ с определенными датой, номером, выпущенный определенным органом и т. д. Для повышения эффективности поиска условия в таком виде поиска можно сочетать, задавать интервалы номеров или дат.

Поиск по тексту документов заключается в простом переборе массива документов и нахождении среди них тех, которые содержат слово, заданное пользователем. Таким образом, такой поиск позволяет найти документы по какой-либо тематике. Недостатком поиска по тексту является большой *шум*, то есть попадание в список найденных документов большого количества документов, удовлетворяющих запросу, но пользователю в данный момент не нужных. Уменьшить уровень такого шума может, пожалуй, лишь опыт составления запросов к системе.

Многие СПС позволяют использовать одновременно поиск и по реквизитам, и по тексту документов, что может существенно повысить эффективность поиска.

Дополнительные сервисные возможности

Некоторые СПС обладают дополнительными возможностями, позволяющими сделать работу с системой более простой и эффективной. Из таких возможностей наибольшее значение для пользователя имеют улучшенная навигация по информационной базе, возможность сохранять личные предпочтения и подборки документов, возможности редактирования документов и вывода их на печать.

Для обеспечения удобной навигации по информационной базе обычно используют гипертекстовые ссылки между документами, а также перемещение по списку ранее просмотренных документов и использование закладок. Возможность сохранять подборки документов позволяет пользователю приспосабливать систему

к своим собственным нуждам. Средства редактирования текста документов необходимы для их обработки пользователем, особенно в том случае, когда информационная база содержит типовые формы документов, предназначенные для заполнения. В СПС может быть не включен собственный текстовый редактор, если предусмотрены функции экспорта документов в форматы популярных текстовых редакторов.

Технологии передачи информации

Доступ пользователя к информации, хранящейся в СПС, может осуществляться двумя способами, каждый из которых имеет свои достоинства и недостатки:

- работа с удаленной базой;
- работа с локальной базой.

При работе с удаленной базой пользователю нет необходимости хранить на своем компьютере данные системы, они хранятся на сервере разработчика и доступны через сеть. Большинство СПС имеют версии, доступные через глобальную сеть Интернет. Основное преимущество работы с такими версиями заключается в том, что пользователь всегда имеет доступ к самым последним данным. Также при таком подходе упрощается процесс предоставления пользователю бесплатной демонстрационной версии системы, которая имеет ограниченный набор функций и документов, но может дать представление о системе.

Очевидным недостатком СПС, расположенной на сервере разработчика, является невозможность работы с ней при отсутствии связи с сервером, что может быть вызвано как перебоями в работе сервера, например из-за большого числа одновременных обращений, так и неполадками в линиях связи. Еще одним минусом СПС с удаленным доступом является то, что не все сервисные функции системы в настоящий момент могут быть представлены в такой версии системы.

При работе с локальной версией этих проблем не существует, но возникает необходимость постоянной поддержки информационной базы в актуальном состоянии. Обновление информации локальной базы может происходить как с помощью Интернета, так и с помощью традиционных носителей информации — дискет, компакт-дисков и т. д.

12.2.5. Справочная правовая система «Консультант»

Формирование запроса и обработка списка найденных документов

Основной этап работы с любой справочной правовой системой — поиск необходимых пользователю документов. Во многом эффективность поиска зависит от возможностей системы по составлению запроса — поискового предписания. В СПС «Консультант» для составления запроса используется Карточка реквизитов (рис. 12.4).

Для формирования запроса с помощью карточки реквизитов пользователь должен заполнить реквизиты тех полей, по которым будет производиться поиск. Для каждо-

го поля в системе «Консультант» составлен словарь из возможных для этого реквизитов. Этот словарь корректируется при изменении состава информационной базы.

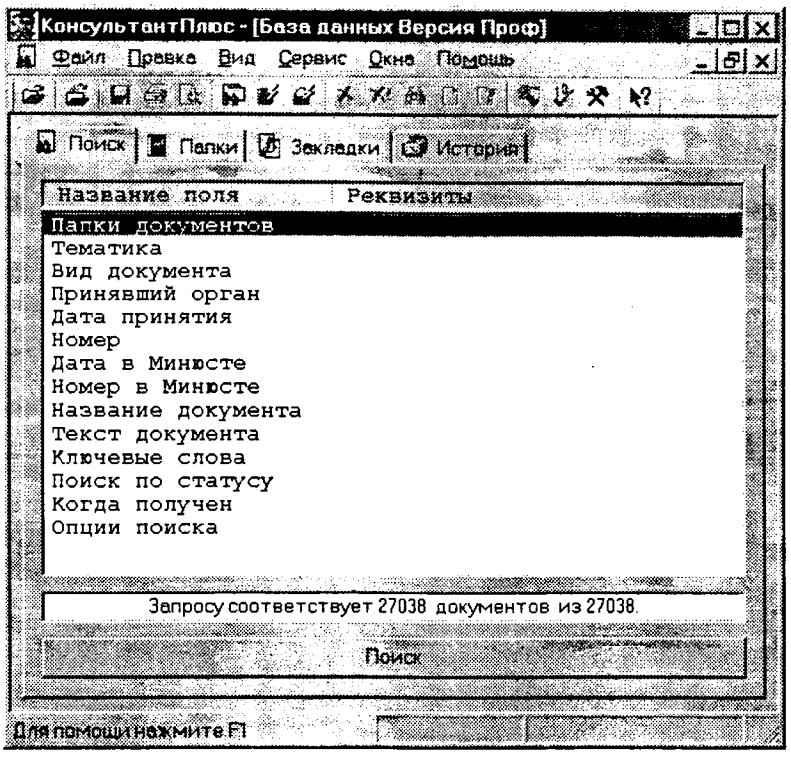


Рис. 12.4. Карточка реквизитов

Реквизиты в одном поле Карточки запросов могут соединяться логическими операторами «И», «ИЛИ», «КРОМЕ», «РЯДОМ». Использование оператора «И» означает, что данное поле искомого документа должно содержать все выбранные значения реквизитов. Использование оператора «ИЛИ» означает, что документ может соответствовать хотя бы одному из перечисленных значений. Оператор «КРОМЕ» означает, что документ не должен соответствовать выбранному условию. Оператор «РЯДОМ» используется при заполнении поля Текст документа и означает, что перечисленные слова должны в тексте документа находиться в пределах, заданных пользователем.

Результатом поиска является список документов, удовлетворяющих запросу. Этот список содержит информацию об основных реквизитах найденных документов, что облегчает выбор необходимого документа. Также для этого списка предусмотрена возможность сортировки, что позволяет, например, начать просмотр документов с имеющих наибольшую дату принятия.

СПС «Консультант» позволяет производить поиск по всем информационным базам, установленным в системе. При этом список найденных документов составляется отдельно для каждой базы.

Папки документов и история запросов

Для повышения эффективности работы пользователя ему может быть предоставлена возможность хранить результаты своей работы с информационной базой. Основные средства для этого — *папки документов* и *история запросов*.

Папки документов используются для формирования подборки документов, наиболее часто использующихся конкретным пользователем. Пользователь может создавать, переименовывать и удалять эти папки.

Если поисковый запрос к системе был выполнен, то существует возможность запомнить Карточку реквизитов, по которой был сформирован этот запрос. В СПС «Консультант» существует возможность хранить 100 Карточек реквизитов для последующего обращения к ним.

12.2.6. Справочно-правовая система «Гарант»

Основное меню

Основное меню системы «Гарант» предназначено для выбора одного из основных действий с информационной базой. Это работа с классификаторами информации, поиск документов, обновление базы. Основное меню имеет различные виды — для Юриста, Эксперта, Руководителя, Менеджера, Бухгалтера, каждый из которых может быть выбран пользователем. Эти виды различаются набором разделов в информационном блоке правовой информации.

Основное меню содержит четыре блока:

- **блок правовой информации.** В этом блоке содержится полный набор правовой информации. В разделе «Документы» этого блока содержатся все документы, хранящиеся в используемой информационной базе;
- **блок экономической информации.** Этот блок содержит информацию, интересную прежде всего экономисту и бухгалтеру. В этот блок включены «Налоговый календарь», информирующий о сроках платежей, информация о количестве рабочих дней в отчетном периоде и т. п.;
- **блок поиска.** Блок содержит различные средства поиска документов в информационной базе;
- **блок обновления.** В этом блоке пользователь может получить статистическую информацию об используемой информационной базе и при необходимости обновить базу.

Поиск документов

Пользователь имеет возможность выбрать между несколькими видами поиска, представленными в системе, в зависимости от информации об искомом документе. В системе «Гарант» предоставлено четыре вида поиска:

- *поиск по реквизитам* — поиск с указанием реквизитов документа;
- *поиск по ситуации* — поиск с использованием ключевых слов, описывающих ситуацию, в которой необходим документ;

- *поиск по классификаторам* — используется для подборки документов по определенной тематике;
- *поиск по источнику опубликования* — позволяет искать документы с известным источником публикации.

Рассмотрим подробнее каждый из этих видов поиска.

Поиск по реквизитам

Такой поиск обычно используется, если нам заранее известны такие реквизиты документа, как вид, номер, дата принятия, орган, принявший документ, фрагмент текста документа и т. п. (рис. 12.5). Условия для поиска можно объединять логическими операторами. Для последующего быстрого использования сложного запроса он может быть сохранен.

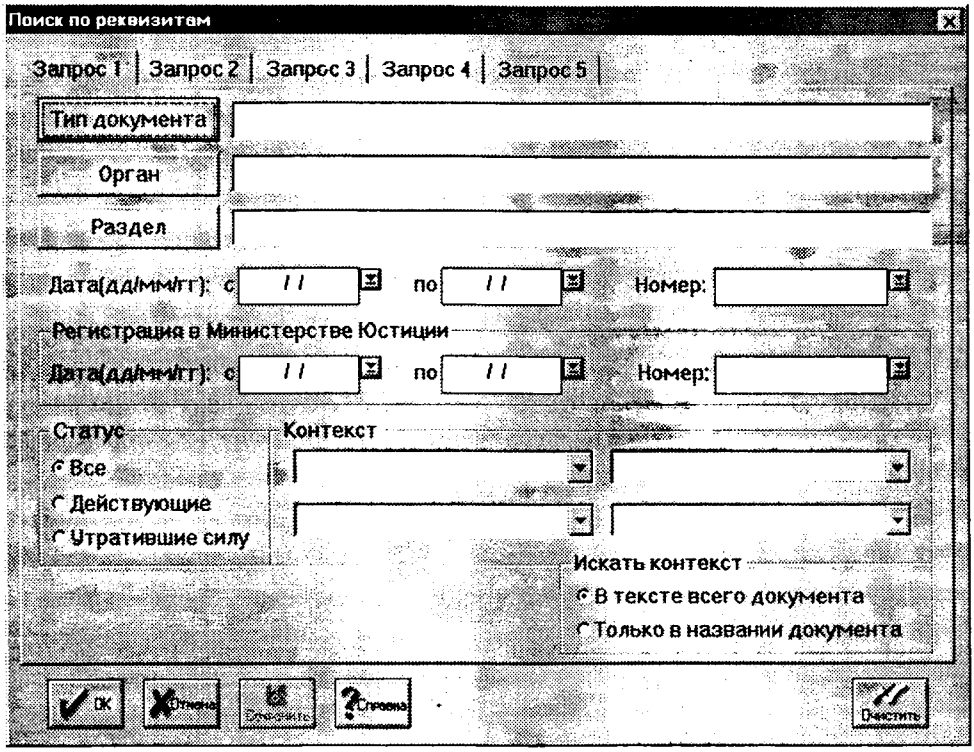


Рис. 12.5. Поиск по реквизитам

Поиск по ситуации

Если реквизиты документа неизвестны, возможен поиск по ситуации (рис. 12.6). Для поиска по ситуации используется словарь, составленный из ключевых (наиболее значимых) слов, содержащихся в документах информационной базы. Для последовательной конкретизации запроса словарь разделен на два уровня — основной и подробный, которые представлены в виде дерева.

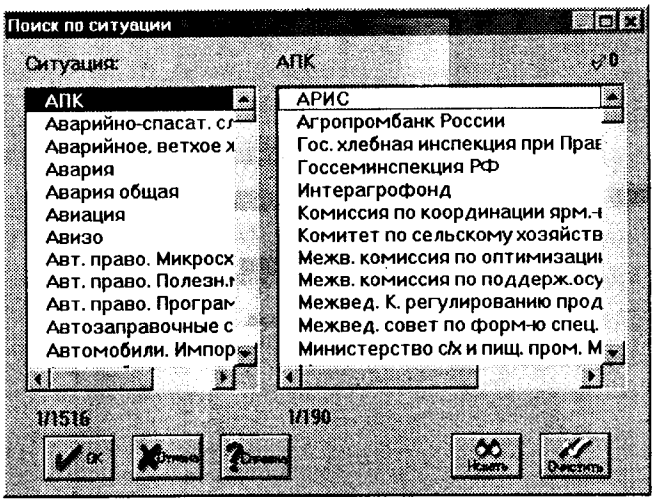


Рис. 12.6. Поиск по ситуации

Процесс формирования запроса при поиске по ситуации заключается в выборе ключевого слова или слов (в запросе они будут соединены логическим оператором «ИЛИ»).

В текущей версии системы возможно сохранить результаты только последнего поиска по ситуации, которые, к тому же, теряются при закрытии программы.

Поиск по классификаторам

Поиск по классификаторам позволяет создавать подборки документов по определенной тематике. Для этого в системе существует пять классификаторов: общий, в котором представлены все документы информационной базы, и специализированные:

- судебная и арбитражная практика;
- международные договоры;
- разъяснения и комментарии;
- проекты законов.

Для поиска документов, соответствующих определенной тематике, можно использовать поиск по реквизитам.

Поиск по источнику опубликования

Такой поиск позволяет искать документы с известным источником опубликования и является аналогом электронного каталога содержания печатных изданий, публиковавших документы, содержащиеся в информационной базе (рис. 12.7).

Списки документов и использование папок

В результате поиска любого представленного в системе вида формируется список документов, соответствующих запросу. Этот список содержит основные реквизиты документа и может быть по ним отсортирован.

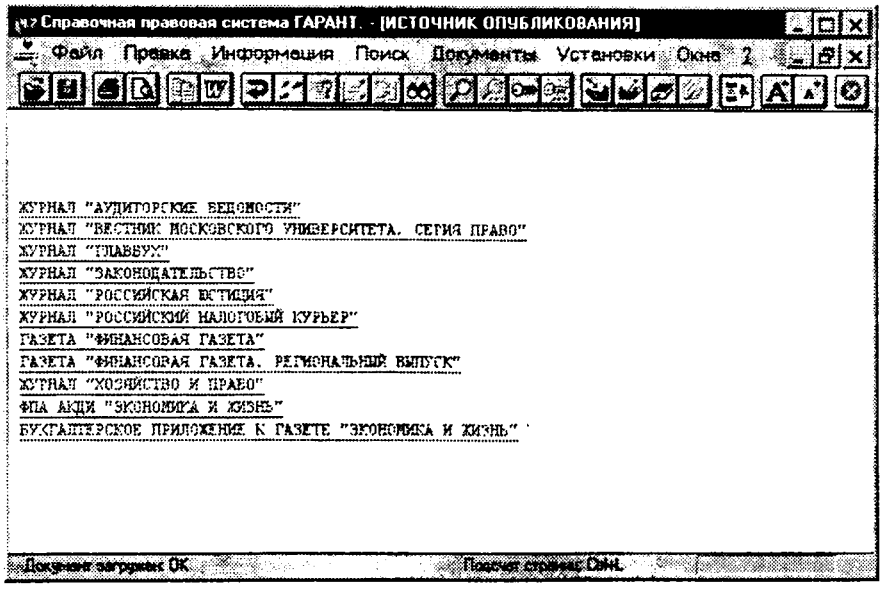


Рис. 12.7. Поиск по источнику опубликования

Списки, полученные в результате поиска, можно сохранить в папке для последующего обращения к ним. В системе «Гарант» реализована многоуровневая система папок, к которым, к тому же, можно применять операции объединения и пересечения. Папки, сформированные пользователем, сохраняются после закрытия программы.

12.3. Интегрированная библиотечно-информационная система «Ирбис»

12.3.1. Описание системы

Интегрированная библиотечно-информационная система «Ирбис» предназначена для автоматизации основных библиотечных функций. Работа системы может быть реализована как в локальной сети, так и в сети Интернет. Система представляет собой совокупность взаимосвязанных автоматизированных рабочих мест (АРМ) следующих типов:

- Администратор — для работы с базами данных системы в целом (обеспечение сохранности и поддержания в актуальном состоянии);
- Комплектатор — программа, позволяющая выполнять функции по комплектованию и учету фондов библиотеки;

- Каталогизатор — для формирования баз данных электронного каталога (базы данных читателей и базы данных алфавитно-предметного указателя), выполнения поиска в библиографических базах данных с последующей корректировкой, копированием, просмотром и печатью найденных документов;
- Книговыдача —рабочее место библиотечного работника, выполняющего функции по выдаче литературы в соответствии с формируемыми заказами и ее возврату. Заказы, формируемые на АРМах Читатель, автоматически поступают на АРМ Книговыдача.
- Читатель — рабочее место конечного пользователя электронного каталога.

Далее мы рассмотрим только работу АРМ Читатель.

12.3.2. АРМ Читатель, общая характеристика

АРМ Читатель используется для поиска в библиотечных базах данных, просмотра, печати найденных документов и формирования заказа на выдачу найденной литературы.

При запуске программы АРМ Читатель появляется окно регистрации, в которое нужно ввести идентификатор читателя или ФИО (рис. 12.8).

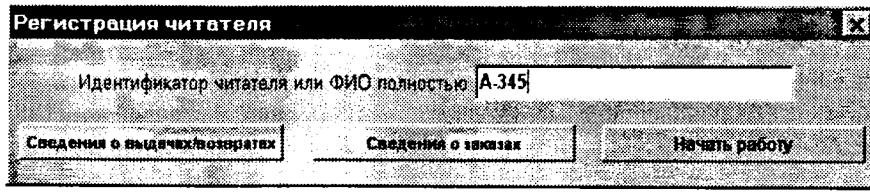


Рис. 12.8. Регистрация читателя

Правильно заполнив карточку, можно приступать к работе с каталогами библиотеки. Вид рабочего окна АРМ Читатель представлен на рис. 12.9.

Функционально интерфейс АРМа состоит из двух основных рабочих плоскостей: Поиск и Просмотр. Как видно из рисунка, работа с базой проста и понятна. Для того чтобы найти интересующую нас книгу или информацию, нужно составить текущий запрос.

12.3.3. Поиск

На рабочем окне Поиск размещены следующие компоненты:

- база данных — выбор той базы данных, в которой предполагается вести поиск;
- вид поиска — выбор элемента библиографического описания и соответствующего словаря, по которому предполагается провести поиск в установленной базе данных;
- словарь — работа со словарем, соответствующим выбранному виду поиска, с целью отбора поисковых терминов для текущего запроса;

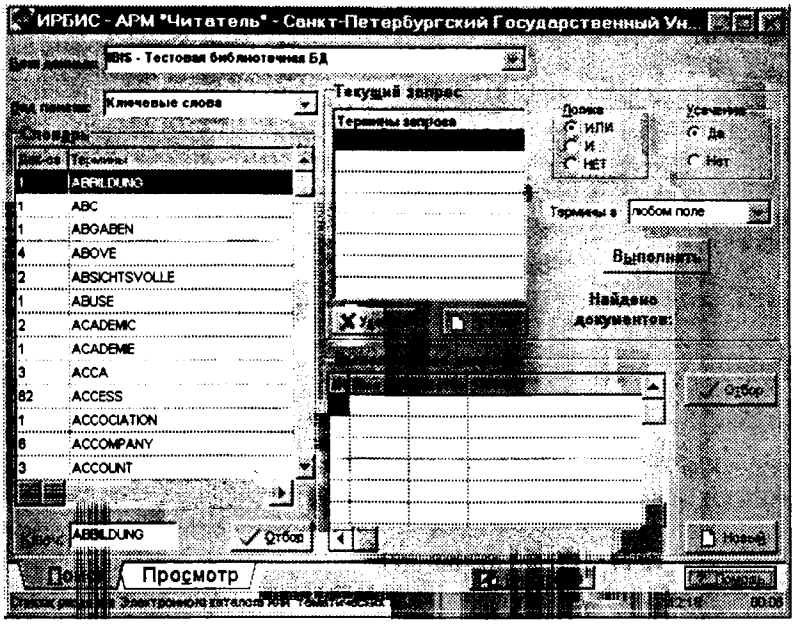


Рис. 12.9. АРМ Читатель, вкладка Поиск

- текущий запрос — формулировка запроса на очередной поиск;
- выполненные запросы — накопление сведений о выполненных запросах и использование их в текущем запросе с целью уточнения или объединения.

Итак, для начала определимся в виде поиска. Он может быть произведен по различным параметрам:

- ключевые слова;
- автор;
- тематический рубрикатор;
- вид, тип документа;
- характер документа;
- заглавие;
- коллектив, мероприятие;
- предметные рубрики;
- географические рубрики;
- страна издания;
- язык;
- год издания;
- журнал за год;
- издающая организация;
- ISBN/ ISSN;

- шифр документа;
- физический носитель информации и др.;

Некоторые из них представлены на рис. 12.10.

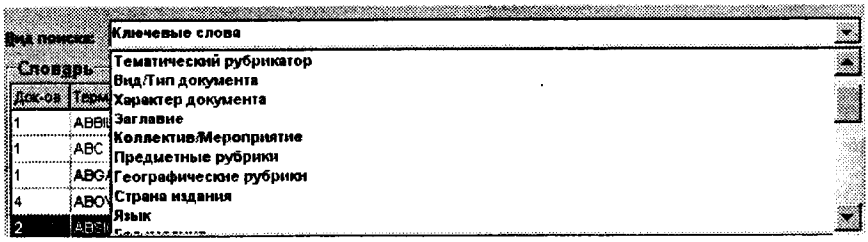


Рис. 12.10. Виды поиска

Естественно, каждый из видов поиска имеет свои особенности. При выборе того или иного вида поиска автоматически формируются словарь, содержащий список ключевых терминов (для данного поиска), а также выводится информация о том, сколько документов содержат соответствующий ключевой термин. На рис. 12.11 представлен в качестве примера словарь тематического рубрикатора.

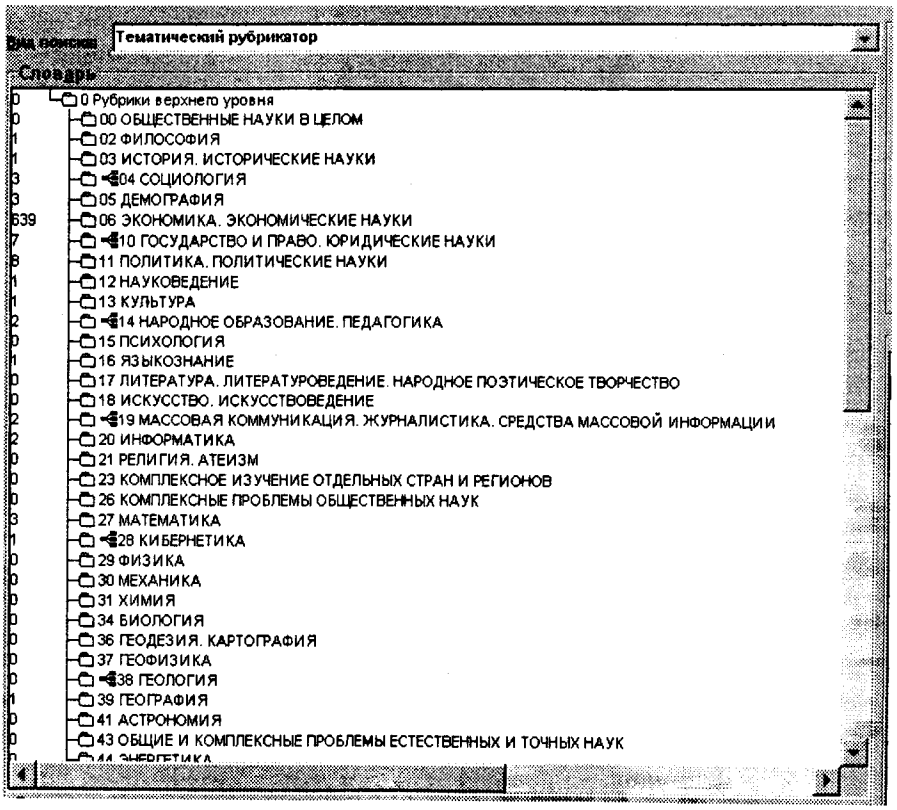


Рис. 12.11. Словарь тематического рубрикатора

По интересующему нас виду поиска из словаря отбираем ключевые слова, составляем запрос и нажимаем кнопку Выполнить. После чего в поле Выполненные запросы отражается наш запрос и количество найденных по нему документов.

12.3.4. Просмотр

На рабочей плоскости Просмотр размещены объекты, реализующие функции просмотра, заказа и печати:

- запрос — выбор того запроса, результаты которого необходимо просмотреть;
- печать — вывод документов на печатающие устройства;
- экспорт — экспорт документов в формат других программных систем;
- список ответов — представление ответов по выбранному запросу в виде списка коротких описаний;
- полное описание — работа с полным описанием выбранного документа.

Итак, на вкладке Просмотр (рис. 12.12) находится список ответов на запрос. Выбрав один из документов, можно посмотреть его полное описание в различных форматах (рис. 12.13).

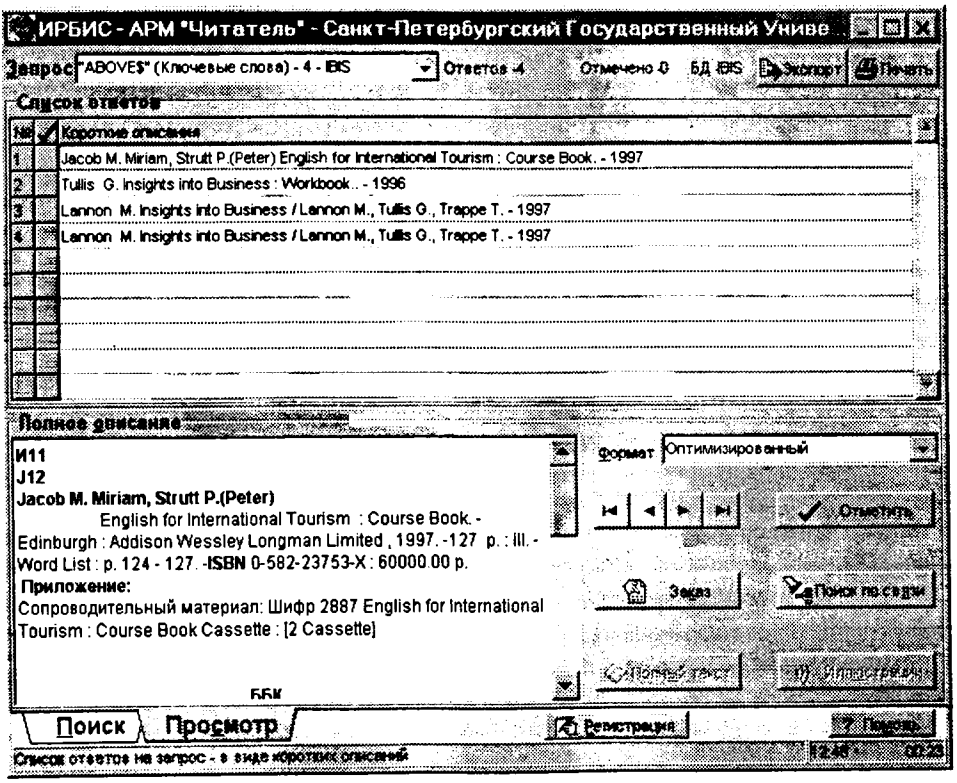


Рис. 12.12. Вкладка Просмотр

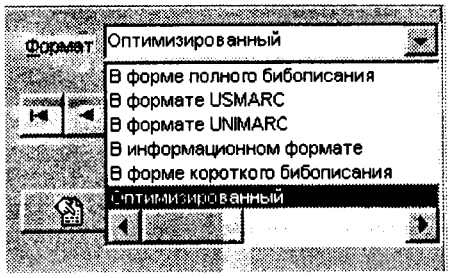


Рис. 12.13. Виды форматов описания документов

Для того чтобы заказать выбранные книги, нужно нажать на кнопку Заказ.

12.4. Глобальные информационные базы данных

12.4.1. История развития рынка информационных услуг

Информационные ресурсы — отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах). Настоящее определение приведено в законе «Об информации информатизации и защите информации».

Понятно, что документы и массивы информации, о которых говорится в этом законе, не существуют сами по себе. В них в разных формах представлены знания, которыми обладали люди, создававшие их. Таким образом, *информационные ресурсы* — это знания, подготовленные людьми и зафиксированные на материальном носителе.

Начало развития рынка информационных услуг приходится на 60-е годы. В то время поставщиками услуг являлись: информационные службы академических, профессиональных и научно-технических обществ, государственных учреждений, учебных заведений. А основными потребителями — ученые и специалисты в области науки и техники.

Параллельно с рынком информационных услуг начал формироваться рынок услуг электронной обработки и передачи информации.

В середине 60-х годов основным видом информационных услуг стали базы данных, которые содержали различные массивы информации по всевозможным темам.

В середине 70-х годов появилась первая глобальная сеть передачи данных. Соответственно стал развиваться новый вид информационных услуг — поиск информации на удаленных серверах, в удаленных базах данных.

С 80-х годов информационная индустрия приобретает все бльшее влияние на экономическую и социальную жизнь общества.

12.4.2. Рынок информационных продуктов и услуг

Для того чтобы использовать информационные ресурсы, нам необходимо знать, где они находятся или как их можно найти, сколько они стоят, кто ими владеет, насколько они доступны и т. п. Для этого существует рынок информационных продуктов и услуг.

Рынок информационных продуктов и услуг — это система экономических, правовых и организационных отношений в сфере торговли продуктами интеллектуального труда.

Как любой другой, информационный рынок имеет свои особенности — это специфичный товар со своими механизмами и условиями продажи. В качестве товара здесь выступают информационные системы, технологии, патенты и лицензии, ноу-хау, товарные знаки, различного рода информация и прочие виды информационных ресурсов.

Основным источником информации для информационного обслуживания в современном обществе являются базы данных. Они интегрируют в себе поставщиков и потребителей информационных услуг, связи и отношения между ними, порядок и условия продажи и покупки информационных услуг.

Стремительному увеличению объемов продажи информационной продукции способствовали широкое внедрение в информационную деятельность вычислительной техники и возможность удаленного доступа к БД по национальным и международным каналам связи и информационным сетям. В первые годы внедрения компьютеров почти вся обработка велась на местах в вычислительных центрах. Пользователи отдаленных районов не могли получать информацию от компьютеров оперативно. Соединение терминалов с компьютерами каналами связи, а также широкое распространение персональных компьютеров позволило удаленным пользователям получать информационные услуги, находясь на значительном расстоянии от поставщиков информации.

С развитием информационных технологий появились новые возможности. Это, в первую очередь:

1) доступ к удаленным базам данных;
2) распределенная обработка информации;
3) электронная почта;
4) видеоконференции.

В настоящее время рынок информационных услуг представляет собой совокупность экономических, правовых и организационных отношений по продаже и покупке информационных услуг, складывающихся между их поставщиками и потребителями.

Поставщиками информационных продуктов и услуг могут быть:

- центры, где создаются и хранятся базы данных;
- центры, распределяющие информацию на основе разных баз данных;
- службы телекоммуникации и передачи данных;
- специальные службы, куда стекается информация по конкретной сфере деятельности для ее анализа, обобщения, прогнозирования, например консалтинговые фирмы, банки, биржи;
- коммерческие фирмы;
- информационные брокеры;

Потребителями информационных продуктов и услуг могут быть различные юридические и физические лица, решающие определенные задачи.

12.4.3. Структура рынка информационных продуктов и услуг

Основными компонентами рынка информационных услуг являются:

- *техническая составляющая* — развивающиеся информационные технологии;
- *нормативно-правовая составляющая* — юридические документы, обеспечивающие правовые нормы на информационном рынке. Среди них могут быть названы:
  - закон «Об информации, информатизации и защите информации»;
  - закон «Об авторском праве и смежных правах»;
  - закон «О правовой охране программ для ЭВМ и баз данных»;
  - закон «О правовой охране топологий интегральных схем».
- *информационная составляющая* — информационно-поисковые системы, помогающие ориентироваться в огромных массивах информации;
- *организационная составляющая* — методы государственного регулирования взаимодействия производителей и распространителей информационных продуктов и услуг.

Инфраструктура информационного рынка — это совокупность секторов, каждый из которых объединяет группу людей или организаций, предлагающих однородные информационные продукты и услуги.

В последние годы сформировалось следующее представление инфраструктуры информационного рынка:

- научно-техническая продукция в виде проектных, технологических, методических разработок по разным отраслям;
- объекты художественной культуры в виде текстовой, визуальной и аудиопродукции;
- услуги образования — все виды обучения;

- управленческие данные и сообщения: политическая и хозяйственная информация, статистические данные, данные о рыночной ситуации, рекламные сообщения, оценки и рекомендации по принятию решений;
- бытовая информация: сообщения общего характера, сведения о потребительском рынке, сведения о рынке труда.

В этой инфраструктуре отсутствуют три основных аспекта, характерных для информационного общества:

- технический — это вся аппаратно-техническая база информатики;
- программный — это все программные продукты информатики;
- коммуникационный — это все виды компьютерных сетей и их возможности по передаче информации, все виды телефонной и факсимильной связи.

Выделим пять секторов рынка информационных продуктов и услуг.

- деловая информация;
- информация для специалистов;
- потребительская информация;
- услуги образования;
- обеспечивающие информационные системы и средства.

Деловая информация

Данный сектор состоит из следующих частей:

- биржевая и финансовая информация — котировки ценных бумаг, валютные курсы, учетные ставки, рынок товаров и капиталов, инвестиции, цены. Поставщиками являются специальные службы биржевой и финансовой информации, брокерские компании, банки;
- статистическая информация — ряды фактических данных и прогнозы социально-экономического и демографического характера. Поставщиками являются государственные службы, компании, консалтинговые фирмы;
- коммерческая информация по компаниям, фирмам, корпорациям, направлениям работы и их продукции, ценам; о финансовом состоянии, связях, сделках, руководителях, деловых новостях в области экономики и бизнеса. Поставщиками являются специальные информационные службы.

Информация для специалистов

Содержит следующие части:

- профессиональная информация — специальные данные и информация для юристов, врачей, фармацевтов, преподавателей, инженеров, геологов, метеорологов и т. д.;
- научно-техническая информация — документальная, библиографическая, реферативная, справочная информация в области естественных, технических,

общественных наук, по отраслям производства и сферам человеческой деятельности;

- доступ к первоисточникам — организация доступа к источникам информации через библиотеки и специальные службы, возможности приобретения первоисточников, их получения по межбиблиотечному абонементу в различных формах.

Потребительская информация

Сектор потребительской информации включает разделы:

- новости и литература — информация служб новостей и агентств прессы, электронные журналы, справочники, энциклопедии;
- потребительская информация — расписания транспорта, резервирование билетов и мест в гостиницах, заказ товаров и услуг, банковские операции и т. п.;
- развлекательная информация — игры, телетекст, видеотекст.

Услуги образования

Данный сектор включает все формы и ступени образования: дошкольное, школьное, специальное, среднее профессиональное, высшее, повышение квалификации и переподготовку. Информационная продукция может быть представлена в компьютерном или некомпьютерном виде: учебники, методические разработки, практикумы, развивающие компьютерные игры, компьютерные обучающие и контролирующие системы, методики обучения и пр. (рис. 12.14).

Обеспечивающие информационные системы и средства

Сектор обеспечивающих информационных систем содержит части:

- программные продукты — программные комплексы с разной ориентацией — от профессионала до неопытного пользователя компьютера: системное программное обеспечение, программы общей ориентации, прикладное программное обеспечение по реализации функций в конкретной области принадлежности, по решению задач типовыми математическими методами и др.;
- технические средства — компьютеры, телекоммуникационное оборудование, оргтехника, сопутствующие материалы и комплектующие;
- разработка и сопровождение информационных систем и технологий — обследование организации в целях выявления информационных потоков, разработка концептуальных информационных моделей, разработка структуры программного комплекса, создание и сопровождение баз данных;
- консультирование по различным аспектам информационной индустрии — какую информационную технику приобретать, какое программное обеспечение необходимо для реализации профессиональной деятельности, нужна ли информационная система, и какая, на базе какой информационной технологии лучше организовать свою деятельность и т. д.;
- подготовка источников информации — создание баз данных по заданной теме, области, явлению и т. п.

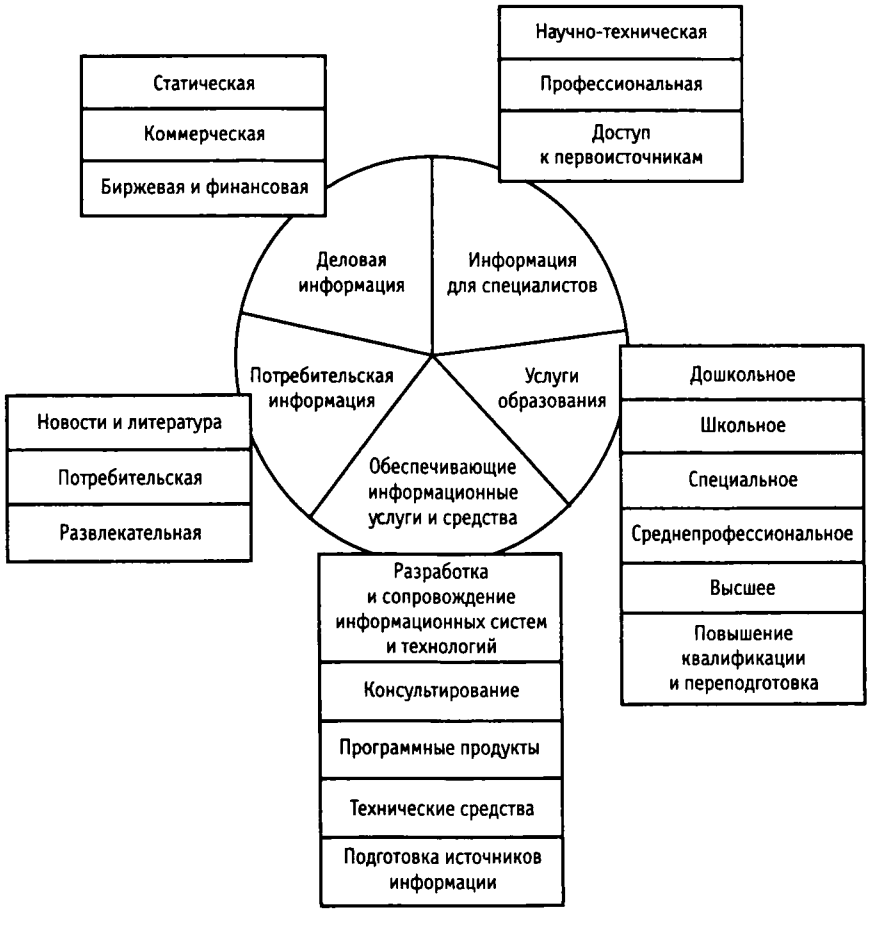


Рис. 12.14. Секторы информационного рынка

В каждом секторе может быть организован любой вид доступа:

- непосредственный — к хранилищу информации на бумажных носителях;
- дистанционный — к удаленным или находящимся в данном помещении компьютерным базам данных.

Информационный рынок, несмотря на разные концепции и мнения относительно его инфраструктуры, существует и развивается, а значит, можно говорить о бизнесе информационных продуктов, услуг, под которым понимается не только торговля и посредничество, но и производство.

Среди фирм, занимающихся производством и торговлей информационными продуктами, могут быть названы:

- Lexis-Nexis;
- Bell & Howell Information and Learning;
- Dow Johns.

Дадим краткую характеристику услуг, предоставляемых первыми двумя фирмами. Информацию о деятельности заинтересованный читатель может найти на ее сайте по адресу www.dowjons.com.

Lexis-Nexis

Основные характеристики фирмы Lexis-Nexis:

- объем базы данных около 23 Тбайт — это более одного миллиарда документов;
- 18 600 информационных источников. Свыше 13 600 полнотекстовых источников информации, включая крупнейшие журналы и газеты (New York Times, Washington Post, Newsweek, Financial Times и др.);
- более 140 информационных агентств (CNN, Agence France Presse, Associated Press, Bloomberg, Business Wire, ITAR-TASS, UPI, Economist Intelligence Unit, Sports Wires);
- данные о десятках миллионов компаний во всем мире — базы Dun&Bradstreet, Standard&Poor, Disclosure, Hoppenstedt, Credireform и др.;
- финансовые отчеты компаний и информация о рынках во всем мире — Investext, Market Research Reports, Predicast, ABI/Inform, ICC, UMI, Country Forecasts, Market Insight и др.;
- законодательство, судебные дела, договоры, юридическая информация по различным странам и регионам (США, страны ЕС, Канада, Китай, Россия и пр.).

Распределение баз данных по тематике:

- 10% — потребительский рынок;
- 12% — законодательство;
- 19% — наука и техника;
- 26% — другие;
- 33% — бизнес.

На рис. 12.15 представлена Web-страничка Lexis-Nexis.

UMI и Bell & Howell Information and Learning

Более 50 лет UMI (University of Microfilm International) обеспечивает библиотеки многих стран мира материалами. Применяя новейшие информационные технологии, она удовлетворяет информационные запросы различных организаций, предлагая широкий спектр традиционных и электронных продуктов и услуг:

- реферативные и полнотекстовые базы данных на CD-ROM;
- диссертации. UMI дает возможность обеспечить тезисами и диссертациями, выносившимися на обсуждение практически во всех университетах Северной Америки. Любую из диссертаций можно получить, приобретя ксерокопию, микроафишу или фильм;

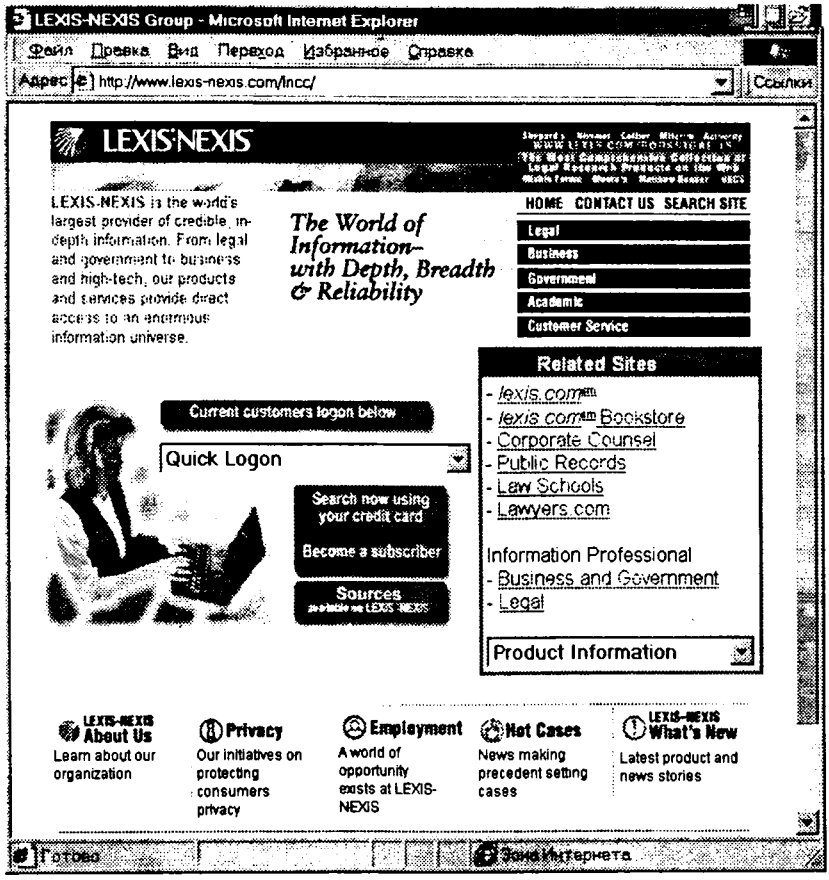


Рис. 12.15. Вид сайта Lexis-Nexis (www.Lexis-Nexis.com)

- сериальные издания на микрофильмах. UMI начала использовать микрофильмы в 1938 году, собрав с тех пор 18 000 названий периодических изданий и 7000 газет, включая многие такие, которые недоступны из других источников;
- репринтные издания;
- собрания исследований.

Реферативные и полнотекстовые базы данных на CD-ROM включают в себя:

- базы данных, содержащие указатели и рефераты статей:
  - ABI/INFORM Ondisc — основные периодические издания по бизнесу и проблемам управления;
  - INSPEC Ondisc — реферативная база данных по физике, электронике и компьютерным технологиям;
  - Periodical Abstracts Ondisc — охватывает основные реферативные издания по всем направлениям, включая социальные и гуманитарные науки, бизнес, общеобразовательные дисциплины, спорт;

- полнотекстовые базы данных, содержащие не только указатели и рефераты статей, но и полный их текст:
  - BusinessLink — полнотекстовая база данных основных 100 журналов по бизнесу и промышленности;
  - Newspapers — тексты наиболее популярных американских газет;
- базы данных, содержащие графику, включают указатели и рефераты статей, а также полное изображение журнальной статьи вместе с иллюстративными материалами:
  - Business Periodicals Ondisk — обширный спектр журналов по бизнесу и проблемам управления, содержащие точные копии журнальных статей;
  - Social Sciences Ondisk — электронные копии статей из 200 наименований по разделу «Социальные науки»;
  - General Periodicals Ondisk — электронные копии большого числа реферативных журналов по 5 основным направлениям: социальные и гуманитарные науки, бизнес и общеобразовательные дисциплины.

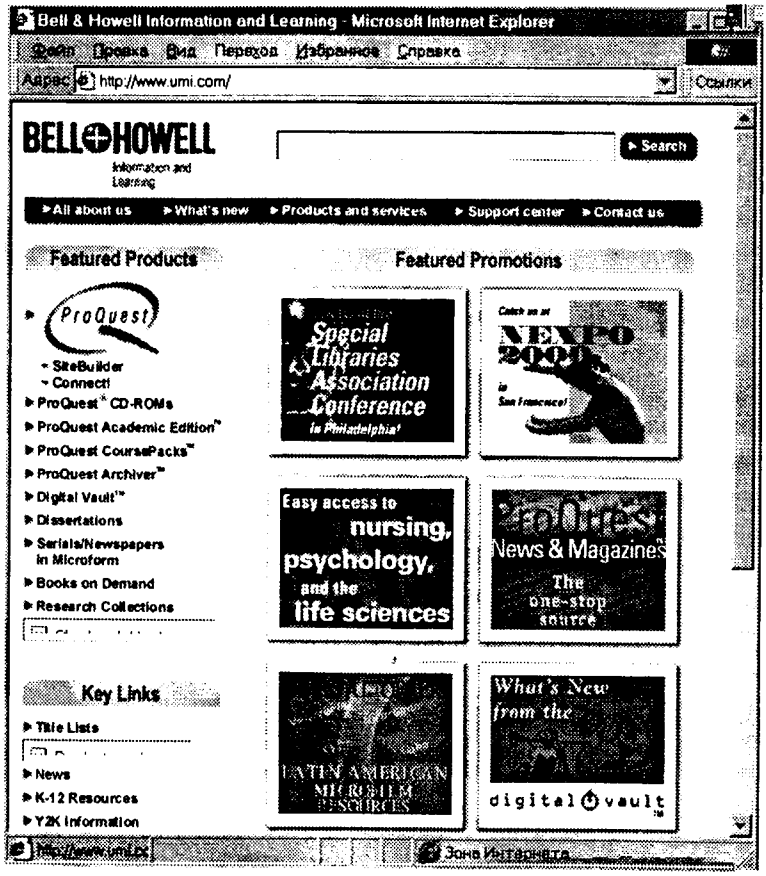


Рис. 12.16. Вид сайта Bell & Howell (www.umi.com)

UMI располагает коллекцией из более 500 собраний исследований, каждое из которых содержит исторические справочные материалы о знаменитых ученых и событиях почти по каждому направлению академических исследований.

UMI обеспечивает доставку материалов. Этим занимается отдел компании UMI InfoStore. Коллекция периодических изданий UMI InfoStore охватывает все направления науки, техники, бизнеса и содержит большое количество журналов другой тематики. Любой документ из этого хранилища может быть извлечен за 24 часа и доставлен.

К настоящему моменту UMI объединилась с информационной компанией Bell & Howell. Web-страничка данной фирмы представлена на рис. 12.16.

Ключевые понятия

- информационно-поисковая система (ИПС);
- указатель;
- словарь;
- термины индексирования;
- информационно-поисковый язык (ИПЯ);
- поисковый образ документа;
- избыточное индексирование;
- справочно-правовые системы;
- информационно-библиотечные системы;
- информационные ресурсы;
- рынок информационных услуг: компоненты и инфраструктура.

Контрольные вопросы

1. Перечислите основные способы организации хранения и поиска документов.
2. Что такое информационно-поисковая система (ИПС)?
3. Что такое избыточное индексирование? Какими преимуществами и недостатками оно обладает?
4. В чем заключена специфика использования ИПС в Интернете?
5. Дайте характеристику термину «справочно-правовая система».
6. Какие параметры определяют качество справочно-правовой системы?
7. Перечислите основные структурные компоненты одной из известных вам справочно-правовых систем.
8. Перечислите основные функции (стандартные возможности) информационно-библиотечных систем.
9. В чем заключается специфика рынка информационных услуг?

10. Из чего складывается инфраструктура информационного рынка?
11. Назовите фирмы, являющиеся ведущими поставщиками информационных услуг на современном этапе.

Литература

1. Васина Н. и др. Информационные ресурсы Internet/РРГУ. М., 1999.
2. Введение в правовую информатику. Справочные правовые системы Консультант Плюс: Учебник для вузов / Под общ. ред. Д. Б. Новикова, В. Л. Камынина. М., 1999.
3. Открытые системы. 1996. № 3, 6, 20, 41.
4. Кулик А. Н. Информационные сети и языковая совместимость дескрипторных информационно-поисковых систем. М.: Сов. радио, 1977.
5. Ланкастер Ф. У. Информационно-поисковые системы. М.: Мир, 1972.
6. Правовая информация: проблемы доступа и управления. Справочник потребителя. М.: Инвента, 1997.

Глава 13. Системы искусственного интеллекта и нейронные сети

Данная глава посвящена вопросам, которые традиционно выходят за рамки базового курса «Экономическая информатика», однако ее содержание будет несомненно полезно для тех читателей, которые захотят более глубоко познакомиться с данным предметом. Основная трудность, которая может возникнуть при работе с излагаемым ниже материалом, состоит в том, что его освоение подразумевает наличие базовых математических знаний. Но даже если читатель на текущий момент таковыми не обладает, то ему все равно рекомендуется ознакомиться с первыми параграфами главы, так как это дает возможность получить представления о тенденциях и перспективах развития информационных дисциплин на современном этапе.

13.1. Искусственный интеллект, основные понятия

Искусственный интеллект — один из разделов информатики, в рамках которого ставятся и решаются задачи аппаратного и программного моделирования тех видов человеческой деятельности, которые традиционно считаются интеллектуальными (творческими).

Термин «искусственный интеллект» (artificial intelligence) предложен в 1956 году на семинаре с аналогичным названием, который состоялся в США и был посвящен решению логических задач.

Современные интеллектуальные информационные технологии — технологии обработки информации и решения задач с помощью вычислительных машин, опирающиеся на достижения в области искусственного интеллекта.

Результаты исследований по искусственному интеллекту используются в *интеллектуальных системах (ИС)* — технических или программных системах, способных решать задачи, считающиеся творческими, принадлежащие конкретной предметной области, знания о которой хранятся в памяти интеллектуальной системы. Системы искусственного интеллекта состоят из трех основных блоков: базы знаний, решателя и интеллектуального интерфейса. Типичным представителем систем искусственного интеллекта являются экспертные системы.

ПРИМЕЧАНИЕ

Решатель — система, способная благодаря встроенной в нее общей стратегии нахождения решения (например, путем логического вывода) находить решение задач. Интеллектуальный интерфейс — интерфейс, в который включены средства, позволяющие человеку вести общение с ЭВМ, не используя для ввода специальные программы.

В целом системы искусственного интеллекта ориентированы на решение большого и очень важного класса задач, называемых неформализуемыми (трудно формализуемыми), к которым относят задачи, обладающие одной или несколькими из следующих особенностей (свойств): алгоритмическое решение задачи неизвестно (хотя, возможно, и существует) или не может быть использовано из-за ограниченности ресурсов ЭВМ; задача не может быть определена (задана) в числовой форме (требуется символьное представление); цели решения задачи не могут быть выражены в терминах точно определенной целевой функции; большая размерность пространства решения; динамически изменяющиеся данные и знания. Как правило, трудно формализуемые задачи обладают неполнотой, неоднозначностью и/или противоречивостью исходных данных и знаний о предметной области.

В исследованиях по искусственному интеллекту можно выделить два основных направления:

1. **Программно-прагматическое** («не имеет значения, как устроено «мыслящее» устройство, главное, чтобы на заданные входные воздействия оно реагировало, как человеческий мозг») — занимается созданием программ, с помощью которых можно решать те задачи, решение которых до этого считалось исключительно прерогативой человека. Сюда относятся распознающие и игровые программы, программы для решения логических задач, поиска, классификации и т. п. Это направление ориентировано на поиски алгоритмов решения интеллектуальных задач на существующих моделях компьютеров.
2. **Бионическое** («единственный объект, способный мыслить — это человеческий мозг, поэтому любое «мыслящее» устройство должно каким-то образом воспроизводить его структуру») — занимается проблемами искусственного воспроизведения тех структур и процессов, которые характерны для живого человеческого мозга и которые лежат в основе процесса решения задач человеком. В рамках бионического подхода к проблеме искусственного интеллекта сформировалась новая наука *нейроинформатика*, практическим выходом которой явилась разработка *нейрокомпьютера* — вычислительной машины VI поколения.

В настоящее время традиционным (классическим) принято считать программно-прагматическое направление, при котором не ставится вопрос об адекватности используемых структур и методов тем, которыми пользуется в аналогичных случаях человек, а рассматривается лишь конечный результат решения задачи. В рамках этого направления сначала велись поиски моделей и алгоритма человеческого мышления. Ни одна из существующих наук (философия, психология,

лингвистика) не смогла предложить такого алгоритма. Тогда специалисты в области искусственного интеллекта предложили собственные модели:

- модель лабиринтного поиска. Этот подход представляет задачу как некоторый граф[1], отражающий пространство состояний, и в этом графе проводится поиск оптимального пути от входящих данных к результирующим;
- эвристическое программирование. *Эвристика* — правило, теоретически не обоснованное, но позволяющее сократить количество переборов в пространстве поиска;
- использование методов математической логики. На основе метода резолюций, позволившего автоматически доказывать теоремы при наличии исходных аксиом, в 1973 г. был создан язык *Пролог.*

Существенный прорыв в практических приложениях искусственного интеллекта произошел в середине 70-х годов, когда на смену поискам универсального алгоритма мышления пришла идея моделировать конкретные знания специалистов-экспертов. В США появились первые коммерческие системы, основанные на знаниях, — *экспертные системы.* Сформировался новый подход к решению интеллектуальных задач — представление и использование знаний.

13.2. Знания и модели их представления

Переход от данных к знаниям — логическое следствие развития и усложнения информационно-логических структур, обрабатываемых на ЭВМ. Понятие *знание* не имеет какого-либо исчерпывающего определения.

Существует множество способов определять понятия. Один из широко применяемых способов основан на идее *интенсионала. Интенсионал понятия* — это определение его через понятие более высокого уровня абстракции с указанием специфических свойств. Другой способ определяет понятие — через перечисление понятий более низкого уровня иерархии или фактов, относящихся к определяемому понятию. Это есть определение через данные, или *экстенсионал* понятия. Другими словами, *интенсионал* — это те общие понятия и отношения, которые характеризуют множество объектов, предметов, явлений. *Экстенсионал* — конкретные характеристики каждого элемента этого множества понятий и отношений.

ПРИМЕР

Рассмотрим понятие *персональный компьютер.* Его интенсионал: персональный компьютер — это ЭВМ, которую можно поставить на стол и купить менее чем за $3000. Экстенсионал этого понятия: персональный компьютер — это IBM PC, Macintosh и т. п.

[1] Напомним, что граф является фундаментальным понятием современной математики. *Графом (ориентированным графом)* называют тройку множеств (I, D, G), в которой непустое множество I — множество вершин, D — множество дуг и G — отображение, которое каждой дуге из D ставит в соответствие упорядоченную пару вершин из I. Визуально граф может быть представлен в виде объекта, в котором точки (вершины) соединены стрелками.

Знания можно определить как совокупность сведений, образующих целостное описание, соответствующее некоторому уровню осведомленности об описываемом вопросе, предмете, проблеме и т. д. Иными словами, *знания* — это выявленные закономерности предметной области (принципы, связи, законы), позволяющие решать задачи в этой области. *С точки зрения искусственного интеллекта знания можно определить как формализованную информацию, на которую ссылаются в процессе логического вывода.*

Для хранения данных используются *базы данных*[1] (для них характерны большой объем и относительно небольшая удельная стоимость информации), для хранения знаний — *базы знаний* (небольшого объема, но исключительно дорогие информационные массивы). *База знаний* — это совокупность знаний, описанных с использованием выбранной формы их представления. База знаний является основой любой интеллектуальной системы.

ПРИМЕЧАНИЕ

Базы данных фиксируют экстенсиональную семантику заданной проблемной области, состояние конкретных объектов, конкретные значения параметров для определенных моментов времени и временных интервалов. База знаний определяет интенсиональную семантику моделей и содержит описание абстрактных сущностей: объектов, отношений, процессов. *Абстрактная сущность* — это понятие об обобщенном представителе некоторого класса объектов, высказывание о свойствах или отношениях между абстрактными объектами, это процедуры, задаваемые в терминах формальных параметров.

Если рассматривать знания с точки зрения решения задач в некоторой предметной области, то их удобно разделить на две большие категории — *факты* и *эвристику*. Первая категория указывает обычно на хорошо известные в данной предметной области обстоятельства, поэтому знания этой категории иногда называют текстовыми, подчеркивая их достаточную освещенность в специальной литературе или учебниках. Вторая категория знаний основывается на собственном опыте специалиста (эксперта) в данной предметной области, накопленном в результате многолетней практики.

Знания можно разделить на процедурные и декларативные. Исторически первыми был *процедурные знания*, то есть знания, «растворенные» в алгоритмах. Они управляли данными. Для их изменения требовалось изменять программы, однако с развитием искусственного интеллекта приоритет данных постепенно изменился и все большая часть знаний сосредоточивалась в структурах данных (таблицы, списки, абстрактные типы данных), то есть увеличивалась роль декларативных знаний. Другими словами, произошел перенос центра тяжести с машинного представления процедур на машинное представление знаний.

Традиционно структуры данных понимаются как декларативные знания, несущие только функцию отображения предметной области. Над структурами данных может осуществляться упорядоченная последовательность операций — программа (процесс), реализующая некоторый алгоритм. Результатом работы програм-

[1] Напомним, что теория организации и управления базами данных, а также соответствующее программное обеспечение были рассмотрены в главе 7.

мы всегда является декларативное знание, а сама программа представляет собой процедурное знание.

Декларативные знания — это совокупность сведений о качественных и количественных характеристиках конкретных объектов, явлений и их элементов, представленных в виде фактов и эвристик. Традиционно такие знания накапливались в виде разнообразных таблиц и справочников, а с появлением ЭВМ приобрели форму информационных массивов (файлов) и баз данных. Декларативные знания часто называют просто данными. Декларативные знания хранятся в памяти ИС так, что они непосредственно доступны для использования. В виде декларативного знания записывается информация о свойствах предметной области, фактах, имеющих в ней место, и тому подобная информация.

Процедурные знания хранятся в памяти ИС в виде описаний процедур, с помощью которых их можно получить. В виде процедурных знаний обычно описывается информация о предметной области, характеризующая способы решения задач в этой области, а также различные инструкции, методики и тому подобная информация. Другими словами, процедурные знания — это методы, алгоритмы, программы решения различных задач, последовательности действий (в выбранной проблемной области) — они составляют ядро баз знаний. Например, в продукционных моделях — это множество продукционных правил вида «ЕСЛИ—ТО»; в производственной сфере аналог процедурных знаний — технологические знания о способах организации и осуществления разнообразных производственных процессов. Процедурные знания образуются в результате осуществления процедур (алгоритмов, программ, аналитических преобразований и т. п.) над фактами как исходными данными.

Дальнейшее развитие структур данных в рамках исследований по искусственному интеллекту привело к появлению специальных структур данных: фреймов, семантических сетей, продукций, названных знаниями. С появлением систем, основанных на знаниях, знаниями считаются предложения, записанные на языках представления знаний, приближенных к естественному и понятных неспециалистам. В семантическом плане обработка информации получает новую окраску, связанную уже с представлением и обработкой знаний, с получением требуемых знаний, но не с процессом.

Одной из наиболее важных проблем, характерных для систем искусственного интеллекта, является представление знаний. Это объясняется тем, что форма представления знаний оказывает существенное влияние на характеристики и свойства системы. Для того чтобы манипулировать всевозможными знаниями из реального мира с помощью компьютера, необходимо осуществлять их моделирование.

Проблема представления знаний — это проблема представления взаимосвязей в конкретной предметной области в форме, понятной системе искусственного интеллекта. *Представление знаний* — это их формализация и структурирование (в целях облегчения решения задачи), с помощью которых отражаются характер-

ные признаки знаний: внутренняя интерпретируемость, структурированность, связность, семантическая метрика и активность. Другими словами, представление знаний — это соглашение о том, как описывать реальный мир. В рамках этого направления решаются задачи, связанные с формализацией и представлением знаний в памяти интеллектуальной системы с помощью специально разработанных моделей представления знаний.

В отличие от методов представления данных, базирующихся на строгих алгоритмах, модели представления знаний имеют дело с информацией, получаемой от специалиста в конкретной предметной области (эксперта), которая часто носит качественный и даже противоречивый характер. Тем не менее, подобная информация должна быть приведена к формализованному виду. Это осуществляется использованием различных методов и приемов, в частности, на основе идей многозначной логики, теории нечетких множеств и других математических моделей.

При работе со знаниями используются два основных подхода:

1. *Логический* (*формальный*) подход, при котором основное внимание уделяется изучению и применению теоретических методов представления знаний, формализации, а также логической полноте (например, создание моделей представления знаний на основе некоторых логических исчислений).
2. *Эвристический* (*когнитивный*) подход, который ориентируется на обеспечение возможностей решения задач. При этом опора делается на принцип организации человеческой памяти и эвристическое моделирование. В отличие от формальных, эвристические модели имеют разнообразный набор средств, передающих специфические особенности той или иной области. Именно поэтому эвристические модели превосходят логические по выразительности и возможности адекватно представить предметную область.

Типичные модели представления знаний: *логические модели*; модели, основанные на использовании правил (*продукционные модели*); *семантические сети*; *фреймовые модели*.

13.2.1. Логические модели

Основная идея подхода при построении *логических моделей* представления знаний состоит в том, что вся информация, необходимая для решения прикладных задач, рассматривается как совокупность фактов и утверждений, которые представляются как формулы в некоторой логике. Знания отображаются совокупностью таких формул, а получение новых знаний сводится к реализации процедур логического вывода.

В основе логических моделей представления знаний лежит понятие формальной теории, задаваемое четверкой: $S = <B, F, A, R>$, где B — счетное множество базовых символов (алфавит), F — множество, называемое формулами, A — выделенное подмножество априори истинных формул (аксиом), R — конечное множество отношений между формулами, называемое правилами вывода.

Достоинства логических моделей представления знаний:

1. В качестве «фундамента» здесь используется классический аппарат математической логики, методы которой достаточно хорошо изучены и формально обоснованы.
2. Существуют достаточно эффективные процедуры вывода, в том числе реализованные в языке логического программирования Пролог.
3. В базах знаний можно хранить лишь множество аксиом, а все остальные знания получать из них по правилам вывода.

Однако действительность не укладывается в рамки классической логики. Так называемая человеческая логика, применяемая при работе с неструктурированными знаниями, — это интеллектуальная модель с нечеткой структурой, и в этом ее отличие от «старой» (классической) логики. Таким образом, логики, адекватно отражающей человеческое мышление, к настоящему времени еще не создано.

13.2.2. Продукционные модели

Психологические исследования процессов принятия решений человеком показали, что рассуждая и принимая решения, человек использует правила продукций, или продукционные правила (от англ. *Production* — правило вывода, порождающее правило).

В общем случае продукционное правило можно представить в следующем виде:

i: S; L; $A \rightarrow B$; Q,

где i — индивидуальный номер продукции; S — описание класса ситуаций, в котором данная структура может использоваться; L — условие, при котором продукция активизируется; $A \rightarrow B$ — ядро продукции, например: «ЕСЛИ A_1, A_2, ..., A_n ТО B» . Такая запись означает, что «если все условия от A_1 до A_n являются истиной, то B также истина» или же «когда все условия от A_1 до A_n становятся истиной, то следует выполнить действие B»; Q — постусловие продукционного правила, описывает операции и действия (процедуры), которые необходимо выполнить после выполнения B. Например, внести изменения в данные либо в саму продукцию.

Пример:

ЕСЛИ	y	является	отцом	x,	(A_1)
	z	является	братом	y,	(A_2)
ТО	z	является	дядей	x,	(B)

В этом случае n=2. При n=0 получаем знания, состоящие только из вывода, то есть простой факт, например, «Атомный вес железа равен **55,8471**».

Суть использования правил продукции для представления знаний состоит в том, что левой части ставится в соответствие некоторое условие, а правой части — действие: ЕСЛИ **<перечень условия>**, ТО **<перечень действий>**. В такой интерпретации левая часть правил оценивается по отношению к базе данных (известному набору фактов) системы, и если эта оценка в определенном смысле соответствует

логическому значению «ИСТИНА», то выполняется действие, заданное в правой части продукции.

В общем случае под условием понимается некоторое предложение — образец, по которому осуществляют поиск в базе знаний, а под действием — действия, выполняемые при успешном исходе поиска, — это могут быть реальные действия, если система управляющая, или заключение — вывод, представляющий собой новое (фактуальное) знание, или некоторая цель.

При использовании продукционной модели база знаний состоит из набора правил. Программа, управляющая перебором правил, называется *машиной вывода*. Механизм выводов связывает знания воедино, а затем выводит из последовательности знаний заключение.

Пример:

Пусть в базе знаний вместе с описанными выше знаниями содержатся еще и такие знания:

ЕСЛИ *z* является отцом *x*,
z является отцом *y*,
x и *y* не являются одним и тем же лицом,
ТО *x* и *y* являются братьями;
где *x*, *y*, *z* – переменные.

Пусть также в базе данных (рабочей памяти) имеются факты:

α является отцом β,
α является отцом γ,
β является отцом δ.

Тогда из этих знаний можно формально вывести заключение, что

γ является дядей δ.

В продукционных системах, основанных на знаниях, процесс обработки информации может осуществляться двумя способами. Первый предполагает обработку информации в прямом направлении (метод сопоставления), когда образцом для поиска служит левая часть продукционного правила — условие, то есть задача решается в направлении от исходного состояния к целевому. Это соответствует стратегии «от данных к цели» или стратегии управления данными. После разрешения возникающих конфликтов выполняются правые части продукционных правил, что соответствует логическому выводу новых утверждений. После добавления выведенных утверждений в базу данных процедура повторяется. Процесс оканчивается, если выполняется продукционное правило, предписывающее прекращение поиска, или в базу данных поступает утверждение, являющееся решением.

При втором подходе обработка информации осуществляется в обратном направлении — метод «генерации» или выдвижения гипотезы и ее проверки (стратегия «от цели к данным»). При каждом обратном движении возникает подцелевое состояние, из которого целевое может быть получено при прямом движении. В этом случае проверяются правые части продукционных правил с целью обнаружить в них искомое утверждение. Если такие продукционные правила существуют, то проверяется, удовлетворяется ли левая часть продукционного правила. Если да, то гипотеза считается подтвержденной, если нет — отвергается.

Таким образом, продукционные правила могут применяться к описанию состояния и описывать новые состояния (гипотезы) или же, напротив, использовать целевое состояние задачи как базу, когда система работает в обратном направлении. При этом продукционные правила применяются к целевому описанию для порождения подцелей (образуют систему редукций).

Пример:

Имеется фрагмент БЗ из двух правил:

$П_1$: **ЕСЛИ** «отдых - летом» и «человек - активный»,
ТО «ехать в горы».

$П_2$: **ЕСЛИ** «любит солнце»,
ТО «отдых летом».

Предположим в систему поступили данные:
«человек - активный» и «любит солнце»

Прямой вывод: исходя из данных, получить ответ.

1-й проход:

Шаг 1. Пробуем $П_1$, не работает -
не хватает данных «отдых - летом».

Шаг 2. Пробуем $П_2$, работает,
в базу поступает факт «отдых - летом».

2-й проход:

Шаг 3. Пробуем $П_1$, работает,
активируя цель «ехать в горы», которая и выступает, например, как совет, который дает система.

Обратный вывод: подтвердить выбранную цель при помощи имеющихся правил и данных.

1-й проход:

Шаг 1. Цель - «ехать в горы»:
пробуем $П_1$ - данных «отдых-летом» нет, они становятся новой целью, и имеется правило, где она в правой части.

Шаг 2. Цель «отдых летом»:
правило $П_2$ подтверждает цель и активизирует ее.

2-й проход:

Шаг 3. Пробуем $П_1$, подтверждается искомая цель.

Свойства продукционных моделей:

- *Модульность* — отдельные продукционные правила могут быть добавлены, удалены или изменены в базу знаний независимо от других; кроме того, модульный принцип разработки (сборки) продукционных систем позволяет автоматизировать их проектирование.
- *Каждое продукционное правило — самостоятельный элемент знаний* (локальный источник знаний); отдельные продукционные правила связаны между собой только через поток данных, которые они обрабатывают.

- *Простота интерпретации* — «прозрачная» структура продукционных правил облегчает их смысловую интерпретацию.
- *Естественность* — знания в виде «что делать и когда» являются естественными с точки зрения здравого смысла.

Недостатки продукционных систем проявляются тогда, когда число правил становится большим и возникают непредсказуемые побочные эффекты от изменения старого и добавления нового правила. Кроме того, затруднительна оценка целостного образа знаний, содержащихся в системе.

Продукционные модели (наряду с фреймами) являются наиболее распространенными средствами представления знаний. Они близки к логическим моделям, что позволяет организовывать на их базе эффективные процедуры вывода, и в то же время более наглядно (чем классические логические модели) отражают знания. Продукционная модель чаще всего применяется в промышленных экспертных системах. Она привлекает разработчиков своей наглядностью, высокой модульностью, легкостью внесения дополнений и изменений и простотой логического вывода.

13.2.3. Семантические сети

Способ представления знаний с помощью *сетевых моделей* наиболее близок к тому, как они представлены в текстах на естественном языке. В его основе лежит идея о том, что вся необходимая информация может быть описана как совокупность троек (arb), где a и b — объекты или понятия, а r — бинарное отношение между ними.

Формально сетевые модели представления знаний могут быть заданы в виде $H = <I, C_1, ..., C_n, \Gamma>$, где I — множество информационных единиц, $C_1, ..., C_n$ — множество типов связей между элементами I, отображение Γ задает между информационными единицами, входящими в I, связи из заданного набора типов связей $\{C_i\}$.

В зависимости от типов связей $\{C_i\}$ различают:

- *Классифицирующие сети* — в них используются отношения структуризации, они позволяют вводить в базы знаний различные иерархические отношения между элементами множества I.
- *Функциональные сети* — вычислительные модели, характеризующиеся наличием функциональных отношений, они позволяют описывать процедуры вычислений одних информационных единиц через другие.
- *Сценарии* — в них используются каузальные отношения (причинно-следственные или устанавливающие влияние одних явлений или фактов на другие), а также отношения типов «средство — результат», «орудие — действие» и т. д.

Если в сетевой модели допускаются связи различного типа, то ее называют семантической сетью.

ПРИМЕЧАНИЕ

Термин «семантическая» означает «смысловая», а сама *семантика* — это наука, устанавливающая отношения между символами и объектами, которые они обозначают, то есть наука, определяющая смысл знаков.

Семантическая сеть — это модель, основой для которой является формализация знаний в виде ориентированного графа с размеченными вершинами и дугами. Вершинам соответствуют объекты, понятия или ситуации, а дугам — отношения между ними. Это наиболее общая модель представления знаний, так как в ней имеются средства реализации всех характерных для знаний свойств: внутренней интерпретации, структурированности, семантической метрики и активности.

Достоинства сетевых моделей: большие выразительные возможности; наглядность системы знаний, представленной графически; близость структуры сети, представляющей систему знаний, семантической структуре фраз на естественном языке; соответствие современным представлениям об организации долговременной памяти человека.

Недостатки: сетевая модель не дает (точнее, не содержит) ясного представления о структуре предметной области, которая ей соответствует, поэтому формирование и модификация такой модели затруднительны; сетевые модели представляют собой пассивные структуры, для обработки которых необходим специальный аппарат формального вывода и планирования. Проблема поиска решения в базе знаний типа семантической сети сводится к задаче поиска фрагмента сети, соответствующего некоторой подсети поставленной задачи. Это, в свою очередь, обусловливает еще один недостаток модели — сложность поиска вывода на семантических сетях.

Еще раз подчеркнем, что сетевые модели являются очень наглядным и достаточно универсальным средством представления знаний. Однако их формализация в конкретных моделях представления, использования и модификации знаний оказывается достаточно трудоемкой, особенно при наличии множественных отношений между ее элементами.

13.2.4. Фреймовые модели

Термин *фрейм* (frame — каркас, рамка) предложен М. Минским в 70-е годы для обозначения структуры знаний для восприятия пространственных сцен. Эта модель, как и семантическая сеть, имеет глубокое психологическое обоснование. Под фреймом понимается абстрактный образ или ситуация. В психологии и философии известно понятие абстрактного образа. Например, слово «комната» вызывает образ комнаты — «жилое помещение с четырьмя стенами, полом, потолком, окнами и дверью». Из этого описания ничего нельзя убрать, например, убрав окна, мы получим уже чулан, а не комнату. Но в нем есть «слоты», или «щели», — незаполненные значения некоторых атрибутов — количество окон, цвет стен, высота потолка, покрытие пола и др. Такой образ и называется фреймом (фреймом минимального описания). Фреймом называется также и формализованная модель этого образа.

Фреймовая модель, основанная на теории М. Минского, представляет собой систематизированную в виде единой теории технологическую модель памяти человека и его сознания. Важным элементом в этой теории является понятие фрейма —

структуры данных для представления некоторого концептуального объекта. Информация, относящаяся к этому фрейму, содержится в составляющих фрейма — слотах. В отличие от моделей других типов, во фреймовых моделях фиксируется жесткая структура, которая называется протофреймом (фреймом-прототипом, или образцом). В общем случае фрейм определяется следующим образом:

$f=[(r_1, v_1,), ..., (r_n, v_n)]$,

где f — имя фрейма; v_i — значение слота,

или

```
(ИМЯ ФРЕЙМА:
                (имя 1-го слота: значение 1-го слота),
                (имя 2-го слота: значение 2-го слота),
                                  . . .
                (имя n-го слота: значение n-го слота)).
```

Значением слота может быть практически что угодно (числа или математические соотношения, тексты на естественном языке или программы, правила вывода или ссылки на другие слоты данного фрейма). В качестве значения слота может выступать набор слотов более низкого уровня, что позволяет во фреймовых представлениях реализовать «принцип матрешки».

В качестве значения слота может выступать имя другого фрейма; так образуются сети фреймов. Все фреймы взаимосвязаны и образуют единую фреймовую структуру, в которой органически объединены декларативные и процедурные знания. Это дает возможность достаточно естественно производить композицию и декомпозицию информационных структур аналогично тому, как это делал бы человек при описании структуры своих знаний.

Кроме фреймов-образцов, или прототипов, хранящихся в базе знаний, различают *фреймы-экземпляры*, которые создаются для отображения реальных ситуаций на основе поступающих данных. При конкретизации (означивании) фрейма ему и слотам присваиваются конкретные имена, и происходит заполнение слотов. Таким образом, из протофреймов получаются фреймы-экземпляры.

Например, структура таблицы, содержащей список работников, записанная в виде протофрейма, имеет вид

```
(СПИСОК РАБОТНИКОВ:
     Фамилия       (значение слота 1);
     Год рождения      (значение слота 2);
     Специальность          (значение слота 3);
     Стаж              (значение слота 4)
).
```

Если в качестве значений слотов использовать реальные данные из таблицы, то получится фрейм-экземпляр.

Важнейшим свойством фреймов является заимствованное из теории семантических сетей наследование свойств. И во фреймах, и в семантических сетях наследование происходит по АКО-связям (от *A Kind Of* = это). Слот АКО указывает на фрейм более высокого уровня иерархии, откуда неявно наследуются, то есть переносятся значения аналогичных слотов, причем наследование свойств может быть частичным.

Фреймовые модели является достаточно универсальными, поскольку позволяют отобразить все многообразие знаний о мире:

- через фреймы-структуры для обозначений объектов и понятий (заем, залог, вексель);
- фреймы-роли (менеджер, кассир, клиент);
- фреймы-сценарии (банкротство, собрание акционеров, празднование именин);
- фреймы-ситуации (тревога, авария, рабочий режим устройства и т. д.).

Основными достоинствами модели фреймов как модели представления знаний являются способность отражать концептуальную основу организации памяти человека, а также естественность, наглядность представления, модульность, поддержка возможности использования значений слотов по умолчанию. Однако фрейм-представление является не конкретным языком представления знаний, а некоторой идеологической концепцией, реализуемой по-разному в различных языках. Теория фреймов послужила толчком к разработке нескольких языков представления знаний, которые благодаря своим широким возможностям и гибкости стали в последние годы довольно распространенными языками. Отметим, кроме того, что концепция объектно-ориентированного программирования может рассматриваться как реальное воплощение понятий, близких фрейму, в традиционных языках программирования.

Основным недостатком фреймовых моделей является отсутствие механизмов управления выводом. Отчасти этот недостаток устраняется при помощи присоединенных процедур, реализуемых силами пользователя системы.

Рассмотренные модели представления знаний во многом близки между собой. По сути, они обладают одинаковыми возможностями описывать и представлять знания. Разница состоит лишь в том, насколько удобно и естественно представлять те или иные знания в виде логических формул, семантических сетей, фреймов или продукций.

Если говорить о взаимосвязях рассмотренных выше моделей, то можно отметить следующее. Логические и продукционные модели отличаются достаточно выраженной процедурной формой, поэтому их часто используют для описания процедурных знаний. Вместо логического вывода, характерного для логических моделей, в продукционных моделях используется вывод, основанный на знаниях. Модели знаний, опирающиеся на семантические сети, используют для описания декларативных знаний.

В системах искусственного интеллекта могут использоваться и несколько моделей представления знаний одновременно. Например, в качестве значений некото-

рых слотов во фрейме могут выступать продукции. Именно смешанные представления оказываются наиболее многообещающими. В продукционных системах используются некоторые элементы логических и сетевых моделей. Поэтому они позволяют организовывать эффективные процедуры вывода (близость к логическим моделям) и наглядно отражать знания в виде сетей (близость к семантическим сетям). В них отсутствуют жесткие ограничения, характерные для логических исчислений, что дает возможность изменять интерпретацию элементов продукции. Кроме того, в результате применения правил вывода к фрагментам сетевого описания происходит трансформация сети за счет смены ее фрагментов, наращивание сети и исключение из нее ненужных фрагментов.

Фрейм можно рассматривать как фрагмент семантической сети, предназначенной для описания объекта (ситуации) проблемной области со всей совокупностью присущих ему свойств. Фреймовый подход к представлению знаний более жесткий, чем основанный на использовании семантической сети. Все то об объекте или ситуации, что важно с позиции решаемой задачи, не «размывается» в сети, а представляется во фрейме. В свою очередь, фрейм можно представить в виде сети, состоящей из вершин и дуг (отношений), так, что «нижние уровни» фрейма заканчиваются слотами, которые заполняются конкретной информацией при вызове фрейма.

Обобщая анализ моделей представления знаний, можно сделать два основных вывода:

- Невозможно дать универсальные рекомендации по выбору модели. Выбор конкретной модели определяется возможностью и удобством представления исследуемой проблемной области с учетом необходимости не только представления, но и использования знаний. Однако чаще используются эвристические, а не логические модели представления знаний.
- Наиболее мощными оказываются смешанные представления.

13.3. Экспертные системы, основные понятия и определения

Современные экспертные системы широко используются для тиражирования опыта и знаний ведущих специалистов практически во всех сферах экономики. Традиционно знания существуют в двух видах – коллективный опыт и личный опыт. Если большая часть знаний в предлагаемой области представлена в виде коллективного опыта (например, высшая математика), эта предметная область не нуждается в экспертных системах. Если в предметной области большая часть знаний является личным опытом специалистов высокого уровня (экспертов), если эти знания по каким-либо причинам слабо структурированы, такая предметная область нуждается в экспертных системах.

Экспертные системы – это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и тиражирующие этот эмпирический опыт для консультаций менее квалифицированных пользователей. На рис. 13.1 изображены основные компоненты экспертной системы.

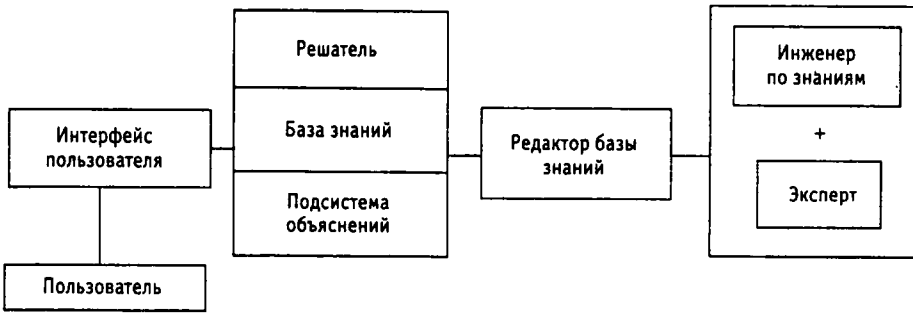


Рис. 13.1. Обобщенная структура экспертной системы

Пользователь — специалист предметной области, для которого предназначена система. Обычно его квалификация недостаточно высока, и поэтому он нуждается в помощи и поддержке со стороны экспертной системы.

Инженер по знаниям — специалист по искусственному интеллекту, выступает в роли промежуточного звена между экспертом и базой знаний (инженер-интерпретатор).

Интерфейс пользователя — комплекс программ, реализующий диалог пользователя с экспертной системой на стадии как ввода информации, так и получения результатов.

База знаний — ядро экспертной системы, совокупности знаний предметной области, записанная на машинный носитель в форме, понятной эксперту и пользователю (обычно на некотором языке, приближенном к естественному). Параллельно такому «человеческому» представлению существует база знаний во внутреннем «машинном» представлении.

Решатель — программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в базе знаний.

Подсистема объяснений — программа, позволяющая пользователю получить ответы на вопросы: «Как была получена та или иная информация?» и «Почему система приняла такое решение?». Ответ на вопрос «Как?» — это трассировка всего процесса получения решения с указанием использованных фрагментов базы знаний, то есть всех шагов цепи умозаключения. Ответ на вопрос «Почему?» — ссылка на умозаключение, непосредственно предшествовавшее полученному решению, то есть отход на один шаг назад.

Интеллектуальный редактор базы знаний — программа, представляющая инженеру по знаниям возможность создавать базы знаний в диалоговом режиме.

Инструментальные средства построения экспертных систем:

- *Традиционные языки программирования.* Наиболее удобными считаются объектно-ориентированные языки (C++, Pascal), это связано с тем, что парадигма объектно-ориентированного программирования тесно связана с фреймовой моделью представления знаний, кроме того, традиционные языки используются для создания других классов инструментальных средств искусственного интеллекта.

- *Языки искусственного интеллекта.* Lisp, Prolog. Универсальность этих языков меньшая, чем у традиционных языков, но это компенсируется богатыми возможностями по работе с символьными и логическими данными, что крайне важно для задач искусственного интеллекта. На основе языков искусственного интеллекта создаются специализированные компьютеры (например, Лисп-машины).
- *«Оболочки»* (shells) — «пустые» версии существующих экспертных систем, то есть готовые экспертные системы без базы знаний. Они вообще не требуют работы программистов для создания готовой экспертной системы. Требуются только специалисты в предметной области для заполнения базы знаний. Однако если некоторая предметная область плохо укладывается в модель, используемую в некоторой оболочке, заполнить базу знаний в этом случае весьма непросто.

ПРИМЕЧАНИЕ

Существующие компьютеры имеют принципы работы, в корне отличные от принципов работы человеческого мозга, поэтому основные надежды на развитие искусственного интеллекта связаны с построением нейрокомпьютеров..

13.4. Нейрокомпьютер и основы нейроинформатики

В 1992 году программа «Пятое поколение компьютеров» была завершена и ее сменила международная программа «Вычисления в реальном мире» (RWC — Real World Computing). В первую очередь речь идет о том, чтобы дать вычислительным и управляющим системам возможность самостоятельно, без помощи «переводчика» — человека воспринимать воздействия внешнего мира и действовать в нем. Авторы программы огромную роль (до 30-40% ее содержания) отводят исследованию естественных и созданию искусственных нейросетевых систем.

Нейробионический подход к проблеме искусственного интеллекта основывается на использовании принципов работы мозга для конструирования интеллектуальных систем. Его привлекательность и перспективность обусловливаются тем, что на функциональном уровне нервная система обеспечивает недоступную (по крайней мере, на текущий момент) для технических устройств способность живых существ адаптироваться в реальном мире, а на «технологическом» уровне — уникальные возможности по быстродействию и надежности.

Имитация работы мозга на ЭВМ (традиционно-последовательной) затруднена принципиальными различиями между конструкциями мозга и ЭВМ. В частности, из-за того, что когда одно устройство моделирует другое, сильно от него отличающееся, процесс моделирования протекает очень медленно. На ЭВМ достаточно просто моделируются формально-логические элементы мышления, а моделирование способности человека (и животных) адаптироваться в изменяющихся и слабо формализованных условиях реального мира сопряжено со значительными сложностями, несмотря на то, что уровень технологии в микроэлектронике позво-

ляет превзойти по плотности упаковки вычислительных элементов и по экономичности энергопотребления нервную ткань. Как раз именно эту возможность адаптироваться к постоянно изменяющимся внешним условиям и необходимо обеспечить системам, претендующим на «интеллектуальность».

В настоящее время сформировалось новое научно-практическое направление — создание нейрокомпьютера, представляющего собой ЭВМ нового поколения, качественно отличающуюся от предыдущих отсутствием заранее созданных алгоритмических программ и способностью к самоорганизации и обучению. Основу нейрокомпьютеров составляют нейронные сети — иерархически организованные параллельные соединения адаптивных элементов — нейронов, которые обеспечивают взаимодействие с объектами реального мира так же, как и биологическая нервная система.

Основные отличия нейрокомпьютера от обычной ЭВМ:

- *параллельная работа* большого числа простых вычислительных устройств обеспечивает огромное быстродействие;
- нейронная сеть *способна к обучению*, которое осуществляется путем настройки параметров сети;
- *высокая отказоустойчивость и помехоустойчивость* сети за счет того, что знания как бы «размыты» в ней и обрыв какой-то связи в общем случае не является достаточным условием отказа, а устранение помех осуществляется за счет «скатывания» поступившего искаженного образа к ближайшему имеющемуся образцу с наименьшим энергетическим уровнем;
- простое строение отдельных *нейронов позволяет использовать новые физические принципы* обработки информации для аппаратных реализаций нейросетей.

ПРИМЕЧАНИЕ

Нейрокомпьютеры создаются для решения определенного фиксированного круга задач. По-видимому, широкое распространение получат устройства, основанные на комбинированных технологиях, включающие по мере необходимости те или иные нейропроцессорные устройства.

В настоящее время дальнейшее повышение производительности компьютеров связывают с системами, обладающими свойствами массового параллелизма. Одна из таких систем — нейрокомпьютер, основу которого составляет искусственная *нейросеть*, реализованная аппаратно на электронных или оптических элементах. В отличие от микропроцессора, имеющего полный набор команд, каждый нейрон, из которых состоит нейросеть, представляет собой лишь простейший аналоговый преобразующий элемент. Однако коллективные свойства сети, содержащей миллионы нейронов, уже не являются тривиальными. Искусственная нейросеть — принципиально параллельная структура, естественным образом реализующая принцип потока данных.

Термин «нейронные сети» сформировался в 40-х годах XX века в среде исследователей, изучавших принципы организации и функционирования биологических нейронных сетей. В настоящее время в области науки *нейроинформатики* разра-

ботан ряд моделей переработки информации, называемых искусственными нейронными сетями или просто нейронными сетями. Обычно под нейронными сетями понимается набор элементарных нейроподобных преобразователей информации — нейронов, соединенных друг с другом каналами обмена информацией для их совместной работы.

Предметом исследования нейроинформатики является решение задач переработки информации с помощью нейросетей в различных предметных областях, особенно в плохо формализуемых, где существующие модели субъективны и неадекватны. Наиболее впечатляющие результаты использования нейросетей достигнуты при распознавании образов, при построении ассоциативной памяти, при создании самообучающихся экспертных систем, при решении оптимизационных задач большой размерности.

Нейроинформатика находится в стадии интенсивного развития. Ежегодно проводится ряд международных конференций по нейросетям, число специализированных периодических изданий более 20. В России в настоящее время издается журнал «Нейрокомпьютер» (Министерство экономики). В 1999 г. в Красноярске состоялся VII Всероссийский семинар «Нейроинформатика и ее приложения».

К настоящему моменту предложено и изучено большое количество моделей нейросетей. Однако основными являются только три принципиально различных типа сетей, большинство остальных распространенных нейросетей состоят из элементов, характерных для сетей трех основных типов:

- сетей прямого распространения (многослойных персептронов);
- полносвязных сетей Хопфилда;
- карт (решеток) Кохонена;

13.5. Нейрон, нейронные сети, основные понятия

Нейрон (*формальный*, *искусственный*) — элементарный преобразующий элемент, как составная часть нейросети, в свою очередь, состоит из элементов трех типов и выполняет две основные функции — взвешенное суммирование и нелинейное преобразование. Элементы нейрона — умножители (синапсы), сумматор и нелинейный преобразователь. Синапсы осуществляют связь между нейронами, умножают входной сигнал (x_j) на число, характеризующее силу связи, — вес синапса (w_j). Сумматор выполняет сложение сигналов, поступающих по синаптическим связям от других нейронов и внешних входных сигналов. Нелинейный преобразователь выполняет нелинейную функцию одного аргумента — выхода сумматора. Таким образом, нейрон в целом реализует скалярную функцию векторного аргумента.

Функционирование нейрона можно разбить на два такта (в соответствии с двумя его функциями):

1. В сумматоре вычисляется величина возбуждения, полученного нейроном, $s = \sum_{j=1}^{n} w_j x_j = (w, x)$ (для простоты смещение w_0 не учитывается).

2. Возбуждение пропускается через преобразующую (активационную) функцию $f(\bullet)$, в результате чего определяется выходной сигнал $y = f(s) = f(\sum_{j=1}^{n} w_j x_j)$.

На рис. 13.2 представлена укрупненная схема нейрона.

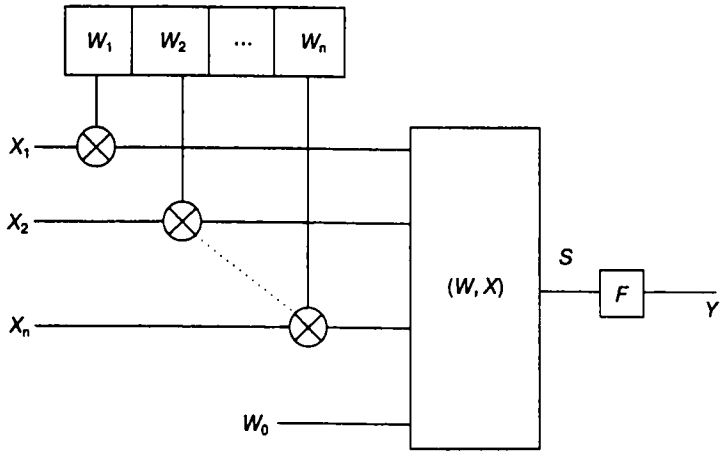


Рис. 13.2. Структурная схема нейрона

Наиболее часто используются следующие функции активации:

1. *Пороговая (ступенчатая) функция* $f(s) = \begin{cases} 1, & s > \theta, \\ 0, & s \le \theta, \end{cases}$ где θ — порог срабатывания нейрона.

ПРИМЕЧАНИЕ

Концептуально каждый нейрон может рассматриваться, как ПРАВИЛО «ЕСЛИ ($\boldsymbol{w}$, $\boldsymbol{x}$) > θ, ТО выдать импульс». Следует отметить, что, например, «небольшая» нейронная сеть для распознавания образов может содержать десятки миллионов нейронов. Обработка базы знаний, содержащей такое число правил, трудно реализуема даже на современных процессорах.

2. *Сигмоидная (логистическая)* функция. Функция называется *сигмоидной*, если она ограничена по минимальному и максимальному значениям и имеет везде положительную производную. Кроме этого предполагается, что функция быстро сходится к верхнему пределу при $s \to +\infty$ и к нижнему при $s \to -\infty$. Например, $f(s) = \dfrac{1}{1+e^{-as}}$, где a — параметр ($a > 0$), $0 < f(s) < 1$.

График сигмоидной функции качественно близок к изображению передаточной характеристики биологического нейрона. Сигмоидная функция приближается к ступенчатой с порогом $\theta = 0$ при $a \to +\infty$. Очевидно, пороговая функция более удобна при аппаратной реализации нейрона, тогда как сигмоидная функция предпочтительна в аналитических исследованиях, поскольку она монотонна,

всюду дифференцируема и имеет непрерывные производные любого порядка. Так, $f' = af(1-f)$.

3. Гиперболический тангенс $f(s) = \text{th}(as) = \dfrac{1-\exp(-as)}{1+\exp(-as)}$, $-1 < f(s) < 1$. Свойство нечетности тангенса, а также то, что $f(0) = 0$, иногда оказывается очень удобным.

Таким образом, каждый нейрон характеризуется вектором весовых множителей и параметрами преобразующей функции. Нейрон способен получать сигналы и в зависимости от их интенсивности и собственных характеристик выдавать выходной сигнал. При этом если выходной сигнал нейрона близок к единице, то говорят, что нейрон возбужден.

Под нейронной сетью обычно понимается структура, состоящая из связанных между собой нейронов, аксиоматическое определение которых было дано ранее. К настоящему времени предложено большое количество способов объединения нейронов в нейросеть (при этом говорят о «топологии» нейросети).

Не ограничивая общности, можно считать, что нейроны в сети расположены слоями. Обычно выделяют входной слой, на который подается возбуждающий сигнал, выходной слой, с которого снимают переработанный сетью сигнал, а все остальные слои называют скрытыми (поскольку они не видны пользователю). Очевидно, что для адекватного решения задачи функционирования сети нужно правильно выбрать значения весов связей между нейронами — обучить сеть.

Возможны несколько *типов классификации* существующих нейросетей. Наиболее часто используются следующие:

- по типу входной информации:
  - сети, анализирующие двоичную информацию;
  - сети, оперирующие с действительными числами;
- по методу обучения:
  - сети, которые для того чтобы удовлетворять поставленным критериям требуют предварительного обучения перед включением их в реальную обстановку, — модели с учителем;
  - сети, не требующие предварительного обучения, способные самообучаться (совершенствовать свои характеристики) в процессе работы;
- по характеру распространения информации:
  - однонаправленные сети, в которых информация распространяется только в одном направлении от одного слоя элементов к другому;
  - рекуррентные сети, в которых выходной сигнал элемента может вновь поступать на этот элемент и другие элементы сети этого же или предыдущих слоев в качестве входного сигнала (сети с обратными связями);

- по способу преобразования входной информации:
  - автоассоциативные;
  - гетероассоциативные.

Они отличаются следующим. На всей сети или на одном ее слое задаются в определенный момент времени исходные состояния нейронов $\mathbf{x}=(x_1, x_2, ..., x_n)$, то есть задается входной образ. Затем этот образ трансформируется в соответствии с функциями преобразования элементов. Информация распространяется по сети, происходит ее обработка. В однонаправленных сетях информация распространяется от слоя к слою, и выходной слой выдает результирующий вектор $\mathbf{y}=(y_1, y_2, ..., y_m)$. (В однослойных сетях происходит стабилизация сети в одном из состояний.) После этого считываются значения узлов сети, представляющих выходной вектор. Возможны два варианта:

- $\mathbf{x} \neq \mathbf{y}$ — гетероассоциативная сеть, осуществляющая отображение входного вектора $\mathbf{x}$ в выходной вектор $\mathbf{y}$;
- входной вектор $\mathbf{x}$ представляет искаженный или сжатый образ эталона $\mathbf{x}_0$. Такая сеть называется автоассоциативной.

Выходной вектор, получающийся в результате работы нейросети, характеризует состояние элементов сети. Смысл, который имеют значения этого вектора, определяется в зависимости от решаемой проблемы.

Решение задачи на нейрокомпьютере принципиально отличается от решения той же задачи на обычной ЭВМ. Решение задачи на обычной ЭВМ заключается в обработке вводимых данных в соответствии с априори разработанной и записанной в нее программой. Для составления программы необходимо разработать алгоритм, то есть определенную последовательность математических и логических действий, необходимых для решения этой задачи. Алгоритмы и программы разрабатываются людьми, а компьютер используется лишь для выполнения большого количества элементарных операций: сложения, умножения, проверки логических условий и т. п.

Нейрокомпьютер же используется как «черный ящик», который можно обучить решению задач из какого-нибудь класса. Нейрокомпьютеру «предъявляются» исходные данные задачи и ответ, который соответствует этим данным. Нейрокомпьютер должен сам построить внутри себя алгоритм решения этой задачи, чтобы выдать ответ, совпадающий с правильным. Естественно ожидать, что чем больше различных пар (исходные данные — ответ) будет предъявлено нейрокомпьютеру, тем более адекватную решаемой задаче модель он сформулирует. Более того, если после этапа обучения нейрокомпьютеру предъявить исходные данные, которых он раньше не встречал, то желательно, чтобы он, тем не менее, выдал правильное решение. В этом заключается способность нейрокомпьютера к обобщению.

Поскольку в основе устройства нейрокомпьютера лежит искусственная нейросеть, то процесс обучения состоит в настройке параметров этой сети. При этом, как правило, топология сети считается неизменной, а к подстраиваемым параметрам обычно относятся параметры нейронов и величины синаптических весов. Обычно в литературе под обучением принято понимать процесс изменения весов связей между нейронами.

Классификация методов обучения нейронных сетей:

- по способу использования учителя:
  - с учителем. Сети предъявляются примеры входных и выходных данных. Сеть преобразует входные данные и сравнивает свой выход с желаемым. После этого проводится коррекция весов с целью получить лучшую согласованность выходов;
  - с последовательным подкреплением знаний. В этом случае сети не дается желаемое значение выхода, а только ставится оценка, хорош выход или плох;
  - без учителя. Сеть сама вырабатывает правила обучения путем выделения оценочных признаков из набора входных данных;
- по использованию элементов случайности:
  - детерминированные методы. В них шаг за шагом осуществляется процедура коррекции весов сети, основанная на использовании их текущих значений, входов сети, выходов нейронов и некоторой дополнительной информации, например значений желаемых выходов сети;
  - стохастические методы. Они основываются на использовании случайных изменений весов в ходе обучения.

13.6. Модели нейронных сетей

13.6.1. Многослойные однонаправленные сети

Такие сети называются также *сетями прямого распространения*, или *многослойными персептронами*. В дальнейшем (там, где это не может привести к неоднозначности) будем называть такие сети многослойными.

Сети этого типа состоят из нескольких слоев нейронов: входного слоя, выходного и нескольких «скрытых слоев». На рис. 13.3 изображена сеть, у которой K слоев. Нейроны каждого слоя не связаны между собой. Выходной сигнал с каждого нейрона поступает на входы всех нейронов следующего слоя. Нейроны входного слоя не осуществляют преобразования входных сигналов, их функция заключается в распределении этих сигналов между нейронами первого скрытого слоя.

Функционирование многослойной сети осуществляется следующим образом: входной сигнал, подаваемый на сеть, поступает на нейроны входного слоя, проходит по очереди через все слои и снимается с выходов нейронов выходного слоя. По мере распространения сигнала по сети он претерпевает ряд преобразований, которые зависят от его начального значения, от преобразующих функций и величин весов связей.

Пусть сеть состоит из K слоев: одного входного, одного выходного и $(K-2)$ скрытых слоев, — каждый слой состоит из $n(k)$ нейронов. Набор выходных сигналов нейронов k-го слоя (k=1:K) обозначим. Далее обозначим w^k набор весов синаптических связей, соединяющих нейроны $k-1$-го слоя с нейронами k-го слоя; w_{ij}^k — вес связи, соединяющий i-й нейрон $k-1$-го слоя с j-м нейроном k-го слоя

($y_j^{k-1} \in y^{k-1}$, $y_j^k \in y^k$). Тогда прямое функционирование сети описывается следующими соотношениями:

- $y^1 = x$;
- $y_j^k = f(\sum_{i=1}^{n(k-1)} w_{ij}^k y_i^{k-1})$, $j=1{:}n(k)$, где y_j^k — выход нейронной сети.

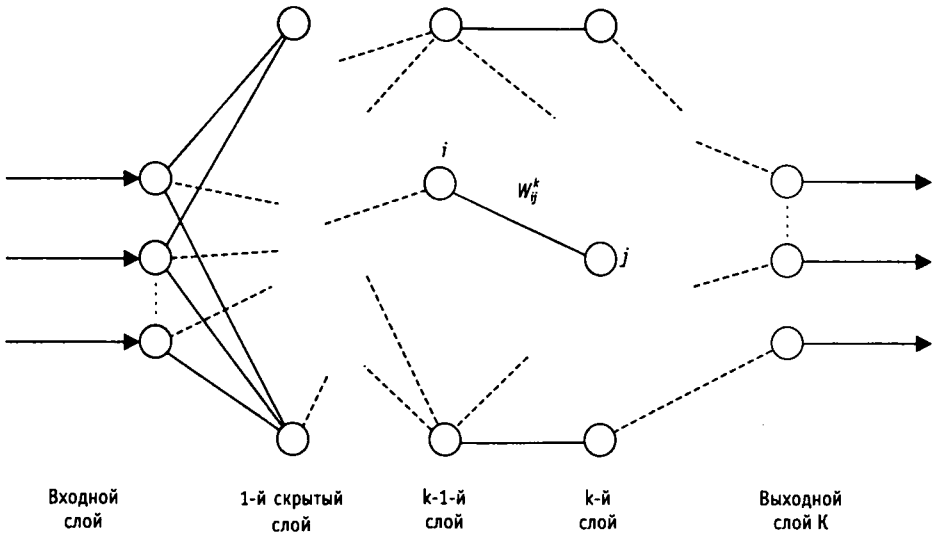


Рис. 13.3. Схема многослойной однонаправленной сети

В основе методов обучения многослойных нейросетей наиболее часто лежит так называемое дельта-правило. Дельта-правило используется при обучении с учителем и реализуется следующим образом:

$$w_{ij}(t+1) = w_{ij}(t) + h x_i (d_j - y_j),$$

где h — параметр (шаг обучения); d_j — эталонное (требуемое) значение выхода элемента.

Таким образом, изменение силы связей происходит в соответствии с ошибкой выходного сигнала $\delta_j = (d_j - y_j)$ и уровнем активности входного элемента x_i.

Обобщение дельта-правила, называемое обратным распространением ошибки (Back — propagation), применимо к сетям с любым числом слоев.

Обучение сети в этом случае состоит из следующих шагов:

1. Выбрать очередную обучающую пару (**x**, **d**). Подать входной вектор на вход сети.
2. Вычислить выход сети **y**.
3. Вычислить разность между выходом сети и требуемым выходом (ошибку).
4. Подкорректировать веса сети так, чтобы минимизировать ошибку.
5. Повторять шаги с 1-го по 4-й для каждой обучающей пары, пока ошибка не достигнет приемлемого уровня.

Ошибка функционирования сети обычно определяется как

$$E = \frac{1}{2}\sum_{j=1}^{n(K)}(d_j - y_j)^2,$$

где $y_j = y_j^K$ — выход сети.

Для уменьшения этой ошибки следует изменить веса сети по правилу $w^k(t+1) = w^k(t+1) - h \cdot \frac{\partial E}{\partial w^k}$. Эта формула описывает процесс градиентного спуска в пространстве весов. Очевидно, для выходного слоя $(k = K)$ $-\frac{\partial E}{\partial w_{ij}^k} = (d_j - y_j) \cdot f_j' \cdot y_i^{k-1}$.

Так как $y_l^{k+1} = f_l(\sum_{j=1}^{n(k)} w_{jl}^{k+1} y_j^k)$, а $y_j^k = f_j(\sum_{i=1}^{n(k-1)} w_{ij}^k y_i^{k-1})$, то для промежуточных (скрытых) слоев (то есть для $k = 2:(K-1)$) $-\frac{\partial E}{\partial w_{ij}^k} = \sum_{l=1}^{n(k+1)}\left[(d_l - y_l) f_l' w_{jl}^{k+1}\right] f_j' y_i^{k-1}$

Если в качестве нелинейной преобразующей функции используется сигмоидная функция, то удобно использовать рекуррентные формулы:

$\delta_j^k = (d_j - y_j) y_j (1 - y_j)$ $(j = 1:n(k))$ при $k = K$ — для выходного слоя;

$\delta_j^k = \sum_{l=1}^{n(k+1)}\left[\delta_l^{k+1} w_{jl}^{k+1}\right] y_j^k (1 - y_j^k)$ $(k = (K-1):2)$ — для скрытых слоев,

тогда $-\frac{\partial E}{\partial w_{ij}^k} = \delta_j^k y_i^{k-1}$ и $w^k(t+1) = w^k(t) + h\delta_j^k y_i^{k-1}$.

Эти соотношения называются формулами обратного распространения ошибки. Если при прямом функционировании входной сигнал распространяется по сети от входного слоя к выходному, то при подстройке весов ошибка сети распространяется от выходного слоя к входному.

Область применения многослойных нейросетей обусловлена тем, что они аппроксимируют отображение $F: D \subset R^{n(1)} \to R^{n(K)}$, используя для этого предварительное обучение на наборах тренировочных данных $(x_1, d_1), (x_2, d_2), ..., (x_L, d_L)$, где $d_c = F(x_c)$. Таким образом, сеть можно рассматривать как модель $\mathbf{y} = \varphi(\mathbf{x})$ реального объекта $\mathbf{y} = F(\mathbf{x})$. Доказаны теоремы о том, что с помощью сети с обратным распространением ошибок можно аппроксимировать любую функцию с любой точностью.

13.6.2. Полносвязные сети Хопфилда

Сеть Хопфилда (представлена на рис. 13.4) — однослойная сеть. Все нейроны связаны друг с другом связями w_{ij}, причем сигнал с выхода нейрона может подаваться на его же вход и необязательно $w_{ij} = w_{ji}$.

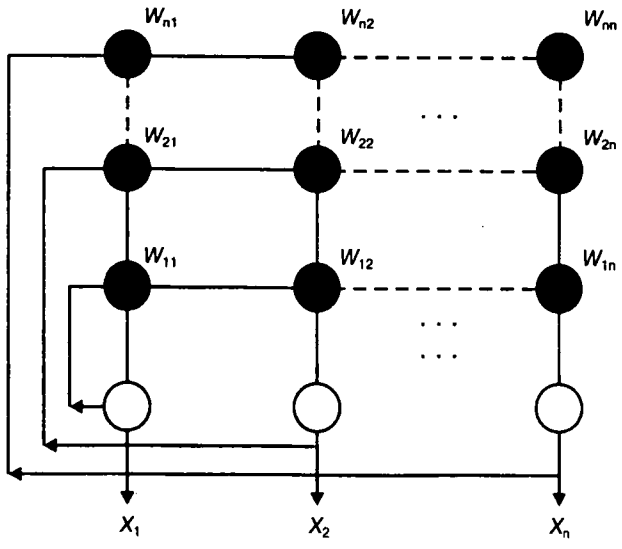


Рис. 13.4. Полносвязная сеть Хопфилда

Поскольку сигнал с выхода каждого нейрона подается на входы всех остальных, входной вектор начинает циркулировать, преобразуясь по сети до тех пор, пока сеть не придет в устойчивое состояние (то есть когда все нейроны на каждом последующем цикле будут вырабатывать тот же сигнал, что и на предыдущем). Очевидно, возможны случаи бесконечной циркуляции входного вектора без достижения устойчивого состояния.

Выберем функцию элементов в виде:

$$x_j(t+1) = \begin{cases} 1, & \sum_{i=1}^{n} w_{ij} x_i(t) > \theta_j, \\ 0\ (-1), & \sum_{i=1}^{n} w_{ij} x_i(t) < \theta_j, \\ x_j(t), & \sum_{i=1}^{n} w_{ij} x_i(t) = \theta_j, \end{cases}$$

Состояние сети — множество текущих значений сигналов x от всех нейронов. Функционирование сети геометрически может быть представлено как движение вектора x, характеризующего состояние сети, на кубе $[0, 1]^n$. Когда подается новый входной вектор, сеть переходит из вершины в вершину, пока не стабилизируется. Устойчивая вершина определяется сетевыми весами, текущими входами и величиной порога. Если входной вектор частично неправилен или неполон, то сеть стабилизируется в вершине, ближайшей к желаемой.

В общем случае все возможные состояния сети образуют некое подобие холмистой поверхности, а текущие состояния сети аналогичны положениям тяжелого

шарика, пущенного на эту поверхность, — он движется вниз по склону в ближайший локальный минимум. Каждая точка поверхности соответствует некоторому сочетанию активностей нейронов в сети, а высота подъема поверхности в данной точке характеризует «энергию» этого состояния. Энергия данного сочетания активностей определяется как сумма весов связей между парами активных нейронов, взятая со знаком минус (при $\theta = 0$).

Таким образом, если связь между двумя какими-то нейронами имеет большой положительный вес, то сочетания, в которых эти нейроны активны, характеризуются низким уровнем энергии — именно к таким сочетаниям и будет стремиться вся сеть. И, наоборот, нейроны с отрицательной связью при активации добавляют к энергии сети большую величину, так что сеть стремится избегать подобных состояний.

Динамику сети Хопфилда удобно описывать так называемой функцией энергии, которая в достаточно общем виде может быть определена как

$$E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij} x_i x_j + \sum_{j=1}^{n} \theta_j x_j .$$

Функция энергии определяет устойчивость сети, другими словами — это функция Ляпунова сети Хопфилда, то есть функция, которая всегда убывает при изменении состояния сети. В конце концов эта функция должна достичь минимума и прекратить изменение, гарантируя тем самым устойчивость сети.

Изменение состояния какого-либо элемента сети всегда вызывает уменьшение энергии сети. Действительно, пусть изменил свое состояние элемент k $(k = 1:n)$, то есть его состояние изменилось с +1 на 0 (или −1) или наоборот, тогда

$$\Delta E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij} x_i(t+1) x_j(t+1) + \sum_{j=1}^{n} \theta_j x_j(t+1) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij} x_i(t) x_j(t) - \sum_{j=1}^{n} \theta_j x_j(t) \Rightarrow$$

$$\Rightarrow \Delta E_k = -x_k(t+1)\sum_{i=1}^{n} w_{ik} x_i(t+1) + \theta_k x_k(t+1) + x_k(t)\sum_{i=1}^{n} w_{ik} x_i(t) - \theta_k x_k(t) \Rightarrow$$

$$\Rightarrow \Delta E_k = (x_k(t) - x_k(t+1))\left(\sum_{i=1}^{n} w_{ik} x_i(t+1) - \theta_k\right) \quad \{x_i(t+1) \equiv x_i(t)\} .$$

Видно, что в результате изменения k-го элемента $\Delta E_k \le 0$, $\Delta E_k = 0$, когда в сети не происходит никаких изменений. Благодаря такому непрерывному стремлению к уменьшению энергия, в конце концов, должна достигнуть минимума и прекратить изменение. По определению такая сеть является устойчивой. Сети Хопфилда называются также сетями, минимизирующими свою энергию.

Сети Хопфилда имеют многочисленные применения. Некоторые из них связаны со способностью этих сетей запоминать, а затем восстанавливать даже по неполной входной информации различные образы. Другие применения связаны с возможностью использования сетей Хопфилда для решения оптимизационных задач.

13.6.3. Двунаправленная ассоциативная память

Основной причиной неудач исследователей в области искусственного интеллекта, потративших свыше 20 лет на безуспешные попытки моделирования интеллектуальной деятельности на базе обычных цифровых ЭВМ, является, по-видимому, тот факт, что в современных ЭВМ существует прямая зависимость времени поиска от количества хранимых образцов. Компьютер запоминает отдельные объекты в отдельных ячейках, как бы заучивает сведения наизусть, и при изучении наук примеры для него, в отличие от человека, отнюдь не полезнее правил.

НейроЭВМ, построенная на базе нейросетей, обладает ассоциативной памятью и классифицирует поступившие образы со скоростью, которая не зависит от количества уже поступивших образцов, — она немедленно связывает новый образ с ближайшим имеющимся.

Память человека является ассоциативной — один предмет напоминает нам о другом, а тот, в свою очередь о третьем и т. д. Наши мысли как бы перемещаются от предмета к предмету по цепочке умственных ассоциаций. Способность к ассоциациям может быть использована для восстановления забытых образов («мы с вами где-то встречались»).

Двунаправленная ассоциативная память является гетероассоциативной; входной вектор поступает на один набор нейронов, а соответствующий выходной вектор вырабатывается на другом наборе нейронов. Как и сеть Хопфилда, двунаправленная ассоциативная память способна к обобщению, вырабатывая правильные реакции, несмотря на возможные искажение входа.

Очевидно, состояние нейронов можно рассматривать, как кратковременную память, так как она может быстро изменяться при появлении другого входного вектора. В то же время значения коэффициентов весовой матрицы образуют долговременную память (ассоциации) и могут изменяться на более длительном отрезке времени, используя соответствующий метод обучения. Обучение производится с использованием обучающего набора из пар векторов x и y. Предположим, что все запомненные образцы представляют собой двоичные векторы.

Решение задачи с помощью двунаправленной ассоциативной памяти можно разбить на два этапа: режим обучения и непосредственно решение (распознавание). Рассмотрим оба эти этапа на примере.

Каждый нейрон a_i в первом слое A имеет синапсы, соединяющие его с нейронами b_j во втором слое B. Пусть нейроны имеют следующий «смысл»: a_1 — валюта, a_2 — доллары, a_3 — марки, a_4 — рубли, b_1 — США, b_2 — Россия, b_3 —Канада, b_4 — Германия.

Режим обучения бинарными образами

Подадим на нейросеть три бинарных связи (x_1, y_1), (x_2, y_2), (x_3, y_3).

Пусть $x_1 = (1, 1, 0, 0) \rightarrow y_1 = (1, 1, 1, 0)$;

$x_2 = (1, 0, 1, 0) \rightarrow y_2 = (0, 1, 0, 1)$;

$x_3 = (1, 0, 0, 1) \rightarrow y_3 = (0, 1, 0, 0)$;

Смысл обучающих связей очевиден: если возбуждены нейроны a_1 и a_2 (в нашем распоряжении есть доллары), то по соответствующим синапсам возбудятся нейроны b_1, b_2 и b_3 (мы можем ими воспользоваться в США, России и Канаде), и т. д.

От бинарных связей перейдем к биполярным (это сделано исключительно для простоты, чтобы не нужно было вводить ненулевой порог срабатывания нейронов):

$$x_1 = (1, 1, -1, -1) \rightarrow y_1 = (1, 1, 1, -1);$$
$$x_2 = (1, -1, 1, -1) \rightarrow y_2 = (-1, 1, -1, 1);$$
$$x_3 = (1, -1, -1, 1) \rightarrow y_3 = (-1, 1, -1, -1).$$

Составим матрицу весов:

$$W = \sum_{l=1}^{3} x_l^T y_l = \begin{pmatrix} -1 & 3 & -1 & -1 \\ 3 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \\ -1 & -1 & -1 & -1 \end{pmatrix}.$$

Режим распознавания

Оценим эффективность запоминания обучающих связей. Убедимся, что матрица W хранит связи (x_1, y_1), (x_2, y_2) и (x_3, y_3). Подадим на вход x_1, тогда $x_1 W = (2, 2, 2, -2)$ — это означает, что в слое B возбудятся первые три нейрона (порог срабатывания принят равным нулю). Тогда в бинарной форме $y = (1, 1, 1, 0)$, что является требуемой ассоциацией. Это означает, что подача на вход x_1 приводит к y_1, то есть ЭВМ действительно «запомнила» связь (x_1, y_1). Аналогично проверяется запоминание остальных связей.

Сеть является двунаправленной: $y_1 W^T = (1, 5, -3, -3) \rightarrow (1, 1, 0, 0) \rightarrow x_1$ и т. д.

Определим энергию связей в памяти:

$E(x_1, y_1) = -x_1 W y_1^T = -6$, аналогично $E(x_2, y_2) = -4$ и $E(x_3, y_3) = -2$. Следует ожидать, что при ошибке в исходной информации связь (x_1, y_1) будет притягивать к себе больше образов, так как это точка устойчивого равновесия с минимальным энергетическим уровнем. Действительно, подадим на вход образ $x' = (1, 1, 0, 1)$ — искаженный на один бит x_1 и x_3, тогда $x'W = (1, 1, 1, -3) \rightarrow (1, 1, 1, 0) \rightarrow y_1$. Аналогично, если взять $x'' = (1, 0, 1, 1)$ — вектор, расположенный «между» x_2 и x_3, то получим $(-3, 1, -3, 1) \rightarrow (0, 1, 0, 1) \rightarrow y_2$ — связь (x_2, y_2) притягивает к себе, так как ее энергия меньше энергии (x_3, y_3).

Работа с неопределенными данными. Рассмотрим случай, когда тип валюты неопределен: $x' = (1, 0, 0, 0)$, тогда $x'W = (-1, 3, -1, -1) \rightarrow (0, 1, 0, 0) \rightarrow y_3$. Это означает, что она может быть использована только в той стране, где в ходу любая валюта.

Если валюта может быть любой, например, доллары и марки, то она может использоваться везде:

$$x' = (1, 1, 1, 0) \rightarrow x'M = (1, 1, 1, 1) \rightarrow y'.$$

Проведенное исследование показывает, что построенная нейросеть способна запомнить необходимую информацию на этапе обучения, а в рабочем режиме по-

зволяет решать задачи распознавания, то есть реализует функции ассоциативной памяти. Вся полученная при обучении информация сосредоточена в матрице W. За счет параллельной структуры сеть решает задачу «мгновенно» — за одно действие — умножение входного вектора на матрицу памяти. Так как информация как бы интегрирована в матрицу W, сеть способна достаточно эффективно решать задачу и при частичных искажениях в исходных данных.

13.6.4. Самоорганизующиеся сети Кохонена

Характерной особенностью мозга является то, что его структура, по-видимому, отражает организацию внешних раздражителей, которые в него поступают. Например, известно соответствие относительного положения рецепторов на поверхности кожи и нейронов в мозге, которые воспринимают и обрабатывают сигналы от этих рецепторов, а участки кожи, которые плотно «населены» рецепторами (лицо, руки), ассоциированы с пропорционально большим числом нейронов. Это соответствие образует то, что называется соматотропической (somatotropic) картой, которая отражает поверхность кожи в часть мозга — соматосенсорную кору, которая и воспринимает ощущение касания.

Системы, построенные на базе, например, многослойных сетей с обратным распределением ошибок имеют очевидный недостаток, заключающийся в том, что мы (как учитель) должны заранее заготовить входы и запастись соответствующими правильными ответами для обучения сети. Принципы функционирования природной соматотропической карты легли в основу создания самоорганизующихся сетей (карт, решеток) Кохонена, для которых не требуется предварительное обучение на примерах. Сеть Кохонена воспринимает только вход и способна вырабатывать свое собственное восприятие внешних стимулов.

Самоорганизующиеся сети Кохонена — это карты или многомерные решетки, с каждым узлом которой ассоциирован входной весовой вектор, то есть набор из k входных весов нейрона трактуется как вектор в k-мерном пространстве. (На рис. 13.5 представлена сеть при $k = 2$.) Входной весовой вектор имеет ту же размерность, что и вход в сеть. Обучение происходит в результате конкуренции, возникающей между узлами сети за право отклика на полученный входной сигнал. Элемент сети, который выигрывает в этой конкуренции (победитель), и его ближайшее окружение (свита) модифицируют веса своих входных связей.

Перед обучением каждая компонента входного весового вектора инициализируется случайным образом. Обычно каждый вектор нормализуется в вектор с единичной длиной в пространстве весов. Это делается делением соответствующего веса на корень из суммы квадратов компонент этого весового вектора. Входные вектора нормализуются аналогично.

Обучение сети состоит из следующих этапов:

1. Вектор $\mathbf{x} = (x_1, x_2, ..., x_k)$ подается на вход сети.
2. Определяется расстояние d_{ij} (в k-мерном пространстве) между $\mathbf{x}$ и весовыми векторами w_{ij} каждого нейрона, например: $d_{ij} = \sqrt{\left[\sum_k (x_k - w_{ij,k})^2\right]}$.

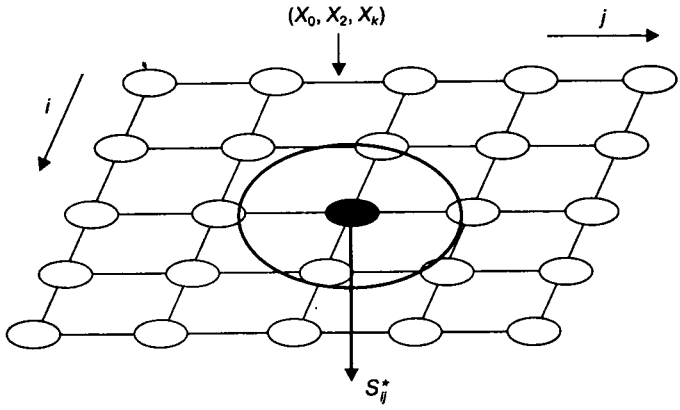


Рис. 13.5. Самоорганизующаяся карта Кохонена

3. Нейрон, который имеет весовой вектор, самый близкий к x, объявляется «победителем». Этот весовой вектор w_{ij} становится основным в группе входных весовых векторов, которые лежат в пределах расстояния D от w_{ij}. Таким образом определяется «свита» победителя.
4. Группа входных весовых векторов модифицируется (поощряется) в соответствии со следующим выражением $w_{ij}(t+1) = w_{ij}(t) + \eta(x - w_{ij}(t))$ для всех весовых векторов в пределах расстояния D от w_{ij}.
5. Шаги 1 — 4 повторяются для каждого входного вектора.

В процессе обучения значения D и η постепенно уменьшаются: η: 1→0, D в начале обучения может равняться максимальному расстоянию между весовыми векторами, а к концу обучения доходить до величины, при которой будет обучаться только один нейрон.

Из формулы адаптации входного весового вектора следует, что он (для победителя и его «свиты») сдвигается по направлению к входному вектору. Таким образом, по мере поступления новых входных векторов весовые векторы сети разделяются на группы, формирующиеся в виде облаков (сгустков, кластеров) вокруг входных векторов. По мере обучения плотность весовых векторов будет выше в тех позициях пространства, где входные векторы появляются чаще, и наоборот. В результате сеть Кохонена адаптирует себя так, что плотность весовых векторов будет соответствовать плотности входных векторов. Так, если, например, на вход сети подать поток равномерно распределенных случайных величин, то сетевые веса будут самоприводиться в порядок в регулярную решетку. На рис. 13.6 приведен пример самообучения двухмерной сети Кохонена.

ПРИМЕЧАНИЕ

В общем случае $h = h(r, t)$, где r — расстояние между нейронами, t — параметр, характеризующий время обучения. Эта функция традиционно имеет вид «мексиканской шляпы» (зависимость силы связи от расстояния до победителя), которая по мере увеличения параметра t делается более узкой.

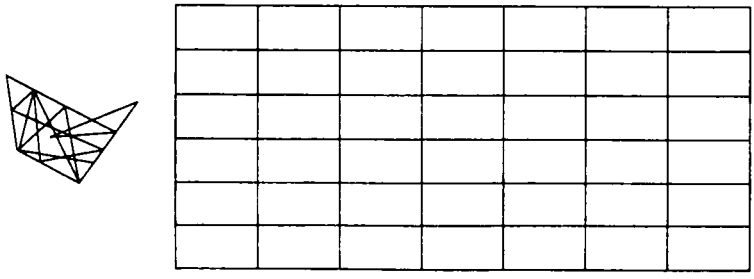


Рис. 13.6. Самоорганизация сети Кохонена

Обучающий алгоритм настраивает входные весовые векторы в окрестности возбужденного нейрона таким образом, чтобы они были более «похожими» на входной вектор. Если все векторы нормализованы в векторы с единичной длиной, то они могут рассматриваться как точки на поверхности единичной гиперсферы. В процессе обучения группа соседних весовых точек перемещается ближе к точке входного вектора.

Очевидно, так как $S = \sum_k x_k w_k$, то победитель может быть найден так: $(i^*, j^*) = \arg\max(S_{ij})$ — наиболее активный нейрон.

Самоорганизующиеся сети, как уже отмечалось, не требуют предварительного обучения на примерах (учитель не нужен) и используются в задачах распознавания — классификации образов, представленных векторными величинами, в которых каждая компонента вектора соответствует элементу образа. После обучения подача входного вектора из данного класса будет приводить к возбуждению нейронов; тогда нейрон с максимальным возбуждением и будет представлять классификацию. Очевидно, в общем случае можно формировать выход, зависящий как от активности победителя, так и от его свиты. Так как обучение проводится без указания целевого вектора, то априори невозможно определить, какой нейрон будет соответствовать данному классу входных векторов. Однако после обучения такие соответствия легко идентифицируются и могут быть использованы, например, в задачах управления.

13.7. Области применения нейроинформатики

Основное место на рынке услуг, оказываемых реальному потребителю нейроинформатикой, сейчас занимают финансовые приложения. Объясняется это тем, что нейросети эффективно справляются с задачами классификации, моделирования и экстраполирования (прогнозирования, предсказания), что особенно важно при решении финансово-экономических проблем.

Нейронные сети служат основой для создания программных пакетов (имитаторов), плат-акселераторов для персональных ЭВМ, нейроБИС, а также специализированных нейрокомпьютеров. Для отработки методологии решения задач в

нейросетевой постановке на первых этапах часто оказывается достаточным использование соответствующего программного пакета.

На мировом рынке представлено более сотни нейросетевых пакетов, преимущественно американских. Объем рынка нейронных сетей превышает 1 млрд долларов в год. Более того, практически каждый разработчик традиционных аналитических пакетов сегодня стремится включить нейронные сети в новые версии своих программ. В США нейронные сети применяются в аналитических комплексах каждого крупного банка.

Остановимся только на краткой характеристике пакета The AI Trilogy и определим место этого пакета на финансовом рынке. Выбор не случаен — ключевые компоненты пакета (лучшего пакета ФБР) были написаны российскими специалистами.

Пакет The AI Trilogy (**«Трилогия искусственного интеллекта»**) американской фирмы Ward Systems Group — это набор из трех программ, каждая из которых может использоваться как самостоятельно, так и в комбинации с остальными: программа NeuroShell II — это набор из 16 типов нейронных сетей, NeuroWindows — нейросетевая библиотека с исходными текстами, GeneHunter — генетическая программа оптимизации. В совокупности они образуют весьма мощный **конструктор**, позволяющий строить аналитические комплексы любой сложности.

Позиции «Трилогии» на американском рынке чрезвычайно сильны. Пакет установлен в 150 крупнейших банках США, многократно побеждал в престижных конкурсах популярных финансовых изданий, помогает управлять капиталами в несколько миллиардов долларов. Отчеты фирмы Du Pont, института стандартов США и ФБР называют входящие в «Трилогию» пакеты лучшими для решения различных задач.

Приход пакета The AI Trilogy на рынок России и создание его русской версии знаменуют собой наступление нового этапа становления отечественного рынка аналитических систем. Сегодня аналитический рынок России созрел для освоения нового класса пакетов — функционально полных сложных систем, позволяющих решать комплексные задачи управления финансовыми ресурсами.

Что может дать пакет финансисту? Приведем только два примера. Во время подготовки презентации русской версии «Трилогии» были взяты реальные данные о портфеле небольшой финансовой компании из Калифорнии, играющей на рынке так называемых индексных опционов. Проведение средствами «Трилогии» классической последовательности «Анализ, прогноз, оптимизация» позволило в первый день «сделать» 25 тыс. долларов при величине портфеля в 2 млн. долларов, а во второй — добавить еще 40 тыс.!

Пример комплексной системы, построенной на основе «Трилогии». Аналитический комплекс финансовой компании LBS (США), имеющей в управлении более 1 млрд долларов клиентских денег, управляется всего одной кнопкой — «Старт». Далее система сама обновляет базу данных по котировкам акций 3000 компаний на Нью-Йоркской бирже, определяет наиболее прогнозируемые акции, параллельно запускает несколько видов прогнозов, выбирает самые перспективные с точки зрения краткосрочной игры компании, оптимизирует портфель и выдает

рекомендации трейдерам. Аналитикам остается только соотнести рекомендации системы с собственными представлениями, инсайдерской информацией и фундаментальными факторами.

Еще раз перечислим основные преимущества нейронных сетей

- наиболее ценное свойство нейронных сетей — способность обучаться на множестве примеров в тех случаях, когда неизвестны закономерности развития ситуации и какие бы то ни было зависимости между входными и входными данными. В таких случаях (а к ним можно отнести до 80% задач финансового анализа) пасуют как традиционные математические методы, так и экспертные системы;
- нейронные сети способны успешно решать задачи, опираясь на неполную, искаженную, зашумленную и внутренне противоречивую информацию;
- для использования методов корреляционного, регрессионного и кластерного анализов вам понадобился бы профессионал-математик. Эксплуатация обученной нейронной сети по силам и школьнику;
- нейросетевые пакеты позволяют исключительно легко подключиться к базам данных, электронной почте и т. д. и автоматизировать процесс ввода и первичной обработки данных;
- внутренний параллелизм, присущий нейронным сетям, позволяет практически безгранично наращивать мощность вашей нейросистемы. Вы можете начать с простого и дешевого пакета, потом перейти на профессиональную версию, потом добавить платы-ускорители, затем перейти на специализированный нейрокомпьютер — с гарантией полной преемственности всего ранее созданного программного обеспечения.

В финансовом мире нейронные сети широко применяются для двух основных задач — прогнозирования котировок основных инструментов (курсов валют, ценных бумаг, ГКО и др.) и распознавания определенных ситуаций (например, подозрительных операций с кредитной картой).

В России наиболее известными приложениями нейросетевых информационных технологий можно признать следующие:

- прогнозирование котировок фьючерсов;
- краткосрочная динамика курсов валют;
- прогноз оптовых цен на продукты питания;
- оценка кредитных рисков;
- оценка объектов недвижимости;
- ряд задач медицинской и промышленной диагностики;
- построение высокодоходного футбольного тотализатора;
- прогноз развития чрезвычайных ситуаций;
- авторизация доступа по индивидуальному «почерку» работы за клавиатурой компьютера.

Сфера финансовых приложений нейронных сетей практически безгранична. Любая задача, связанная с манипулированием финансовыми инструментами — будь то валюта или ценные бумаги, — сопряжена с риском и требует тщательного расчета и прогнозирования. Как изменится завтра котировка основных валют? Вернет ли кредит внешне благополучная фирма? Как подобрать прибыльный и вместе с тем надежный «портфель инвестора»? Эти и сотни других вопросов приходится ежедневно решать аналитическим отделам финансовых (да и не только финансовых) компаний, привлекая все виды аналитических инструментов. Поэтому не случайно, что четвертую часть рынка нейросетевых продуктов составляют финансовые приложения.

Ключевые понятия

- искусственный интеллект;
- знания;
- модели представления знаний;
- экспертная система;
- интеллектуальный интерфейс;
- искусственный нейрон;
- нейронная сеть;
- топология нейронных сетей;
- нейрокомпьютер;
- нейроинформатика.

Контрольные вопросы

1. Искусственный интеллект. Дать определение, охарактеризовать направления исследований.
2. Знания, определение понятия, классификация, отличие от данных.
3. Перечислить и дать анализ основных моделей представления знаний.
4. Привести обобщенную структуру экспертной системы. Дать характеристику отдельных блоков.
5. Изложить суть бионического подхода к проблеме искусственного интеллекта. Сформулировать принципиальные отличия нейрокомпьютера от «обычной» ЭВМ.
6. Изобразить схему искусственного нейрона. Перечислить его основные блоки и функции. Привести примеры функций активации.
7. Провести классификацию нейронных сетей по типу входной информации и по принципам обучения.
8. Назвать и изобразить основные модели (топологии) нейронных сетей. Указать их области применения.

9. Привести пример (задачу) использования двунаправленной ассоциативной памяти.
10. Назвать основные преимущества использования нейронных сетей в области планирования финансовой деятельности.

Литература

1. Информатика: Учебник для экономических специальностей вузов/ Под. ред. Н. В. Макаровой. М.: Финансы и статистика, 1999.
2. Фридланд А. Я., Ханамирова Л. С. Информатика: Толковый словарь основных терминов: Учеб. пособие. М.: ПРИОР, 1998.
3. Толковый словарь по информатике. 2-е изд., доп. М.: Финансы и статистика, 1995.
4. Толковый словарь по искусственному интеллекту. М.: Радио и связь, 1992.
5. Искусственный интеллект. Справ. В 3-х т./ Под ред. Д. А. Поспелова. Т. 2. Модели и методы. М.: Радио и связь, 1990.
6. Плотников В. М., Суханов В. А. Системы, основанные на знаниях. М.,1995.
7. Уоссерман Ф. Нейрокомпьютерная техника: Теория и практика. М.: Мир, 1992.
8. Нейроинформатика/ А. Н. Горбань, В. А. Дунин-Барковский, А. Н. Кирдин и др. Новосибирск: Наука., 1998.
9. Ежов А. А., Шумский С. А. Нейрокомпьютеринг и его приложения в экономике и бизнесе. М.: Изд-во МИФИ, 1998.
10. Горбань А. Н., Россиев Д. А. Нейронные сети на персональном компьютере. Новосибирск: Наука, 1996.

Заключение

В заключение хочется высказать некоторые мысли о будущих (возможных) путях и тенденциях развития информационных технологий, перспективах информационного общества.

Обыденное сознание, поддерживаемое некоторыми средствами массовой информации, рассуждая на эту тему, склонно к двум крайностям. С одной стороны, это чрезмерно оптимистичные ожидания (порой близкие к эйфории) по поводу бурного прогресса науки, информатизации общества, всеобщей мгновенной доступности любых сведений через телекоммуникационные сети и т. д. А с другой, пессимистические рассуждения о том, что достигшие совершенства информационные системы, оказавшись в руках злонамеренного меньшинства, могут стать мощным инструментом подчинения общества, подавления прав, свобод и индивидуальности отдельной личности.

Вряд ли было бы конструктивным безусловно присоединиться к сторонникам одной из этих точек зрения. С нашей точки зрения, имеет смысл трактовать информатизацию как неизбежный и объективный этап в развитии человечества. Такой, каковыми в свое время были Эпоха Великих географических открытий или «промышленный переворот» начала XIX века. Как и любое событие такого масштаба, информационная революция несет в себе положительные и отрицательные стороны, решая одни проблемы, неизбежно порождает другие. По мнению авторов, принципиальным качеством мировоззрения современного человека должно являться осознание объективной природы тех трудностей, с которыми столкнется наша цивилизация на путях развития современных информационных технологий. Их можно предвидеть, их можно преодолевать, можно тем или иным способом снижать их негативное влияние, но избежать их – невозможно!!!

Мы находимся в самом начале формирования информационного общества, а жизнь уже сильно изменилась по сравнению с тем, что было 10 или 20 лет назад. В экономически развитых странах количество занятых в секторе информации (производство, хранение, передача данных) намного больше занятых непосредственно в промышленности. Россия догоняет их в этом отношении. Производство материальных благ становится более информационно емким. В себестоимости продукции большую часть занимают интеллектуальные и маркетинговые затраты. В результате жизнь стала разнообразнее, потому что рутина уходит из производственного процесса (за конвейером следят АСУ ТП, массивы данных обрабатывает компьютер). Теперь работники, и не только высшего звена, заняты решением сложных творческих задач, не поддающихся формализации: это разработка перспективных планов на основе предвидения, интуиции, аккумуляции опыта, решение стратегических задач.

Нельзя оставить без внимания развитие сферы развлечений, более тесно связанной теперь с информационными технологиями. Понемногу наш быт также меняется под влиянием этих новшеств.

Почему же не все довольны этим?

Прежде всего недовольство высказывают те, кто не сумел вписаться в новые условия организации труда. Переучиваться могут и хотят не все. Кому-то кажется высоким темп современной жизни. Идет объективный процесс изменения социальных слоев общества в сторону роста количества занятых в информационных сферах, что требует новых квалификаций.

Эта категория работающих должна осознавать проблемы, встающие перед ними в результате развития нового типа социума. Им искать пути решения и оценивать, что здесь первостепенно важно. Такие проблемы, как потери информации при передаче и хранении, получение недостоверных данных, расхождения, вызванные нестыковкой терминологии, как раз на начальном этапе создают большие трудности. Хочется верить, что они технически вполне решаемы.

То обстоятельство, что большое количество людей заняты в сфере производства информации, порождает трудность поиска нужных данных. Кроме того, что наблюдается лавинообразный рост документов, содержащих разнообразные сведения, их использование затруднено из-за нечеткой организации хранения, каталогизирования и систематизации. Такое рассеяние информации приводит к тому, что проще и дешевле получить данные самому, чем заниматься поиском. Путь решения проблемы здесь достаточно ясен. Остается надеяться, что в данном случае совершенствование информационных технологий и прогресс информационного общества способны успешно разрешить это противоречие.

Политические факторы могут способствовать тому, что вред от развития информационных технологий будет все более ощутимо влиять на общество, породившее их. Уже сейчас видно, как информационные технологии активно используются в борьбе за влияние на людей, за установление контроля над ними и, в конечном счете, за их подчинение. Банки данных с адресами, телефонами и паспортными данными уже сейчас можно свободно купить, а если на очереди диски с данными по налогам, крупным приобретениям и состоянию здоровья? Однако наблюдается и противоположный процесс сокрытия информации. Это касается секретных ведомственных данных, необоснованно скрываемых. В идеале информационное общество — то, где доступ к любой информации, кроме государственных тайн, свободен и дешев, но не нарушает прав и свобод человека. Развитие же вышеупомянутых тенденций приведет к информационному хаосу.

Другая группа противоречий, вызванных технологическим прогрессом, связана с внутренним конфликтом между целевым назначением компьютеров и нецелевым использованием технологий. Мы имеем в виду те последствия развития информационного общества, которые еще только угадываются на современном этапе. Пока трудно предвидеть, насколько опасны те или иные тенденции, что и объясняет порой излишне эмоциональные оценки и предостережения, касающиеся новых технологий и возможностей. Противоречивость прогресса проявляется в том, что на каждое полезное нововведение или изобретение изворотливый ум от-

вечает отрицательным применением этого полезного изобретения. Это взламывание банковских кодов, рассылка вирусов по электронной почте, обман людей с помощью компьютерного MLM-маркетинга, «интернетный аутизм», злоупотребление виртуальными играми... Вряд ли можно предсказать сейчас, чего ожидать в будущем в ответ на современное развитие технологии. Но стоит иметь в виду, что нарушения технологических процессов, сбои или злоупотребления в эпоху информационной революции могут иметь фатальные последствия в силу того, что могут распространятся очень быстро по сетям, не имеющим какой-либо подходящей защиты, или «только» причинят большой ущерб, выведя из строя большие системы.

Человек всегда стремился к независимости от факторов природы и влияния общества себе подобных. Социальное и индивидуальное породили много проблем невозможностью их гармоничного уравновешивания. Теперь, получив в распоряжение *персональный* компьютер, человек, кажется, добился своего. Но если рассмотреть это явление в исторической перспективе, то можно заметить, что зависимость от общества только возрастает. Подобную обратную связь мы уже ощутили, столкнувшись с проблемами экологии. Теперь прогресс информационного общества усиливает зависимость отдельного индивида во всех сферах деятельности. И люди сами идут в эту кабалу.

На современном этапе развития информационных технологий специалисты предпочитают или беспристрастно констатировать наличие технических и психологических проблем, или с оптимизмом смотреть в будущее. Речь о результатах решения глобальных проблем не ведется. Думается, осознание этих трудностей приходит постепенно — прогресс, как мы сказали ранее, порождает проблемы. Они затрагивают уже не только сферу компьютерных технологий и информации, но, проникая в общественную жизнь, политику и быт. Информационное общество должно развиваться, имея в виду постоянно и объективно возникающие новые противоречия, которые могут быть разрешены человеческим интеллектом.

Алфавитный указатель

A

ActiveX, 116, 132, 183
Adobe, 151
Apple, 173

B

Bell & Howell Information and Learning, 507
Bell Telephone Laboratories, 143
BIOS, 65, 100
Bridge, 397

C

CD-R, 70
CD-ROM, 70
CD-RW, 70
CGI (Common Gateway Interface), 451, 453
Clipboard, 112
CONFIG.SYS, 107
Corel, 172, 314

D

DDE, 112
DNS, 439
DNS-сервер, 445
DR DOS, 88
DVD, 70
DVD-RAM, 70

E

Ethernet, 411, 442

F

Fast Ethernet, 413
FAT, 91
FAT16, 95
FAT32, 95
FDDI, 442
Floppy, 69

G

Gateway, 397
Gopher, 456
Groupware, 350

H

HTML, 157, 317, 331, 338, 451, 453, 457
HTTP, 451, 453, 456, 458

I

IBM, 90
Internet
  Society, 441
  технология, 402
IP-телефония, 463
ISAM
  драйверы, 300

J

Java, 116, 132, 451
Jet, 302

K

Kerberos, 116

L

Lexis-Nexis, 504
Linux, 315
Lotus Domino, 351
Lotus Notes, 350

M

Microsoft, 88, 90, 173, 312
MIME, 447, 453
MS Access, 271
  версии, 271
  объекты, 272
MS Clip Gallery, 312
MS DOS, 88, 173
MS Equation, 196
MS Excel, 210, 318, 342
  абсолютная ссылка, 220
  ввод данных
    автозаполнение, 217
    прогрессии, 217

MS Excel *(продолжение)*
диаграммы, 221
внедренная, 224
мастер, 222
редактирование, 224
лист, 211
структура, 211
мастер функций, 219
общий интерфейс, 212
относительная ссылка, 221
панели инструментов, 213
перемещение, 214
повторный ввод данных, 217
подбор параметра, 247
поиск решения, 250
рабочая книга, 211
рабочее пространство, 211
сводные таблицы, 226
сортировка, 225
список, 224
поле, 225
таблица подстановки, 255
ускоренный ввод данных, 216
фильтр
автофильтр, 225
расширенный, 226
финансовые расчеты, 232
формула, 218
порядок вычисления, 219
редактирование, 220
функции, 219
статистические, 227, 229
ячейка, 211, 215
адресация, 211
имя, 212
содержимое, 214
типы данных, 212, 216
ячейки
ввод данных, 216
изменение данных, 216
формат, 216
MS Exchange, 351
MS Front Page, 175
MS FrontPage, 319
MS Graph, 206, 312
MS Internet Explorer, 317
MS Office, 155, 212, 351
Office 2000, 173
professional, 320
MS Organization Chart, 342
MS Outlook,.318
MS PowerPoint, 312
авторазметка, 323, 340
местозаполнитель, 332
панели инструментов, 323
настройка, 324
MS PowerPoint *(продолжение)*
режимы
заметок, 326
показа слайдов, 326
слайдов, 325
сортировщика слайдов, 326
структуры, 325
форматирование
логическое, 331
физическое, 330
экранный интерфейс
встроенные меню, 323
окна, 322
панели инструментов, 322
рабочая область, 322
строка состояния, 323
строка меню, 322
MS Project, 354
MS Windows, 110, 173, 315, 335
INI-файлы, 113
Windows 2000
особенности, 116
Windows 3.1, 111, 113
Windows 95, 113
Windows 98, 114
Windows NT, 116
корзина, 120
окно, 111
рабочий стол, 118
реестр, 114
MS Word, 173, 312, 326, 339
Add-in-файлы, 177
Word 2000, 155, 174
новые возможности, 174
усовершенствованные возможности, 175
Word 95, 173
Word Art, 341
«горячие» клавиши, 184
закладки, 200
макровирусы
защита, 178
механизм работы, 177
макросы
создание, 176
нововведения, 174
оглавление
сборка, 206
организатор, 195
параметры
другие, 189
страницы, 188
перекрестные ссылки, 200
поля, 198
режимы отображения, 187
Web-документ, 188
обычный (Normal), 187
разметки (для печати), 188
структура (Outline), 188

MS Word *(продолжение)*
списки, 206
стили, 192
форматирование
логическое, 193
физическое, 192
шаблон
Normal.Dot, 186
экранный интерфейс, 179
SDI, 179
заголовок меню, 180
меню, 180, 181
настройка, 181
панели инструментов, 181
рабочая область, 181
строка состояния, 181

N

NetShow, 318
NetWare, 90
Novell, 90
NTFS, 91

O

ODBC, 298
ODBCDirect, 302
OLE, 112, 271
объект, 196
сервер, 342
элементы управления OLE, 287
OS/2, 90, 315

P

PC DOS, 88
Postscript, 160

R

RAID, 71
Repeater, 396

S

Santa Crus Operations, 90
SGML (Standard Generalized Markup Language), 449
Solaris 2.x, 90
SQL, 265
DDL, 265
DML, 265
предложения, 266
язык управления транзакциями, 267
StarOffice, 314
Sun, 90, 315
Switch, 397
Symantec, 109
System R, 265

T

Token Ring, 415

U

UMI (University of Microfilm International), 504
Universal Resource Identification, URI, 452
Universal Resource Locator, URL, 451, 452, 456
UNIX, 91, 142, 173, 315

V

Visual Basic for Applications, 176, 212, 270, 314, 324, 373

W

WordPerfect, 314
WWW (World Wide Web), 451, 453
WYSIWYG, 151, 188

X

X Window, 173

А

Абстрактная сущность, 512
Автоматизированные банковские системы (АБС), 270
Автофигура, 320
Антивирусные программы, 139
AntiViral Toolkit Pro (AVP), 140
Архивация, 132
программы, 132
Архитектура
двойной независимой шины, 58
фон Неймана, 39
Ассемблер, 42
Атрибут, 260
Аутентификация, 472
S/Key, 473

Б

База данных, 260
защита данных, 307
нормальные формы, 264
экономические приложения, 274
этапы разработки, 273
База знаний, 512
Бейсик, 45
Битовая синхронизация, 408
Блокировка, 303
виды, 304

Брандмауэр, 473
Броузер, 456, 458
Булева алгебра, 35
Буфер обмена, 112
Бэббидж, Чарльз, 33

В

Вакуумная лампа, 38
Ввод текста
  голосовой, 154
  рукописный, 154
  с клавиатуры, 150
Видеоадаптер, 81
Винчестер, 68
Вирусы, 136
  загрузочные, 137
  классификация, 137
  макровирусы, 138
  полиморфные, 138
  резидентные, 138
  сетевые, 138
  стелс-вирусы, 138
  файловые, 137
  черви (wormes), 138
Внешние данные
  импорт, 299
  связь, 300

Д

Данные, 15
Дейтаграмма, 443
Делопроизводство
  автоматизация, 350
Джойстик, 72
Диски
  дефрагментация, 128
  проверка, 129
  форматирование, 126
Домен, 264, 444
Драйверы, 101, 107

З

Запись, 261, 264
Запросы
  SQL-режим, 284
  групповые операции, 283
  конструктор, 281
Защита данных
  на уровне пользователей, 308
  установка пароля, 308
Звуковая карта, 85
Знания, 15, 512
  декларативные, 513
  процедурные, 512

И

Инструкции KNI (Katmai New Instruction), 60
Интегрированная библиотечно-информационная система «Ирбис», 493
Интеллектуальные системы, 509
Интенсионал, 511
Интернет, 116, 313, 327, 331, 337, 436, 449, 456
  поиск, 459
    индекс, 460
    индексатор, 460
    классификаторы, 459
    поисковые машины, 460
Интерфейс, 403
  FireWare, 84
  накопителей, 66
Информатика, 16
Информационная система, 260
Инфомационные
  ресурсы, 498
  технологии, 16
Информация, 14
  внешняя, 20
  входная, 20
  выходные данные, 22
  директивная, 20
  защита, 470
  измерение, 15
  нормативно-справочная, 21
  осведомляющая, 20
  первичная, 20
  плановая, 22
  показатели, 22
  структура, 15
  угрозы безопасности, 469
  учетная, 21
  формы представления, 15
  экономическая, 19
ИПС
  в глобальных сетях, 482
  запросы пользователя, 482, 483
  избыточное индексирование, 480
  индекс базы данных, 483
  индексирование, 478
  интерфейс
    рубрикационного типа, 481
    структурно-логические системы, 482
  информационно-поисковые системы, 477
  поисковая машина, 483
  поисковый образ документа, 478
  робот индексирования, 483
  словарь, 478
  термины индексирования, 478
  указатель (индекс), 477
  язык, 478

Искусственный интеллект, 509
направления исследований, 510
бионическое, 510, 524
программно-прагматическое, 510

К

Кадр, 405
Каталог, 92
Кибернетика, 17
Клавиатура, 71
Класс, 260
Кластер, 95
Клиент-сервер, 306
AS-модель, 307
DBS-модель, 306
RDA-модель, 306
Ключ, 261
вторичный, 261
первичный, 261
Кобол, 43
Коллизия, 413
Командные файлы, 105
автонастройки, 106
Команды ОС, 102
классфикация, 103
формат, 102
Коммуникационное оборудование, 394
Коммуникационный узел, 396
Коммутатор, 397, 408, 418
Коммутация
динамическая, 408
каналов, 408
пакетов, 408
постоянная, 408
сообщений, 409
Компьютер, 50
мейнфрейм, 51
микрокомпьютеры, 52
мини-компьютер, 52
персональные компьютеры, 52
рабочая станция, 52
сервер, 52
сетевые компьютеры, 52
суперкомпьютер, 51
Кортеж, 264
Криптография, 470
Кэш-память, 57

Л

Лейбниц, Готфрид Вильгельм, 32
Линия связи
коммутируемая, 408
некоммутируемая, 408

М

Магнитооптические накопители, 69
Макровирусы, 177
Макросы, 294
Autoexec, 295
вызов, 295
группа макросов, 294
макрокоманды, 294
Маркер, 415
Маршрутизатор, 117, 394, 396, 397, 406, 419, 438, 469, 473
маршрут, 420
таблица маршрутизации, 420
Математический сопроцессор, 57
Материнская плата, 53
Метод доступа
CSMA/CD, 413
Demand Priority, 414
маркерный, 415
Микрокомпьютер, 46
Микроплатежи, 469
Микропроцессор, 45, 51
разрядность, 56
тактовая частота, 56
Микросхема, интегральная, 43
Мини-компьютер, 44
Модель OSI, 403, 442
уровни
канальный, 405
представления, 407
прикладной, 407
сеансовый, 407
сетевой, 406
транспортный, 407
физический, 405
Модель данных, 261
иерархическая, 261
реляционная, 263
сетевая, 262
Модель доменов Windows NT, 426
доверительные отношения, 426
Модель рабочих групп Windows NT, 426
Модем, 84
Модули, 296
класса, 297
стандартные, 297
Монитор
CRT, 73
LCD, 76
Мост, 394, 396, 397, 406, 418
адресная таблица, 418
Мультипроцессирование, 90
Мышь, 71
Мэйнфрейм, 44

Н

Нейман, Джон фон, 39
Нейроинформатика, 510, 525
  применение, 539
Нейрокомпьютер, 525
Нейрон, 526
  элементы, 526
Нейросеть, 525
  обучение
    дельта-правило, 531
  персептрон, 530
  сети Кохонена, 537
  сеть Хопфилда, 532
  типы, 526
  типы классификации, 528
  топология, 528

О

Объект, 260
ООП, 296
  инкапсуляция, 296
  класс, 296
  метод (method), 296
  модульность, 297
  наследование, 297
  полиморфизм, 297
  свойство (property), 296
  семейства, 297
  экземпляр класса, 296
Операционная система
  сетевая, 422
    одноранговая, 424
    с выделенным сервером, 424
Операционные системы
  многозадачные, 90
  многопользовательские, 90
  однозадачные, 89
  однопользовательские, 90
  понятие, 87
  сетевые, 394
Отношение, 263
Отчеты, 291
  MS Project, 372
    финансовые, 390
  автоотчет в столбец, 291
  автоотчет ленточный, 291
  вычисление итогов и среднего, 294
  конструктор, 292
  сортировка и группировка, 292, 293

П

Пакет, 406, 408, 419
Память
  динамическая, 63
  статическая, 63
Параллельный интерфейс, 83
Паскаль, Блез, 32, 45
Перевод, автоматизированный, 164
  программы, 165
    Magic Gooddy, 171
    POCKET PROMT, 171
    PROMT, 167
    PROMT Internet, 168
    Systran, 167
Передача данных
  режим
    асинхронный, 408
    полнодуплексный, 414
    полудуплексный, 414
    симплексный, 415
    синхронный, 408
Переменные среды, 106
Перфокарта, 34
Повторитель, 396
Поле, 264
Порт USB, 84
Порты ввода/вывода, 84
Последовательный интерфейс, 83
Предметная область, 260
Представление знаний, 513
  модели, 514
    логические, 514
    продукционные, 515
    сетевые, 518
    фреймовые, 519
Презентация, 312
Принтер, 81
  LED, 82
  лазерный, 82
  струйный, 81
Провайдер, 441
Проект, 351
  базовый план, 381
  затраты
    прямые (постоянные), 390
  критический путь, 377
  метод PERT, 352
  метод критического пути, 352
  оперативно-календарное планирование, 351
  оперативное управление, 381
  операция
    критическая, 377
    поздний финиш, 377
    ранний старт, 377
    свободный резерв времени, 377
  описание
    вершинный граф, 354
    диаграмма Гантта, 354
    стрелочный граф, 353

Проект *(продолжение)*
представление (визуальное)
детальная таблица загрузки ресурсов, 370, 384, 390
диаграмма загрузки ресурсов, 370, 384, 390
диаграммы Гантта, 367, 369
календарный график, 366
комбинированное, 370
сводная таблица загрузки ресурсов, 370, 384
сетевая диаграмма, 369
создание, 370
таблица данных по операциям, 369, 384, 390
пример, 356
промежуточный план, 382
ресурсы, 363
возобновляемые, 364
невозобновляемые, 364
сверхурочная работа, 387
задание, 389
сетевое планирование и управление, 352
структура
параллельная, 376
последовательная, 376
управление
программное обеспечение, 351
финансами, 386
фильтры, 371
Протокол, 404
FTP, 442
HTTP, 442
IPX, 421
IMAP, 442
IP, 421, 439
MIME, 442
NCP, 438
NSP, 421
OSPF, 421
POP, 442
PPP, 442
RIP, 421
SLIP, 442
SMTP, 442
SNMP, 442
SPX, 421
TCP, 421, 438
номер порта, 444
telnet, 442
UDP, 438
дейтаграммный, 404
маршрутизации, 407
разрешения адресов, 407
с установлением соединения, 404
сетевой, 407
Процессоры электронных таблиц, 210
назначение, 210
Публикация документа, 158
печать, 158
электронная, 158
электронные книги, 159

P

Рабочая станция, 395
Распознавание речи, 154
Dragon Dictate, 154
Распознавание текстов
CuneiForm, 153
FineReader, 153
Режимы
монопольного доступа, 303
разделенного доступа, 303
Релевантность, 462
Решатель, 510
Рынок информационных услуг, 499
инфраструктура, 500
компоненты, 500

C

Семантика, 518
Семантическая сеть, 519
Сервер, 395
Novell NetWare, 428
Windows 2000, 117
Windows NT, 425
баз данных, 395, 430
доступа, 396
коммуникационный, 396
печати (print-server), 429
прикладных программ, 396
приложений, 351
резервного копирования данных, 396
файловый, 395
факс-сервер, 396
электронной почты, 430
Сетевая карта, 85
Сетевые приложения, 394
Сети, 394
EUNet, 441
RUNet, 441
X.25, 442
адресация
InterNIC, 442
IP-адрес, 439, 442
глобальные, 401, 436
защита, 473
Интернет, 436
кампусов, 402
корпоративные, 402
локальные, 400

Сети *(продолжение)*
отделов, 402
рабочих групп, 402
сегмент, 396
составная, 419
топология, 406
характеристики
безопасность, 398
время реакции, 397
интегрируемость, 399
масштабируемость, 398
надежность, 398
прозрачность, 399
производительность, 397
пропускная способность, 397
расширяемость, 398
управляемость, 398
Системная
память, 63
шина, 54
Сканер, 72
Слайд, 312
Смарт-карты, 469
Современные информационные технологии, 509
Сокет, 444
Сообщение, 408
Сортировка, 280
СПС
Гарант, 490
поиск, 490
гипертекстовые ссылки, 487
Консультант, 488
папки документов, 490
параметры, 486
поиск
по реквизитам документов, 487
по тексту документов, 487
примеры, 485
справочно-правовые системы, 484
технологии передачи информации, 488
шум, 487
юридическая обработка, 486
Стандарт IEEE 802.3, 411
Стек
коммуникационных протоколов, 404
протоколов IPX/SPX, 421
протоколов TCP/IP, 420
Стример, 67
СУБД, 261
Informix, 268
Ingres, 268
Oracle, 268
Sybase, 268
персональные, 269
профессиональные, 267
Схема данных
создание, 279
тип связей, 280

Т

Таблицы
конструктор, 275
режим таблицы, 278
Тактовый генератор, 66
Тачпад, 72
Текстовые процессоры, 172
MSWord, 173
StarWriter, 172
Word Perfect, 172
Текстовые редакторы, 172
Технология
SIMD, 58
SSE, 60
обработки множественных данных одной инструкцией, 58
Транзакция, 267
Транзисторы, 41
Транспортировка документов
PDF, 160
PostScript, 160
Трекбол, 72
Тьюринг, Алан, 36
Тьюринга, машина, 36

У

Узлы сети, 406
Умножение тактовой частоты, 57
Управление, 18
Уровень LLC, 411
Уровень MAC, 411
Уровень межсетевых взаимодействий стека TCP/IP, 421
Уровень сетевых интерфейсов стека TCP/IP, 421
Утилиты, 103

Ф

Файл, 91
атрибуты, 92
имя, 91, 92
полное имя, 92
путь, 92
расширение, 92
типы, 104
Файл-сервер
технология, 305
Файловое кэширование диска, 94
Файловые системы, 90
FAT, 91, 95
NTFS, 91

Файловые системы *(продолжение)*
блок, 93
каталоги, 92
кластер, 95
составные части, 91
функции, 91
Факс-сервер, 396
Ферритовые сердечники, 41
Фильтр, 281
Форма
автоформа, 286
конструктор, 286
подчиненная форма, 290
простая форма, 286
разделы, 287
сложная, составная форма, 289
Форматирование, 126
Форматы текстовых документов, 155
ASCII, 155
DOC, 157
HTML, 157
PDF, 160
RTF, 157
Фортран, 43
Фреймы, 519
прототипы (протофреймы), 520
слоты, 520
экземпляры, 520

Х

Холлерит, Герман, 34
Хранение документов
гипертекстовая модель, 476
иерархическая модель, 476
Хранимые процедуры, 306

Ш

Шифрование
алгоритм, 470
ключ, 470
Шифрование *(продолжение)*
с открытым ключом, 471
симметричное, 471
Шлюз, 396, 397, 454
Шрифты, 150
векторные, 150
контурные, 150
PostScript Type 1, 151
TrueType, 151
растровые, 150

Э

Эвристика, 511
Экономическая информация, 19
характеристики, 23
Экспертные системы, 511
Экстенсионал, 511
Электронная
коммерция, 464
подпись, 472
почта, 445
адреса, 445
Электронные
деньги, 469
платежные средства, 468
ЭНИАК, 38

Я

Языки программирования
C++, 523
Lisp, 524
Pascal, 523
ассемблер, 42
Бейсик, 45
высокого уровня, 43
Кобол, 43
машинный язык, 41
Паскаль, 45
Пролог (Prolog), 511, 515, 524
Фортран, 43

КНИГА-ПОЧТОЙ



ЗАКАЗАТЬ КНИГИ ИЗДАТЕЛЬСКОГО ДОМА «ПИТЕР» МОЖНО ЛЮБЫМ УДОБНЫМ ДЛЯ ВАС СПОСОБОМ:

- по телефону: **(812) 294-01-04;**
- по электронному адресу: **postbook@piter.com;**
- на нашем сервере: **www.piter.com;**
- по почте: **197198, Санкт-Петербург, а/я 619 ЗАО «Питер Пост».**

ВЫ МОЖЕТЕ ВЫБРАТЬ ОДИН ИЗ ДВУХ СПОСОБОВ ДОСТАВКИ И ОПЛАТЫ ИЗДАНИЙ:

- Наложенным платежом с оплатой заказа при получении посылки на ближайшем почтовом отделении. Цены на издания приведены ориентировочно и включают в себя стоимость пересылки по почте (**но без учета авиатарифа**). Книги будут высланы нашей службой «Книга – почтой» в течение двух недель после получения заказа или выхода книги из печати.
- Оплата наличными при курьерской доставке (**только для Санкт-Петербурга**). Курьер бесплатно доставит заказ по указанному адресу в удобное для вас время в течение трех дней. Такой заказ лучше оформлять по телефону.

ПРИ ОФОРМЛЕНИИ ЗАКАЗА УКАЖИТЕ:

- фамилию, имя, отчество, телефон, факс, e-mail;
- почтовый индекс, регион, район, населенный пункт, улицу, дом, корпус, квартиру;
- название книги, автора, код, количество заказываемых экземпляров.

Вы можете заказать бесплатный альманах профессиональной литературы Издательского дома «Питер».

ИЗДАТЕЛЬСКИЙ ДОМ ПИТЕР®
WWW.PITER.COM

КНИГА-ПОЧТОЙ

БЕСТСЕЛЛЕР



512 с., 13×20, обл.
Код 1395
Цена наложенным платежом 68 р.

СЕРИЯ «СПРАВОЧНИК»

Б. Карпов

MICROSOFT EXCEL 2000: СПРАВОЧНИК, 2-Е ИЗД.

Во втором издании в связи с выходом официальной русской версии программы исправлены и дополнены некоторые разделы. Отличительная особенность данного издания – наличие полного справочника по встроенным функциям Excel. Им можно воспользоваться при самостоятельном составлении программ для анализа и обработки данных. Удобная структура справочника и объемный алфавитный указатель позволяют за считанные минуты отыскивать нужную информацию. Последовательности шагов, необходимые для выполнения задуманных операций, описаны простым языком, понятным даже начинающему. Все это позволяет рекомендовать книгу всем пользователям, которые используют в повседневной работе Excel – от начинающих и до специалистов, желающих узнать о новинках версии 2000.

БЕСТСЕЛЛЕР



448 с., 13×20, обл.
Код 3007
Цена наложенным платежом 68 р.

СЕРИЯ «СПРАВОЧНИК»

Б. Карпов

MICROSOFT OFFICE 2000: СПРАВОЧНИК

В справочнике в простой и доступной форме излагаются основные принципы создания различных документов и способы решения связанных с этим задач, а также различные новшества, которые выгодно отличают Office 2000 от предыдущих версий этого пакета.
Книга написана для широкой пользовательской аудитории: как для достаточно опытных пользователей, которым нужен удобный справочник по решению конкретных задач, так и для начинающих пользователей, решивших освоить новый продукт корпорации Microsoft.

НОВИНКА



384 с., 13×20, обл.
Код 4009
Цена наложенным платежом 68 р.

СЕРИЯ «СПРАВОЧНИК»

Б. Карпов

MICROSOFT WORD 2000: СПРАВОЧНИК, 2-Е ИЗД.

Второе издание, исправленное и дополненное, является наиболее полным справочным руководством по русской и английской версии Word 2000. Справочник содержит информацию обо всех главных функциональных элементах Word, новшествах, в том числе связанных с работой в сети Интернет, и краткий словарь терминов. В справочнике имеется множество разделов, посвященных нетрадиционным методам решения задач, которые позволяют добиться необходимого результата с минимальными затратами времени. Простые и ясные пошаговые процедуры, а также обширный алфавитный указатель позволяют быстро решить любую поставленную задачу. Книга предназначена для широкой пользовательской аудитории, поскольку содержит в себе не только способы решения конкретных задач, но и ответы на общие вопросы, которые часто задают начинающие пользователи.

ПИТЕР®

КНИГА-ПОЧТОЙ

БЕСТСЕЛЛЕР



1056 с., 17×24, обл.
Код 2757
Цена наложенным платежом 192 р.

СЕРИЯ «ЭФФЕКТИВНАЯ РАБОТА»

М. Додж, К. Стинсон

ЭФФЕКТИВНАЯ РАБОТА С MICROSOFT EXCEL 2000

Книга, издаваемая по лицензии Microsoft Press, представляет новую версию программы обработки электронных таблиц. Авторы – признанные специалисты по Microsoft Excel, написавшие уже не одну книгу по этой теме. Не имеющая аналогов по глубине и широте охвата материала, книга поможет читателю не только разобраться в предоставляемых этой программой возможностях, но и в совершенстве овладеть различными функциями и командами, научиться создавать Web-страницы и работать в Интернете, используя Excel 2000. Более того, книга может использоваться в качестве руководства, шаг за шагом вводящего вас в мир Excel. Издание содержит сотни конкретных примеров и иллюстраций, а также детальную справочную информацию по Microsoft Excel 2000.

БЕСТСЕЛЛЕР



1040 с., 17×24, обл.
Код 2792
Цена наложенным платежом 315 р.

СЕРИЯ «ЭФФЕКТИВНАЯ РАБОТА»

Дж. Вейскас

ЭФФЕКТИВНАЯ РАБОТА С MICROSOFT ACCESS 2000 (+ CD)

Книга посвящена новой версии популярной системы управления базами данных.

Вашему вниманию предлагаются подробные теоретические сведения, подкрепленные многочисленными практическими примерами, которые прекрасно иллюстрируют процессы планирования, создания, заполнения и работы с базой данных. Вы научитесь самостоятельно проектировать и связывать в единое приложение таблицы, запросы, фильтры, формы и отчеты, автоматизировать вашу систему с помощью макросов, создавать пользовательские меню и нестандартные панели инструментов, использовать, если это действительно необходимо, всю мощь языков SQL и Access Basic. Вы узнаете о новых возможностях Access 2000, сможете воспользоваться советами профессионала и найти ответы на большинство возникающих у вас вопросов.

БЕСТСЕЛЛЕР



944 с., 17×24, обл.
Код 2759
Цена наложенным платежом 196 р.

СЕРИЯ «ЭФФЕКТИВНАЯ РАБОТА»

Ч. Рабин

ЭФФЕКТИВНАЯ РАБОТА С MICROSOFT WORD 2000

Неважно, используете ли вы Word с начала 80-х годов или являетесь только начинающим пользователем; пишете ли вы записку, создаете Web-узел или готовите рабочий отчет, объем которого сопоставим с объемом солидного издания – эта книга поможет вам научиться быстро и просто создавать документы, которые имеют привлекательный вид и которыми удобно пользоваться. В ней вы найдете абсолютно все, что необходимо для продуктивного использования Word на работе и дома. Уже первые несколько глав помогут найти ответы на большинство ваших вопросов, а в последующих разделах автор раскроет такие возможности Word, о которых не подозревают даже опытные пользователи.

ПИТЕР®

ИЗДАТЕЛЬСКИЙ ДОМ ПИТЕР®
WWW.PITER.COM

ВНИМАНИЕ СПЕЦИАЛИСТАМ КНИЖНОГО БИЗНЕСА!

УВАЖАЕМЫЕ ГОСПОДА! ИЗДАТЕЛЬСКИЙ ДОМ «ПИТЕР» ПРИГЛАШАЕТ ВАС К ВЗАИМОВЫГОДНОМУ СОТРУДНИЧЕСТВУ. ЗА ДОПОЛНИТЕЛЬНОЙ ИНФОРМАЦИЕЙ ОБРАЩАЙТЕСЬ В НАШИ ПРЕДСТАВИТЕЛЬСТВА, КОТОРЫЕ НАХОДЯТСЯ ПО СЛЕДУЮЩИМ АДРЕСАМ:

Россия, г. Москва
Представительство издательства «Питер», м. «Калужская», ул. Бутлерова, д. 17-Б, оф. 207 и 240, тел./факс (095) 234-38-15,333-15-73.
E-mail: sales@piter.msk.ru

Россия, г. Санкт-Петербург
Представительство издательства «Питер», м. «Электросила», ул. Благодатная, д. 67, тел. (812) 327-93-37, 294-54-65.
E-mail: sales@piter.com

Украина, г. Харьков
Представительство издательства «Питер», тел. (0572) 23-75-63, факс (0572) 14-96-09. Почтовый адрес: 310093, г. Харьков, а/я 9130.
E-mail: piter@tender.kharkov.ua

Украина, г. Киев
Филиал Харьковского представительства издательства «Питер», тел./факс (044) 490-35-68, тел. 490-35-69. Почтовый адрес: 04073, г. Киев, пр. Красных Казаков, д. 6, корп.1.
E-mail: office@piter-press.kiev.ua

Беларусь, г. Минск
Представительство издательства «Питер», тел./факс (0172) 16-00-06. Почтовый адрес: 220012, г. Минск, ул. Кедышко, д.19, а/я 104.
E-mail: piterbel@carry.nsys.by

КАЖДОЕ ИЗ ЭТИХ ПРЕДСТАВИТЕЛЬСТВ РАБОТАЕТ ПО ЕДИНОМУ СТАНДАРТУ ИЗДАТЕЛЬСКОГО ДОМА «ПИТЕР».

КНИГИ ИЗДАТЕЛЬСКОГО ДОМА «ПИТЕР» ВЫ МОЖЕТЕ ПРИОБРЕСТИ ОПТОМ И В РОЗНИЦУ У НАШИХ РЕГИОНАЛЬНЫХ ПАРТНЕРОВ.

Башкортостан
Уфа, «Азия», ул. Зенцова, д. 70 (оптовая продажа), маг. «Оазис», ул. Чернышевского, д. 88, тел./факс (3472) 50-39-00.
E-mail: asiaufa@ufanet.ru

Дальний Восток
Владивосток, «Приморский Торговый Дом Книги», тел./факс (4232) 23-82-12. Почтовый адрес: 690091, ул. Светланская, д. 43.
E-mail: bookbase@mail.primorye.ru

Хабаровск, «Мирс», тел. (4212) 22-74-58, факс 22-73-30. Почтовый адрес: 680000, г. Хабаровск, ул. Ким-Ю-Чена, д. 21.
E-mail: postmaster@bookmirs.khv.ru

Хабаровск, «Книжный мир», тел. (4212) 32-85-51, факс 32-82-50. Почтовый адрес: 680000, ул. Карла Маркса, д.37.

Европейские регионы России
Архангельск, «Дом Книги», тел. (8182) 64-00-56, факс 65-41-34. Почтовый адрес: 163061, пл. Ленина, д. 3.
E-mail: book@atnet.ru

Калининград, «Пресса», тел. (0112) 53-66-97, факс 53-63-87. Почтовый адрес: 236040, ул. Подполковника Иванникова, д. 3-А.
E-mail: book@pressa.gazinter.net

Краснодар, «Когорта», тел./факс (8612) 62-54-97. Почтовый адрес: 350033, ул. Ленина, д. 101.

Северный Кавказ
Ессентуки, «Россы», ул. Октябрьская, 424, тел./факс (86534) 6-93-09. Почтовый адрес: 357600, г. Ессентуки, а/я 80.
E-mail: rossy@kmv.ru

Сибирь
Братск, «Прометей», тел./факс (3953) 43-18-76. Почтовый адрес: 665708, ул.Кирова, д. 35.

Иркутск, «Антей-книга», тел./факс (3952) 34-69-38. Почтовый адрес: 664003, ул. Карла Маркса, д. 20.
E-mail: antey@irk.ru

Иркутск, «Продалить», тел. (3952) 59-13-70, факс 51-23-31. Почтовый адрес: 664031, г. Иркутск, ул. Байкальская, д. 172, а/я 1397.
E-mail: prodalit@irk.ru; http://www.prodalit.irk.ru

Красноярск, «Книжный Мир», тел./факс (3912) 27-39-71. Почтовый адрес: 660049, пр. Мира, д. 86.
E-mail: book-world@public.krasnet.ru

Нижневартовск, «Дом книги», тел. (3466) 23-59-50, факс 23-27-14. Почтовый адрес: 626440, пр. Победы, д.12.

Новосибирск, «Топ-книга», тел. (3832) 36-10-26, факс 36-10-27. Почтовый адрес: 630117, а/я 560.
E-mail: office@top-kniga.ru; http://www.top-kniga.ru

Тюмень, «Друг», ул. Республики, д. 211, тел./факс (3452) 21-02-20.
E-mail: drug@tyumen.ru

Тюмень, «Фолиант», тел. (3452) 27-36-06, факс 27-36-11. Почтовый адрес: 625039, ул. Харьковская, д. 83-А.
E-mail: foliant@tyumen.ru

Татарстан
Казань, «Таис», ул. Гвардейская, д. 9-А, тел. (8432) 76-34-55, факс 38-24-32.
E-mail: tais@bancorp.ru

Урал
Екатеринбург, магазин № 14, ул. Челюскинцев, д. 23, тел. (3432) 53-24-90.
E-mail: gvardia@mail.ur.ru

Екатеринбург, «Валео Плюс», тел. (3432) 42-07-75, 42-56-00.
E-mail: valeo@emts.ru

Экономическая информатика
под ред. П. В. Конюховского и Д. Н. Колесова

Главный редактор *В. Усманов*
Ответственный редактор *Е. Строганова*
Руководитель проекта *И. Корнеев*
Литературный редактор *Н. Дубнова*
Художник *М. Юркова*
Верстка *Р. Гришанов*
Корректор *Н. Рощина*

Лицензия ИД № 01940 от 05.06.00.

Подписано в печать 28.12.00. Формат 70×100$^{1}/_{16}$.
Усл. п. л. 45,15. Доп. тираж 5000 экз. Заказ № 2616.

ЗАО «Питер Бук».
196105, Санкт-Петербург, Благодатная ул., 67.

Налоговая льгота — общероссийский классификатор продукции ОК 005-93,
том 2; 953000 — книги и брошюры.

Отпечатано с диапозитивов в ГПП «Печатный двор»
Министерства РФ по делам печати, телерадиовещания и средств массовых коммуникаций.
197110, Санкт-Петербург, Чкаловский пр., 15.