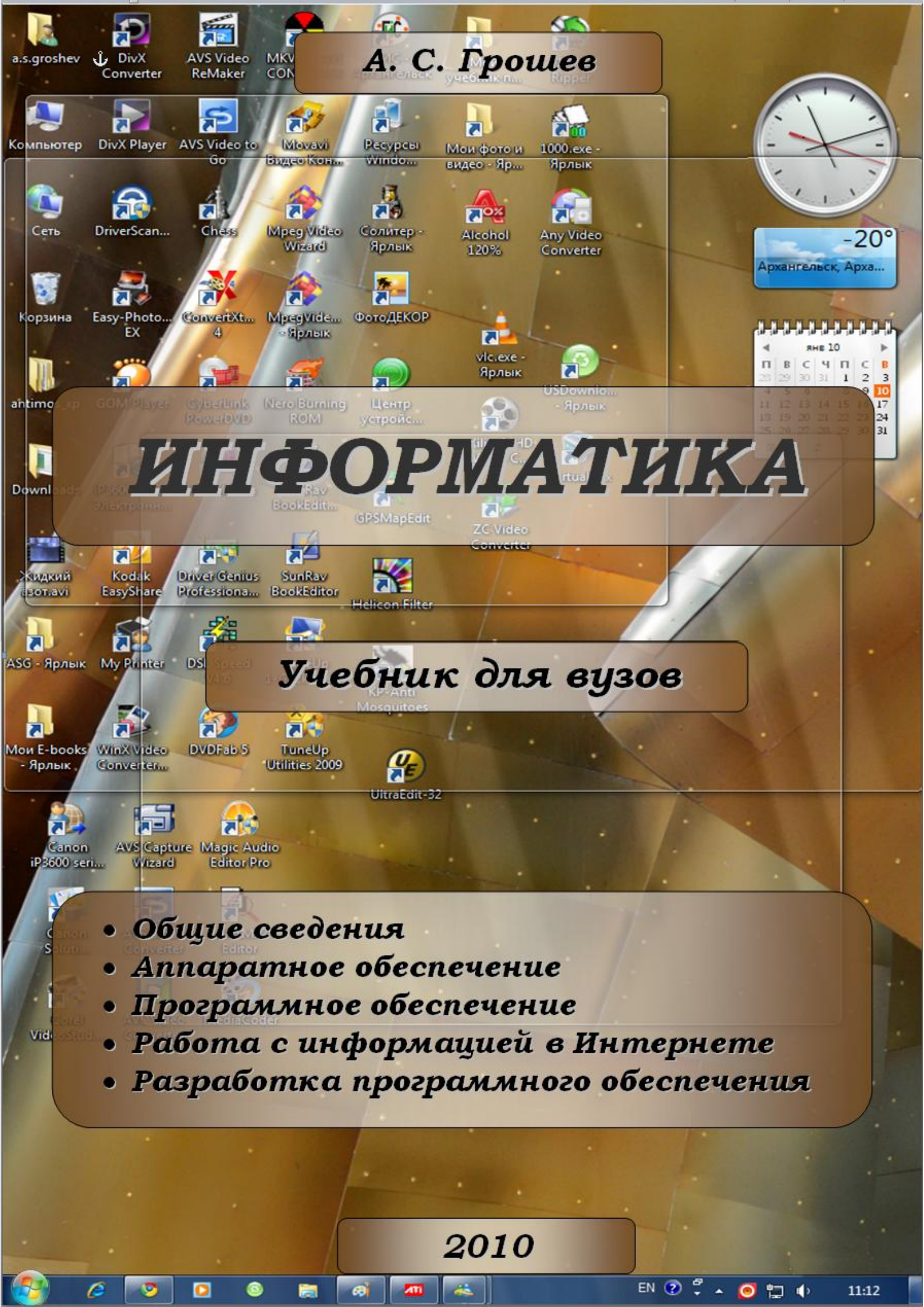

А. С. Грошев
ИНФОРМАТИКА
Учебник для вузов
Общие сведения
Аппаратное обеспечение
Программное обеспечение
Работа с информацией в Интернете
Разработка программного обеспечения
2010

А. С. Грошев

Информатика

Учебник для вузов

Рекомендован государственным образовательным учреждением высшего профессионального образования «Московский государственный технический университет имени Н. Э. Баумана» в качестве учебника для студентов высших учебных заведений, обучающихся по направлению подготовки 200500 «Стандартизация, сертификация и метрология» по специальностям 200501.65 «Метрология и метрологическое обеспечение», 140104.65 «Промышленная теплоэнергетика», 130504.65 «Бурение нефтяных и газовых скважин», 270205.65 «Автомобильные дороги и аэродромы», 250403.65 **«**Технология деревообработки». *Регистрационный номер рецензии 565 от 17.11.2009 г. МГУП*

Архангельск
2010

УДК 681.3 (075)
ББК 32.973.233
Г 89

Грошев А. С.
Г89 Информатика: Учебник для вузов / А.С. Грошев. – Архангельск, Арханг. гос. техн. ун-т, 2010. – 470 с.
ISBN 978-5-261-00480-6

Структура и содержание учебника полностью соответствуют примерной программе дисциплины «Информатика» для студентов вузов, рекомендованной Министерством образования. В главе 1 книги рассматриваются общие вопросы, глава 2 посвящена аппаратному, 3 – программному обеспечению персональных компьютеров, глава 4 содержит сведения об Интернете, глава 5 – основы программирования на языках VBS и VBA.

Особенностью учебника является практическая направленность изложения материала, призванная помочь студенту в совершенстве овладеть навыками использования персонального компьютера в профессиональной деятельности и в быту для работы со всеми видами информации.

Ил. 227. Табл. 70.

ISBN 978-5-261-00480-6
© Архангельский государственный технический университет, 2010

ОГЛАВЛЕНИЕ

Введение 8
Глава 1. Общие сведения об информационных процессах 9
1.1 Понятие информации, ее виды и свойства 9
1.2 Кодирование информации 15
1.2.1 Кодирование текстовой информации 15
1.2.2 Кодирование числовой информации 22
1.2.3 Кодирование изображений 25
1.2.4 Кодирование звуковой информации 27
1.2.5 Кодирование видеоинформации 29
1.2.6 Сжатие (архивация) различных видов информации 32
1.3 Законодательство Российской Федерации о защите компьютерной информации 38
1.3.1 Уголовный Кодекс РФ о преступлениях в сфере компьютерной информации 38
1.3.2 Законодательство РФ о защите программ для ЭВМ, как объекта авторского права 39
1.3.3 Федеральный закон РФ «Об информации, информационных технологиях и о защите информации» 40
1.4 Требования к организации рабочих мест пользователей ПК 41
Контрольные вопросы к главе 1 44
Глава 2. Аппаратное обеспечение персональных компьютеров 45
2.1 Процессор 46
2.2 Чипсет 57
2.3 Материнская плата 61
2.4 Оперативная память 69
2.5 Устройства хранения информации 73
2.5.1 Винчестер 73
2.5.2 Дисководы оптических дисков 78
2.5.3 Флэш-память 88
2.5.4 Стример 91
2.6. Устройства ввода информации 94
2.6.1. Клавиатура и мышь 94
2.6.2. Сканер 96
2.6.3. Дигитайзер, графический планшет 99
2.6.4. Сенсорный монитор 100
2.6.5. Музыкальные устройства ввода 102
2.6.6. WEB-камера 102
2.7. Устройства вывода информации 103

2.7.1. Видеоадаптер 103
2.7.2. Монитор 106
2.7.3. Принтер 110
2.7.4. Плоттер 116
2.7.5. Мультимедиапроектор 118
2.7.6. Устройства вывода звука 120
2.8 Оборудование компьютерных сетей 123
2.9 Оборудование беспроводных сетей 129
2.10. Дополнительное оборудование 130
Контрольные вопросы к главе 2 132
Глава 3. Программное обеспечение 133
3.1 Операционные системы 136
3.1.1 История развития операционных систем для ПК 137
3.1.2 Файловые системы 142
3.1.3 Операционные системы Windows Vista и Windows 7 149
3.2 Сервисные программы 190
3.2.1 Сервисные программы корпорации Symantec 191
3.2.2 Защита от вирусов 192
3.2.3 Архивация файлов 202
3.2.4 Работа с оптическими дисками 203
3.2.5 Создание и просмотр специальных форматов документов 205
3.3 Microsoft Office System 207
3.3.1 Общая характеристика компонентов системы 207
3.3.2 Подготовка текстовых документов в системе Microsoft Office Word 214
3.3.3 Выполнение вычислений в системе Microsoft Office Excel 226
3.3.4 Работа с базами данных в системе Microsoft Office Access 236
3.3.4.1 Общие принципы разработки информационной системы 239
3.3.4.2 Общая характеристика системы Microsoft Office Access 247
3.3.4.3 Создание базы данных в Microsoft Office Access 2007 251
3.3.4.4 Разработка форм 257
3.3.4.5 Разработка отчетов 268
3.3.4.6 Использование запросов 277
Контрольные вопросы к главе 3 283
Глава 4. Работа с информацией в Интернете 285
4.1 История Интернета 285
4.2 Структура Интернета 288
4.3 Адресация в Интернете 297
4.4 Способы подключения к Интернету конечных пользователей 305
4.5. Поиск информации в Интернете 307

4.6 Электронная почта в Интернете 311
4.7 RRS-каналы 315
4.8 Общение в Интернете в реальном времени 317
4.8.1 Службы мгновенных сообщений 318
4.8.2 Коллективное виртуальное общение 320
4.8.3 IP-телефония 323
4.9 Интернет-радио и Интернет-телевидение 325
4.10 Электронная коммерция 328
4.11 Обеспечение конфиденциальности информации в Интернете 334
4.12 Основы создания Web-страниц 337
Контрольные вопросы к главе 4 342
Глава 5. Основы разработки программного обеспечения 343
5.1 Проектирование программного обеспечения 344
5.1 Классификация языков программирования 350
5.2 Алгоритмизация 357
5.3 Программирование на языке Visual Basic 361
5.3.1 Язык Microsoft Visual Basic Scripting Edition 361
5.3.1.1 Алфавит языка программирования 365
5.3.1.2 Основные правила образования лексем 367
5.3.1.3 Типы данных 368
5.3.1.4 Константы 370
5.3.1.5 Переменные и массивы 371
5.3.1.6 Управляющие структуры в языке VBScript 375
5.3.1.7 Процедуры и функции пользователя в языке VBScript 384
5.3.1.8 Работа с числовой информацией 387
5.3.1.9 Работа со строковой информацией 388
5.3.1.10 Работа с информацией дата и время 389
5.3.1.11 Работа с логическими выражениями 392
5.3.1.12 Работа с объектами 394
5.3.1.13 Использование скриптов на HTML-страницах 409
5.3.2 Язык Microsoft Visual Basic for Applications 412
5.3.2.1 Программирование на языке VBA в Microsoft Office Word 415
5.3.2.2 Программирование на языке VBA в Microsoft Office Excel 426
5.3.2.3 Программирование на языке VBA в Microsoft Office Access 439
Контрольные вопросы к главе 5 450
Список литературы 452
Сайты Интернет-ресурсов 453
Приложение 1. Национальные кодовые страницы (CP) Windows 454
Приложение 2. Некоторые многоядерные процессоры Intel для настольных ПК и их цены 456
Приложение 3. Различия выпусков ОС Windows 7 457

Приложение 4. Использование галереи стилей в Excel 2007 458
Приложение 5. Использование пакета анализа в Excel 459
Приложение 6. Спецификации системы Microsoft Office Access 461
Приложение 7. Математические функции языка VBScript 463
Приложение 8. Строковые функции языка VBScript 465
Приложение 9. Функции работы с датой и временем языка VBScript 466
Приложение 10. Логические функции и операторы языка VBScript 467
Приложение 11. Методы и свойства объекта ADO.Recordset 468

Введение

Информатика – наука, изучающая общие свойства информации, закономерности и способы ее создания, хранения, поиска, преобразования и использования с помощью компьютерных систем.

На сайте Министерства образования и науки Российской Федерации приведена «Примерная программа по информатике и информационным технологиям» для среднего (полного) общего образования (*http://window.edu.ru/window_catalog/files/r37206/09-1-s.pdf*) и «Примерная программа дисциплины информатика», составленная в соответствии с государственными образовательными стандартами высшего профессионального образования для направлений «Технические науки», «Образование» и пр. (*http://www.edu.ru/db/portal/spe/prog/htm/mf0201.htm*). Обе программы содержат разделы:

- общие сведения об информации,
- аппаратное и программное обеспечение информационных систем,
- технологии обработки текстовой, числовой, графической и мультимедийной информации,
- работа с базами данных,
- алгоритмизация и программирование,
- локальные и глобальные компьютерные сети.

Сравнение двух программ показывает, что школьный и вузовский курсы информатики достаточно близки по содержанию. Следовательно, задача вуза – углубить знания по информатике, полученные в школе, познакомить студентов со всеми новостями в области технического и программного обеспечения информационных процессов, а также с использованием информационных технологий в их будущей профессиональной деятельности. При выполнении практических работ студент должен усовершенствовать свои навыки в работе на компьютере в процессах обработки, поиска и хранения всех видов информации.

Компьютер для грамотного специалиста должен стать привычным инструментом автоматизации его интеллектуального труда.

Глава 1. Общие сведения об информационных процессах

1.1 Понятие информации, ее виды и свойства

В литературе можно найти достаточно много определений термина **информация**, отражающих различные подходы к толкованию этого понятия.

В Федеральном законе Российской Федерации ***«Об информации, информатизации и защите информации»*** (*http://www.bnti.ru/*dbtexts/analmat/06_2003/licenz/1.htm) дается следующее определение этого термина: «информация – сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления».

Толковый словарь русского языка Ожегова приводит 2 определения слова **информация**:

1) Сведения об окружающем мире и протекающих в нем процессах, воспринимаемые человеком или специальным устройством.
2) Сообщения, осведомляющие о положении дел, о состоянии чего-нибудь. (Научно-техническая и газетная информации, средства массовой информации – печать, радио, телевидение, кино).

В ***информатике*** наиболее часто используется следующее определение этого термина:

Информация – это осознанные сведения об окружающем мире, которые являются объектом хранения, преобразования, передачи и использования. *Сведения* – это знания, выраженные в сигналах, сообщениях, известиях, уведомлениях и т.д.

Информация и ее свойства являются объектом исследования целого ряда научных дисциплин, таких как ***теория информации*** (математическая теория систем передачи информации), ***кибернетика*** (наука о связи и управлении в машинах и животных, а также в обществе и человеческих существах), ***семиотика*** (наука о знаках и знаковых системах), ***теория массовой коммуникации*** (исследование средств массовой информации и их влияния на общество), ***информатика, соционика*** (теория информационного метаболизма индивидуальной и социальной психики), ***информодинамика*** (наука об открытых информационных системах), ***информациоло-***

гия (наука о получении, сохранении и передаче информации для различных множеств объектов) и т.д.

Каждого человека в мире окружает море информации различных видов. Стремление зафиксировать, сохранить надолго свое восприятие информации было всегда свойственно человеку. Мозг человека хранит множество информации, и использует для хранения ее свои способы, основа которых – двоичный код, как и у компьютеров. Человек всегда стремился иметь возможность поделиться своей информацией с другими людьми и найти надежные средства для ее передачи и долговременного хранения. Для этого в настоящее время изобретено множество способов хранения информации на внешних (относительно мозга человека) носителях и ее передачи на огромные расстояния.

Основные **виды информации** по ее форме представления, способам ее кодирования и хранения, что имеет наибольшее значение для информатики, это:

- **графическая** или **изобразительная** – первый вид, для которого был реализован способ хранения информации об окружающем мире в виде наскальных рисунков, а позднее в виде картин, фотографий, схем, чертежей на бумаге, холсте, мраморе и др. материалах, изображающих картины реального мира;
- **звуковая** – мир вокруг нас полон звуков и задача их хранения и тиражирования была решена с изобретение звукозаписывающих устройств в 1877 г. (см., например, историю звукозаписи на сайте – *http://radiomuseum.ur.ru/index9.html*); ее разновидностью является **музыкальная** информация – для этого вида был изобретен способ кодирования с использованием специальных символов, что делает возможным хранение ее аналогично графической информации;
- **текстовая** – способ кодирования речи человека специальными символами – буквами, причем разные народы имеют разные языки и используют различные наборы букв для отображения речи; особенно большое значение этот способ приобрел после изобретения бумаги и книгопечатания;
- **числовая** – количественная мера объектов и их свойств в окружающем мире; особенно большое значение приобрела с развитием торговли, экономики и денежного обмена; аналогично текстовой информации для ее отображения используется метод кодирования специальными символами – цифрами, причем системы кодирования (счисления) могут быть разными;

– **видеоинформация** – способ сохранения движущихся картин окружающего мира, появившийся с изобретением кино.

Существуют также виды информации, для которых до сих пор не изобретено способов их кодирования и хранения – это **тактильная** информация, передаваемая ощущениями**, органолептическая**, передаваемая запахами и вкусами и другие виды, для которых современная наука даже не нашла признанных всеми терминов определения (например, экстрасенсорная информация).

Для передачи информации на большие расстояния первоначально использовались кодированные световые сигналы, с изобретением электричества – передача закодированного определенным образом сигнала по проводам, позднее – с использованием радиоволн.

Создателем ***общей теории информации*** и основоположником цифровой связи считается Клод Шеннон (Claude Shannon). Всемирную известность ему принес фундаментальный труд 1948 года – «Математическая теория связи» (A Mathematical Theory of Communication), в котором впервые обосновывается возможность применения двоичного кода для передачи информации.

Хранение информации при использовании компьютеров осуществляется на магнитных дисках или лентах, на лазерных дисках (**CD** и **DVD**), специальных устройствах энергонезависимой памяти (флэш-память и пр.). Эти методы постоянно совершенствуются, изобретаются новые устройства и носители информации. Обработку информации (воспроизведение, преобразование, передача, запись на внешние носители) выполняет процессор компьютера. С помощью компьютера возможно создание и хранение новой информации любых видов, для чего служат специальные программы, используемые на компьютерах, и устройства ввода информации.

Особым видом информации в настоящее время можно считать информацию, представленную в глобальной сети **Интернет**. Здесь используются особые приемы хранения, обработки, поиска и передачи распределенной информации больших объемов и особые способы работы с различными видами информации.

Постоянно совершенствуется программное обеспечение ведущих разработчиков (например, *Microsoft Corporation*), обеспечивающее коллективную работу с корпоративной информацией всех видов.

Свойства информации

Как и всякий объект, информация обладает свойствами. Характерной отличительной особенность информации от других объектов природы и

общества, является дуализм: на свойства информации влияют как свойства исходных данных, составляющих ее содержательную часть, так и свойства методов, фиксирующих эту информации.

С точки зрения информатики наиболее важными представляются следующие ***общие качественные свойства***: объективность, достоверность, полнота, точность, актуальность, полезность, ценность, своевременность, понятность, доступность, краткость и пр.

1) Объективность информации. Объективный – существующий вне и независимо от человеческого сознания. Информация – это отражение внешнего объективного мира. Информация объективна, если она не зависит от методов ее фиксации, чьего-либо мнения, суждения.

Пример. Сообщение «На улице тепло» несет субъективную информацию, а сообщение «На улице 22^{o}C» – объективную, но с точностью, зависящей от погрешности средства измерения.

Объективную информацию можно получить с помощью измерительных приборов. Отражаясь в сознании конкретного человека, информация перестает быть объективной, так как, преобразовывается (в большей или меньшей степени) в зависимости от мнения, суждения, опыта, знаний конкретного субъекта.

2) Достоверность информации. Информация достоверна, если она отражает истинное положение дел. Объективная информация всегда достоверна, но достоверная информация может быть как объективной, так и субъективной. Достоверная информация помогает принять нам правильное решение.

Недостоверной информация может быть по следующим причинам:

- преднамеренное искажение (дезинформация) или непреднамеренное искажение субъективного свойства;
- искажение в результате воздействия помех и недостаточно точных средств измерений.

3) Полнота информации. Информацию можно назвать полной, если ее достаточно для понимания и принятия решений. Неполная информация может привести к ошибочному выводу или решению.

4) Точность информации определяется степенью ее близости к реальному состоянию объекта, процесса, явления (погрешностью средства измерения).

5) Актуальность информации – важность для настоящего времени, злободневность, насущность. Иногда только вовремя полученная информация может быть полезна.

6) Полезность (ценность) информации. Полезность может быть оценена применительно к нуждам конкретных ее потребителей и оценивается по тем задачам, которые можно решить с ее помощью.

Самая ценная информация – объективная, достоверная, полная, и актуальная. При этом следует учитывать, что и необъективная, недостоверная информация (например, художественная литература), имеет большую значимость для человека.

Социальная (общественная) информация обладает еще и дополнительными свойствами:

- имеет семантический (смысловой) характер, т. е. понятийный, так как именно в понятиях обобщаются наиболее существенные признаки предметов, процессов и явлений окружающего мира.
- имеет языковую природу (кроме некоторых видов эстетической информации, например изобразительного искусства). Одно и то же содержание может быть выражено на разных естественных (разговорных) языках, записано в виде математических формул и т. д.

С течением времени количество информации растет, информация накапливается, происходит ее систематизация, оценка и обобщение. Это свойство назвали ростом и кумулированием информации. (Кумуляция – от лат; cumulatio – увеличение, скопление).

Старение информации заключается в уменьшении ее ценности с течением времени. Старит информацию появление новой информации, которая уточняет, дополняет или отвергает полностью или частично более раннюю. Научно-техническая информация стареет быстрее, эстетическая (произведения искусства) – медленнее.

Логичность, компактность, удобная форма представления облегчает понимание и усвоение информации.

Количественные характеристики информации

Измерение информации в технике

В технике (теория кодирования и передачи сообщений) под количеством информации понимают количество кодируемых, передаваемых или хранимых символов. При этом используют простой способ определения количества информации, который может быть назван объемным. Он основан на подсчете числа символов в сообщении, т. е. связано с его длиной и не учитывает содержания.

В вычислительной технике применяются две стандартные единицы измерения: бит (binary digit) и байт (byte).

Бит – минимальная единица измерения информации – величина, которая может принимать одно из двух значений (в математическом представлении 0 или 1).

Байт – единица количества информации в системе СИ. Байт - восьмиразрядный двоичный код, с помощью которого наиболее часто представляют один символ текста.

Информационный объем сообщения (информационная емкость сообщения) – количество информации в сообщении, измеренное в битах, байтах или производных единицах (Кбайтах, Мбайтах и т. д.).

1 килобайт (Кбайт) = 2^{10} байт = 1024 байт,
1 мегабайт (Мбайт) = 2^{10} Кбайт = 1024 Кбайт = 1048576 байт,
1 гигабайт (Гбайт) = 2^{10} Мбайт = 1024 Мбайт = 1073741824 байт,
1 терабайт (Тбайт) = 2^{10} Гбайт = 1024 Гбайт = 1099511627776 байт
и т. д.

Измерение информации в теории информации (информация как снятая неопределенность)

В теории информации количеством информации называют числовую характеристику сигнала, не зависящую от его формы и содержания и характеризующую неопределенность, которая исчезает после получения сообщения в виде данного сигнала – в этом случае количество информации зависит от вероятности получения сообщения о том или ином событии. Оценка количества информации основывается на законах теории информации.

Клод Шеннон ввел понятие информационной энтропии – меры хаотичности информации, определяющей неопределённость появления какого-либо символа первичного алфавита. При отсутствии информационных потерь численно равна количеству информации на символ передаваемого сообщения.

Информационная энтропия для независимых случайных событий *x* с *n* возможными состояниями (от 1 до *n*) рассчитывается по формуле Шеннона:

$$H(x) = -\sum_{i=1}^{n} p(i) \log_2 p(i)$$

где *p(i)* – вероятность i-го события.

Информация, в том числе графическая и звуковая, может быть представлена в аналоговой и дискретной форме. При аналоговом представлении информации физическая величина может принимать бесконечное

множество значений. При дискретном представлении информации физическая величина может принимать конечное множество значений, при этом она изменяется скачкообразно. Преобразование графической и звуковой информации из аналоговой формы в дискретную производится путем дискретизации, т. е. разбиения непрерывного графического изображения или непрерывного (аналогового) звукового сигнала на отдельные элементы. В процессе дискретизации производится кодирование, т. е. присвоение каждому элементу конкретного значения в форме кода.

1.2 Кодирование информации

Вся информация, которую хранит, обрабатывает и передает по сетям компьютер, представлена в виде двоичных чисел. Существуют международные стандарты и методы кодирования текстовой, числовой, изобразительной, звуковой и видеоинформации. Знание основных кодовых таблиц очень важно для правильного чтения информации Интернета, электронной почты, текстовых документов в кодировке различных операционных систем.

1.2.1 Кодирование текстовой информации

Для кодирования букв и других символов, используемых в печатных документах, необходимо закрепить за каждым символом числовой номер – код.

В англоязычных странах используются 26 прописных и 26 строчных букв (A … Z, a … z), 9 знаков препинания (., : ! " ; ? ( )), пробел, 10 цифр, 5 знаков арифметических действий (+,-,*, /, ^) и специальные символы (№, %, _, #, $, &, >, <, |, \) – всего чуть больше 100 символов. Таким образом, для кодирования этих символов можно ограничиться 7-разрядным двоичным числом (от 0 до 1111111, в десятичной системе счисления – от 0 до 127).

Первой такой 7-разрядной кодовой таблицей была ***ASCII*** (American Standard Code for Information Interchange), опубликованная как стандарт в 1963 г. американской организацией по стандартизации American Standards Association (ASA), которая позднее стала именоваться ***ANSI*** (American National Standards Institute, *http://www.ansi.org/*, поэтому данную кодовую таблицу называют также и ***ANSI***). Таблица содержала 32 кода команд или управляющих символов (от 0 до 31), большая часть которых сегодня не используется, и 95 кодов (от 33 до 127) для различных знаков, достаточных

для работы с английскими текстами, как показано на рисунке 1.1. На рисунке 1.1 символы построчно имеют следующие коды в шестнадцатеричной системе счисления (в скобках – в десятичной):

- 1-я строка с 00 по 0F и далее с 10 по 1F (0 – 15, 16 - 31),
- 2-я строка с 20 по 2F и 30 – 3F (32 – 47, 48 - 63),
- 3-я строка с 40 по 4F и 50 – 5F (64 – 79, 80 -95),
- 4-я строка с 60 по 6F и 70 – 7F (96 – 111, 112 -127).

В данной таблице для преобразования прописных букв в строчные достаточно к коду букву прибавить 32 и, наоборот, для преобразования строчных в прописные из кода буквы вычесть 32.

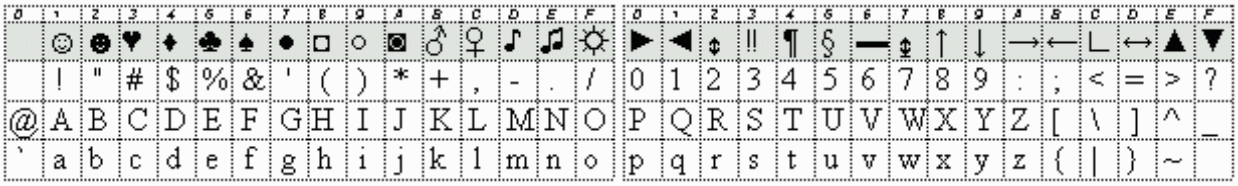


Рисунок 1.1. 7-битная кодовая таблица ***ASCII*** (***ANSI***)

В последующем данная таблица ***ASCII*** была принята как стандарт ведущими международными организациями по стандартизации:

ISO/IEC 646:1991 (***ISO*** – *http://www.iso.org/* – International Organization for Standardization и ***IEC*** – *http://www.iec.ch/* – International Electrotechnical Commission – ведущие международные организации по стандартизации, в области электротехники – совместные стандарты), ***ITU-T Recommendation T.50 (09/92)*** (The International Telecommunication Union – *http://www.itu.int/*), ***ECMA*-6** (European Computer Manufacturers Association).

Однако для нашей страны и многих других стран необходимо было добавить в кодовую таблицу символы национальных алфавитов. Для этого было предложено использовать 8-битную кодовую таблицу, которая могла содержать дополнительно еще 128 символов (с 128 по 255).

В дальнейшем был принят стандарт на 8-битную таблицу ***ASCII – ISO/IEC 8859***, в которой первые 128 символов оставались те же, что и в 7-битной таблице, а символы с 128 по 255 отводились для неанглийских символов.

Существует несколько частей этого стандарта:

- ***ISO/IEC 8859-1:1998*** – Part 1: Latin alphabet No. 1,
- ***ISO/IEC 8859-5:1999*** – Part 5: Latin/Cyrillic alphabet,
- ***ISO/IEC 8859-6:1999*** – Part 6: Latin/Arabic alphabet,
- ***ISO/IEC 8859-7:2003*** – Part 7: Latin/Greek alphabet,
- ***ISO/IEC 8859-8:1999*** – Part 8: Latin/Hebrew alphabet и т. д.

На рисунке 1.2 представлена вторая половина кодовой таблицы (коды 128-255) для стандарта ***ISO 8859-5*.**

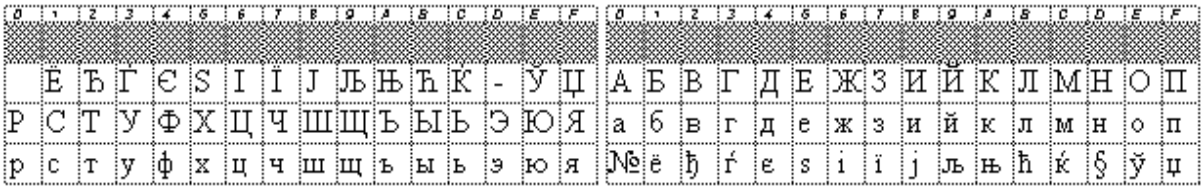


Рисунок 1.2. Кодовая таблица ***ISO 8859-5*** (коды с 128_{10} по 255_{10})

Первые русские ЭВМ использовали 7-битную кодировку символов ***КОИ-7*** (Код Обмена Информацией семибитный – рисунок 1.3), в которой присутствовали прописные латинские буквы, а на месте строчных латинских были русские прописные буквы (кириллица).

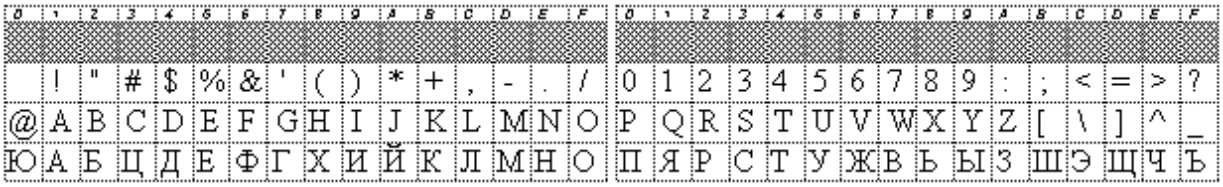


Рисунок 1.3. 7-битная таблица символов ***КОИ-7***

Позднее на первых отечественных персональных компьютерах использовалась так называемая «Основная кодировка ВЦ Академии Наук СССР», в руководствах к старым матричным принтерам обозначаемая просто как **«ГОСТ»** – 8-битная кодовая таблица, вторая половина которой содержала символы псевдографики, русские прописные и строчные буквы (коды с 128 по 255 – рисунок 1.4).

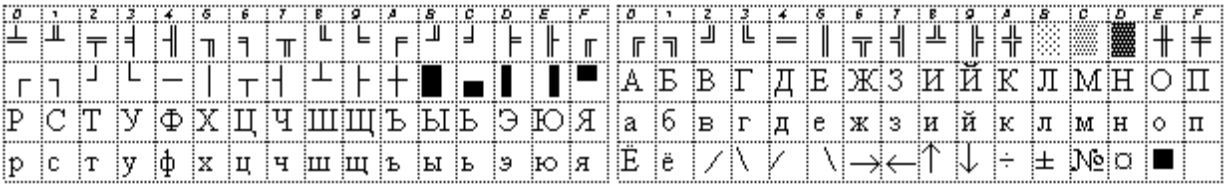


Рисунок 1.4. Кодировка символов ***«ГОСТ»*** (коды с 128_{10} по 255_{10})

В дальнейшем основной кодировкой русских букв для первой операционной системы ПК – MS DOS стала «Альтернативная кодировка ВЦ Академии Наук СССР» (вторая половина таблицы для кодов 128-255 приведена на рисунке 1.5). Содержит псевдографику (позволяющую в текстовом режиме рисовать рамки из одинарных и двойных линий). Существует несколько модификаций, отличающихся символами в последних 14 позициях. Зарегистрирована в IANA (*Internet Assigned Numbers Authority* – ор-

ганизация, отвечающая за административное управление в Internet) как IBM866 или ***CP866***.

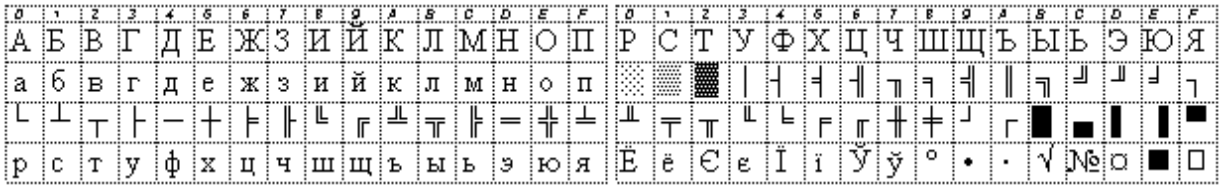


Рисунок 1.5. Таблица символов DOS Cyrillic (***CP866,*** коды с 128_{10} по 255_{10})

С широким распространением операционной системы Microsoft Windows и появлением ее национальных локализаций для второй половины таблицы ***ASCII*** было введено понятие «кодовая страница» (code page, **CP**). Полная таблица кодовых страниц Microsoft Windows приведена в Приложении 1. Несколько наиболее важных кодировок показаны в таблице 1.1.

Таблица 1.1. Некоторые национальные кодовые страницы (CP) операционной системы Microsoft Windows

Локализация	ANSI CP	Mac CP	DOS (OEM) Primary CP	DOS (OEM) Secondary CP
English (United States)	1252	10000	437	850
English (Britain, Canada и др.)	1252	10000	850	437
French (Standard)	1252	10000	850	437
German (Standard)	1252	10000	850	437
Russian	***1251***	***10007***	***866***	**855**

Кодировка русских букв в операционной системе Microsoft Windows – **CP1251** – показана на рисунке 1.6.

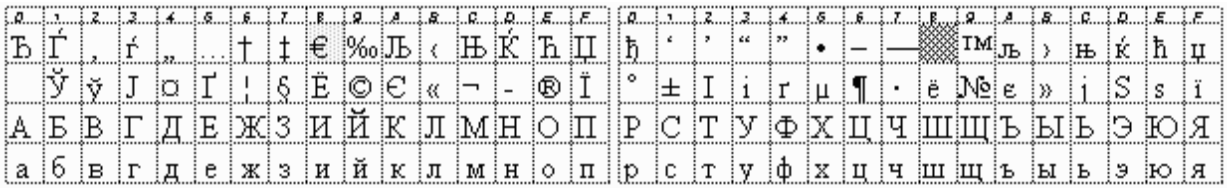


Рисунок 1.6. Таблица символов **CP1251** (коды с 128_{10} по 255_{10})

В операционной системе Linux для представления русских букв используется кодировка ***КОИ-8R*** (рисунок 1.7), зарегистрированная в IANA как ***KOI8-R*** (*http://koi8.pp.ru/*).

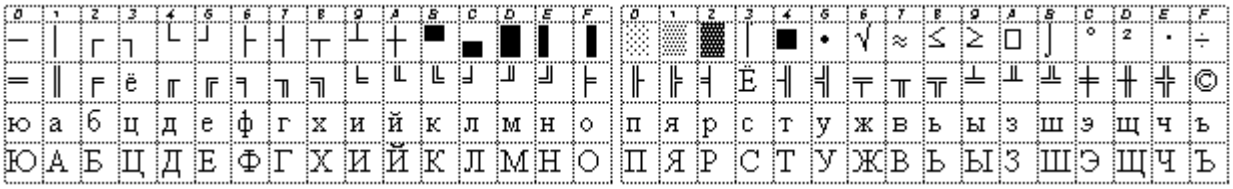


Рисунок 1.7. Таблица символов ***KOI8-R*** (коды с 128_{10} по 255_{10})

Стандартизован и зарегистрирован также украинский клон ***КОИ-8*** – ***KOI8-U*** (рисунок 1.8), имеющий отличия от ***KOI8-R*** во второй строке символов псевдографики.

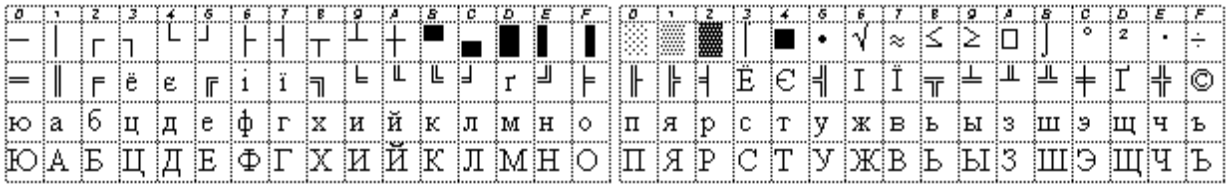


Рисунок 1.8. Таблица символов ***KOI8-U*** (коды с 128_{10} по 255_{10})

Кириллица Macintosh (компьютеров фирмы Apple), она же **CP10007,** довольно близка к **CP1251**. Не зарегистрирована в IANA, но часто обозначается как **x-mac-cyrillic** (рисунок 1.9).

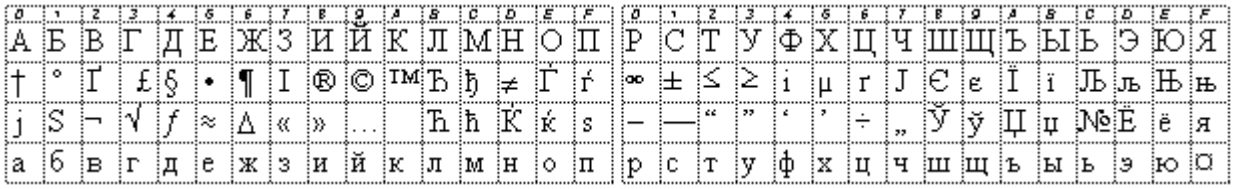


Рисунок 1.9. Macintosh Cyrillic, **CP10007** (коды с 128_{10} по 255_{10})

В 1991 году в Калифорнии была создана некоммерческая организация **Unicode Consortium**, в которую входят представители многих компьютерных фирм (Borland, IBM, Lotus, Microsoft, Novell, Sun, WordPerfect и др.), и которая занимается развитием и внедрением стандарта **«The Unicode Standard»**.

Стандарт кодирования символов **Unicode** становится доминирующим в интернациональных программных многоязычных средах. Microsoft Windows NT и его потомки Windows 2000, 2003, XP, Vista используют **Unicode**, точнее **UTF-16**, как внутреннее представление текста. UNIX-подобные операционные системы типа Linux, BSD и Mac OS X приняли **Unicode** (**UTF-8**), как основное представления многоязычного текста.

Unicode резервируют 1114112 ($2^{20}+2^{16}$) символов кода, в настоящее время используются более 96000 символов. Первые 256 кодов символов точно соответствуют таковым ***ISO 8859-1***, наиболее популярной 8-разрядной таблицы символов «западного мира»; в результате, первые 128

символов также идентичны таблице ***ASCII*.** На рисунке 1.10 показан русский блок **Unicode** (коды от 0400_{16} до $04FF_{16}$).

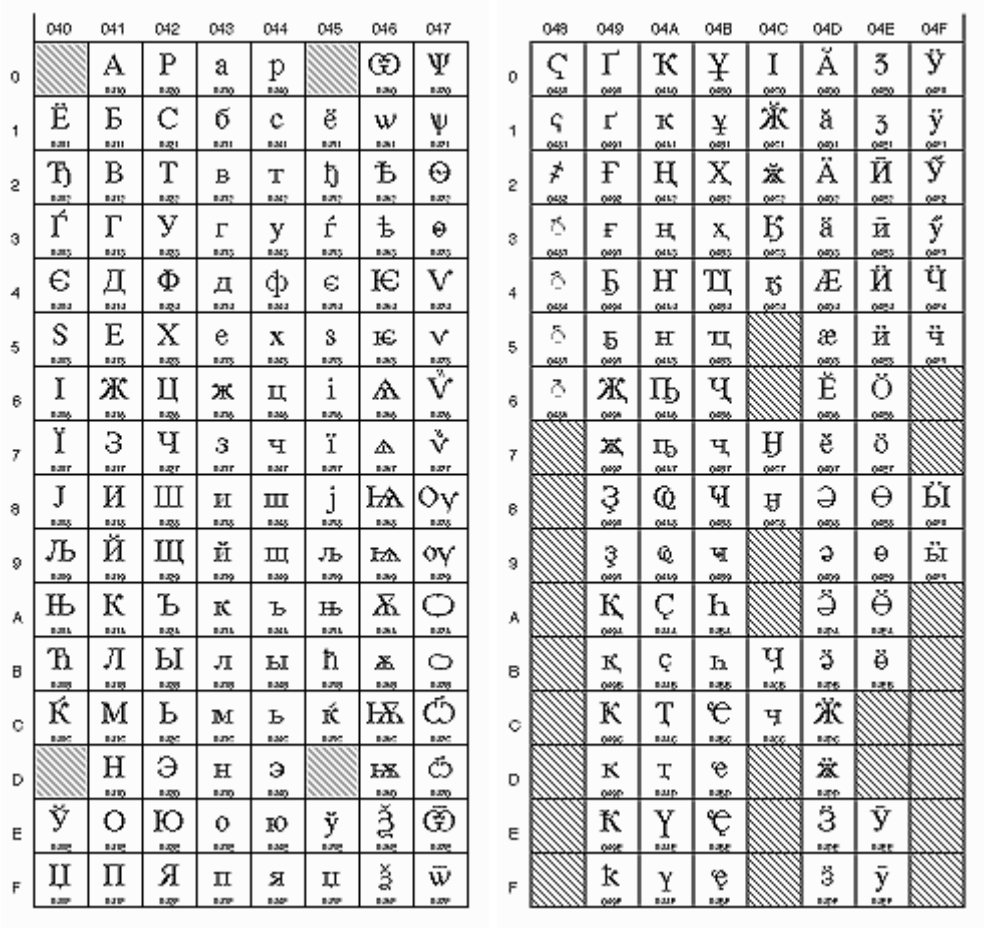


Рисунок 1.10. Кириллица в **Unicode**

Кодовое пространство стандарта **Unicode** разделено на 17 планов («planes»), и каждый план имеет 65536 (= 2^{16}) точек кода. Первый план (план 0), Основной Многоязычный План (BMP – *Basic Multilingual Plane*) – тот, в котором описано большинство символов. BMP содержит символы почти для всех современных языков, и большое количество специальных символов.

Еще два плана используются для «графических» символов. План 1, Дополнительный Многоязычный План (SMP – *Supplementary Multilingual Plane*) главным образом используется для исторических символов, а также используется для музыкальных и математических символов. План 2, *Supplementary Ideographic Plane* (SIP), используется для приблизительно 40000 редких китайских иероглифов. План 15 и План 16 открыт для любого частного использования.

В то же время, подобно двоичным файлам, кодировка **Unicode** мало подходит для непосредственной передачи по сети – байты в тексте вполне могут приходиться на область управляющих символов, поэтому обычно применяются две другие основанные на **Unicode** кодировки переменной длины, обозначаемые как **UTF** (**Unicode Transformation Format**): 7-битная **UTF-7** (последний пересмотр – *RFC2152*, 1997 г., зарегистрирована в IANA как **UTF-7**) и 8-битная **UTF-8** (*RFC2279*, 1998 г., зарегистрирована в IANA как **UTF-8**). Обе они в каком-то смысле уже не являются языковыми кодировками, а являются программно распознаваемым кодом относительно исходного **Unicode**, но зарегистрированы они именно как кодировки, наравне с ***ISO 8859-5*** или ***KOI8-R***. Естественно, обе они не являются специфически «русскими», а пригодны для написания «сколько угодно»-язычного письма.

В **UTF-8** все символы разделены на несколько групп по значению первых битов. Символы с кодами менее 128_{10} кодируются одним байтом, первый битом которого равен нулю, а последующие 7 бит в точности соответствуют 128 символам 7-й таблицы ***ASCII*** (см. таблицу 1.2), следующие 1920 символов – двумя байтами (*Greek, Cyrillic, Coptic, Armenian, Hebrew* и *Arabic* символы). Последующие символы кодируются тремя и четырьмя байтами.

Таблица 1.2. Принцип кодирования символов в **UTF-8**

Диапазон кодов (*hexadecimal*)	UTF-8 (*binary*)	Notes
000000 - 00007F	0xxxxxxx	Первый бит 0, следующие 7 соответствуют таблице **ASCII**
000080 - 0007FF	110xxxxx 10xxxxxx	Первые 3 бита 110 – всего используется 2 байта, второй байт начинается с 10
000800 - 00FFFF	1110xxxx 10xxxxxx 10xxxxxx	Первые 4 бита 1110 – всего используется 3 байта, второй и третий байты начинаются с 10
010000 - 10FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	Первые 5 бит 11110 – всего используется 4 байта, второй, третий и четвертый байты начинаются с 10

Особняком стоит 7-битная, русская кодировка – **транслитерация**, или **транскириллица**, когда русские буквы передаются похожими по звучанию английскими primerno takim obrazom.

В конце 1997 г. *Microsoft* подвергла ревизии свои кодовые таблицы и включила в них новый символ евровалюты "Евро" (рисунок 1.11), он помещен в позицию 128 (0x80) большинства таблиц и в позицию 136 (0x88) русской таблицы ***CP1251***.

Рисунок 1.11. Символ "Евро"

Таким образом, в настоящее время при работе в Интернете Вы можете встретить следующие кодировки для кириллицы:

- ✓ ***CP1251*** – Cyrillic Windows – операционной системы Microsoft Windows;
- ✓ ***CP866*** – Cyrillic DOS – операционной системы MS DOS;
- ✓ ***ISO 8859-5*** – Cyrillic ISO – 8-ми битная таблица **ASCII** ;
- ✓ ***KOI8-R*** – операционной системы Linux;
- ✓ ***CP10007*** – операционной системы компьютеров Macintosh;
- ✓ **UTF-8** – универсальная **Unicode** кодировка переменной длины.

1.2.2 Кодирование числовой информации

Числовая информация, как и любая другая, хранится и обрабатывается в компьютерах в двоичной системе счисления – числа представляются в виде последовательностей нулей и единиц.

Существуют два вида чисел и два способа их представления: форма с фиксированной точкой и форма с плавающей точкой. Форма с фиксированной точкой применяется для целых чисел, форма с плавающей точкой – для вещественных (действительных) чисел.

Как это ни странно, не все студенты 1 курса вуза могут ответить на вопрос, что такое действительные или вещественные числа. Это рациональные и иррациональные числа, у которых может быть как целая, так и дробная часть, записываемая справа от разделителя целой и дробной части. Как разделитель при работе на компьютере раньше всегда использовалась точка, но в современных системах разделитель – точка или запятая – может настраиваться в соответствии со стандартами страны пользователя или с его привычками.

ЭВМ оперирует с числами, содержащими конечное число двоичных цифр (разрядов). Количество разрядов ограничено длиной разрядной сетки машины. Под разрядной сеткой понимается совокупность двоичных разря-

дов, предназначенных для хранения и обработки машинных слов (двоичных кодов).

Количество двоичных разрядов и положение запятой в разрядной сетке машины определяют такие важные характеристики ЭВМ, как точность и диапазон представляемых чисел.

Кроме бита и байта, для указания длины формата чисел используется машинное слово, полуслово и двойное слово. Двойное слово и полуслово по-разному определяются для разных систем ЭВМ. Кроме того, может использоваться понятие ***тетрада*** – 4 двоичных разряда, которыми может кодироваться, например, одна двоичная цифра.

Двоичные разряды в форматах формируются слева направо (начиная с нулевого разряда).

Кодирование целых чисел

Целые числа в компьютере хранятся в памяти в формате с фиксированной запятой. В этом случае каждому разряду разрядной сетки соответствует всегда один и тот же разряд числа.

Целые числа без знака (положительные) – для их хранения может отводиться последовательность из 8, 16 или 32-х бит памяти. Например, максимальное 8-битное число $A_2 = 11111111_2$ будет храниться следующим образом (прямой код):

1	1	1	1	1	1	1	1

Максимальное значение целого неотрицательного числа достигается в случае, когда во всех ячейках хранятся единицы и равно 2^N-1, где N – разрядность числа.

Для 8-разрядных целых положительных чисел оно будет равно 2^8 - 1 = 255, для 16-разрядных 2^{16} - 1 = 65 535, для 32-разрядных 2^{32} - 1 = 4 294 967 295.

Целые числа со знаком (могут быть положительные и отрицательные) – при их хранении используется последовательность из 8, 16 или 32-х бит памяти, причем старший бит (первый слева) обозначает знак числа – 0 - положительное, 1 – отрицательное. При записи чисел используется не прямой, а дополнительный код двоичного числа равный 2^N – A, где N – разрядность числа, A – прямой код двоичного числа.

Дополнительным называется код, в котором для положительного числа в знаковом разряде пишется "0", в цифровых – модуль числа, а для отрицательного в знаковом разряде пишется "1", в цифровых – дополнение числа до единицы (инвертирование цифр).

Например, число -1 в 8-разрядном двоичном коде выглядит, как 11111111, -2 – как 11111110 и т. д.

Дополнительный код позволяет заменить арифметическую операцию вычитания операцией сложения, что исключает операцию вычитания их набора команд двоичной арифметики процессора.

Таким образом, при использовании 8-ми разрядов для хранения целых чисел со знаком диапазон их изменения составит от – 128 до 127, если использовать 16 разрядов – от -32 768 до 32 767, 32 разряда – от -2 147 483 648 до 2 147 483 647, что следует учитывать при работе с целыми типами данных при программировании и работе с базами данных.

Кодирование вещественных чисел

Для того чтобы представить действительное число **X** в виде набора целых чисел (двоичных – для представления в компьютерной памяти), его необходимо привести к нормализованной форме:

$$\mathbf{X = \pm M \cdot N^P};$$

где **M** – ***мантисса*** (дробная часть), **N – основание** системы счисления, а **P** – порядок числа.

Для десятичной системы счисления нормальная форма $\mathbf{X = \pm M \cdot 10^P}$, для двоичной $\mathbf{X = \pm M \cdot 2^P}$.

Например, число 22.22_{10} в таком виде будет выглядеть, как $+0{,}2222 \cdot 10^2$ (при записи чисел в памяти ЭВМ ноль и запятая отсутствуют**).**

Таким образом, действительные число на компьютерах хранится в двоичной системе счисления в виде:

где **S** – признак знака числа.

Поскольку размер памяти, отводимый под мантиссу и порядок, ограничен, то действительные числа представляются с некоторой погрешностью, определяемой количеством разрядов в мантиссе числа, и имеют определенный диапазон изменения, определяемый количеством разрядов в порядке числа.

Конкретные характеристики различных типов вещественных типов данных для ПК определены в стандарте IEEE-754-1985 (Institute of Electrical and Electronic Engineers - *http://www.ieee.org/*), согласно которому используются 3 основных формы (см. табл. 1.3).

Особенности арифметики для чисел с плавающей точкой могут существенно влиять на результаты расчётов, вплоть до того, что погреш-

ность может сделать невозможным получение какого-либо результата вообще, поэтому знание деталей представления в памяти таких чисел и реализации арифметики вещественных чисел является необходимым для программистов.

Таблица 1.3. Данные с плавающей точкой по стандарту IEEE-754-1985

Тип	Размер, бит	Диапазон изменения чисел		Точность, количество цифр в числе	Машинное ε
		максимум	минимум		
single	32	$3.4 \cdot 10^{-38}$	$3.4 \cdot 10^{38}$	6	$1{,}192 \cdot 10^{-7}$
double	64	$1.7 \cdot 10^{-308}$	$1.7 \cdot 10^{308}$	15	$2{,}221 \cdot 10^{-16}$
long double	80	$3.4 \cdot 10^{-4932}$	$3.4 \cdot 10^{4932}$	19	$1{,}084 \cdot 10^{-19}$

1.2.3 Кодирование изображений

Изображение – некоторая двумерную область, свойства каждой точки (***pixel, пиксель***) которой могут быть описаны (координаты, цвет, прозрачность…).

Множество точек называется ***растром*** (bit map, dot matrix, raster) (см. рис. 1.12), а изображение, которое формируется на основе растра, называются ***растровым***. На экране монитора всегда формируется растровое изображение, однако, для хранения может использоваться и векторное представление информация, где изображение представлено в виде набора графических объектов с их координатами и свойствами (линия, овал, прямоугольник, текст и т. п.).

Рис. 1.12. Растровое изображение на экране монитора

На мониторе и в растровых изображениях число пикселей по горизонтали и по вертикали называют ***разрешением*** (resolution). Наиболее часто используются 1024×768 или 1280×800, 1280×1024 (для 15″, 17″, 19″), 720×576 (качество обычных DVD-фильмов), 1920×1080 и 1920×720 (телевидение высокой четкости HDTV – стандарты 1080i и 720p). Каждый пиксель изображения нумеруется, начиная с нуля слева направо и сверху вниз.

Для представления цвета используются цветовые модели. ***Цветовая модель*** (color model) – это правило, по которому может быть определен цвет. Самая простая двухцветная модель – битовая. В ней для описания цвета каждого пикселя (чёрного или белого) используется всего один бит.

Для представления полноцветных изображений используются несколько более сложных моделей. Известно, что любой цвет может быть представлен как сумма трёх основных цветов: красного, зелёного и синего. Если интенсивность каждого цвета представить числом, то любой цвет будет выражаться через набор из трёх чисел. Так определяется наиболее известная цветовая ***RGB***-модель (Red-Green-Blue). На каждое число отводится один байт. Так можно представить 2^{24} цвета, то есть примерно 16,7 млн. цветов. Белый цвет в этой модели представляется как (1,1,1), чёрный – (0,0,0), красный (1,0,0), синий (0,0,1). Жёлтый цвет является комбинацией красного и зелёного и потому представляется как (1,1,0).

Цветовая модель ***RGB*** была стандартизирована в 1931 г. и впервые использована в цветном телевидении. Модель ***RGB*** является ***аддитивной*** моделью, то есть цвет получается в результате сложения базовых цветов. Существуют и другие цветовые модели, которые для ряда задач оказываются более предпочтительными, чем ***RGB***-модель. Например, для представления цвета в принтерах используется ***субтрактивная*** **CMYK**-модель (Cyan-Magenta-Yellow-blacK), цвет в которой получается в результате вычитания базовых цветов из белого цвета. Белому цвету в этой модели соответствует (0,0,0,0), чёрному - (0,0,0,1), голубому - (1,0,0,0), сиреневому - (0,1,0,0), жёлтому - (0,0,1,0). В цветовой модели ***HSV*** (Hue-Saturation-Value) цвет представляется через цвет, насыщенность и значение, а в модели ***HLS*** (Hue-Lightness-Saturation) через оттенок, яркость и насыщенность. Современные графические редакторы, как правило, могут работать с несколькими цветовыми моделями.

Кроме растрового изображения на экране монитора существуют графические форматы файлов, сохраняющие растровую или векторную графическую информацию. С такой информацией работают специальные программы, которые преобразуют векторные изображения в растровые, отображаемые на мониторе.

1.2.4 Кодирование звуковой информации

Звук можно описать в виде совокупности синусоидальных волн определённых частоты и амплитуды. Частота волны определяет высоту звукового тона, амплитуда – громкость звука. Частота измеряется в герцах (Гц, Hz). Диапазон слышимости для человека составляет от 20 Гц до 17000 Гц (или 17 кГц).

Задача цифрового представления звука сводится измерению интенсивности звука через заданный интервал времени (например, 48 раз за 0,001 секунды). Принцип такого представления изображён на рис. 1.13.

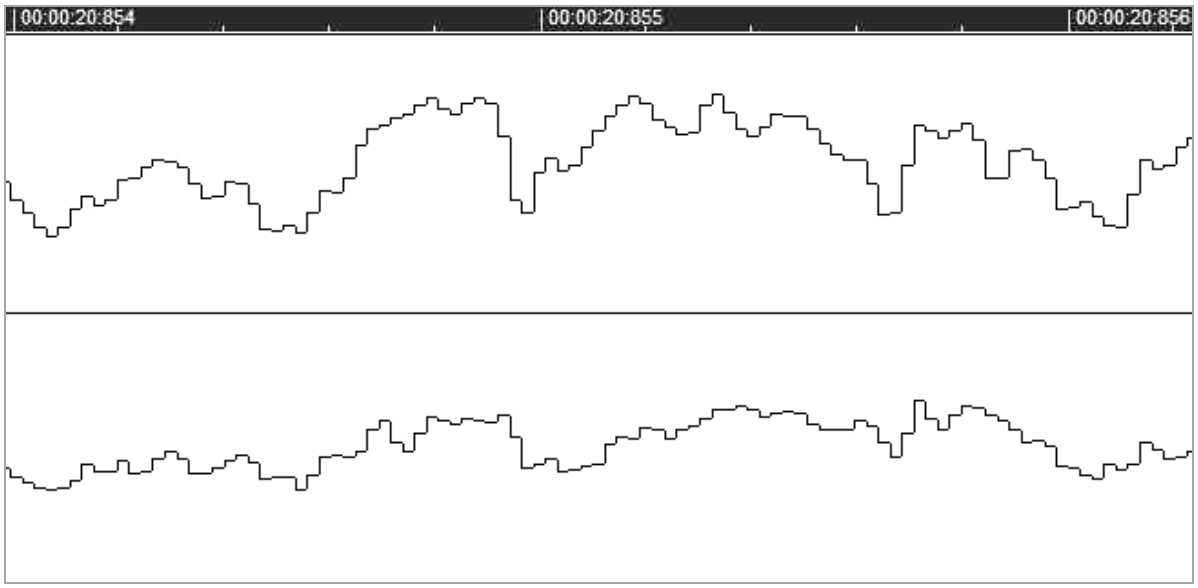


Рис. 1.13 Диаграмма стереозвука в музыкальном редакторе (верхняя ось – время ~ от 20,854 до 20,856 сек., т. е. 0,002 сек.)

Каждому измерению присваивается числовое значение амплитуды. Количество измерений в секунду называется ***частотой выборки*** (sampling rate). Количество возможных значений амплитуды называется ***точностью выборки*** (sampling size). Таким образом, звуковая волна представляется в виде ступенчатой кривой. Ширина ступеньки тем меньше, чем больше частота выборки, а высота ступеньки тем меньше, чем больше точность выборки.

Возможности наиболее распространённой современной аппаратуры предусматривают работу с частотой выборки до 48 кГц (48 тысяч раз в секунду!), что позволяет правильно описывать звук частотой до 22,05 кГц.

Непрерывная звуковая волна разбивается на отдельные участки по времени, для каждого устанавливается своя величина амплитуды. Каждой

ступеньке присваивается свой уровень громкости звука, который можно рассматривать как набор возможных состояний.

Характеристики качества звука

1) ***Точность выборки или глубина кодирования звука*** – количество бит на одно измерение величины звукового сигнала.

Современные звуковые карты обеспечивают 16-битную глубину кодирования звука. Количество уровней (градаций амплитуды) можно рассчитать по формуле:

$N = 2^I = 2^{16} = 65\ 536$ уровней сигнала (градаций амплитуды)

2) ***Частота выборки или частота дискретизации*** – это количество измерений уровня звукового сигнала за 1 секунду.

Одно измерение в 1 секунду соответствует частоте 1 Гц.

1000 измерений в 1 секунду - 1 кГц.

Количество измерений может лежать в диапазоне **от 8000 до 48 000** (8 кГц – 48 кГц).

8 кГц соответствует частоте радиотрансляции,

48 кГц – качеству звучания аудио-CD.

Опыт показывает, что точное соответствие цифрового сигнала аналоговому достигается, если частота дискретизации будет вдвое выше максимальной звуковой частоты, то есть составит не менее 40 кГц.

На практике значения частоты дискретизации, применяемые в звуковых системах, равны 44,1 кГц или 48 кГц.

Чем больше частота дискретизации, тем качественнее звук.

3) Для характеристики сжатого звука и видео используется понятие ***битрейт*** – количество единиц информации, необходимых для хранения или передачи одной секунды потока данных. Величина измеряется в килобитах в секунду (kbps). Битрейт характеризует как плотность упаковки информации, так и её качество. Например, из двух MP3 файлов сжатых с разным битрейтом, более качественный (близкий к оригиналу) звук будет у файла с большим битрейтом. В тоже время, файл другого формата, при равном битрейте, может дать как лучшее, так и худшее качество звука. Стандартов кодирования двухканальной и многоканальной (5.1 и 7.1) аудиоинформации насчитывается несколько десятков, наименования некоторых из них, используемых в современных методах записи мультимедиаинформации, приведены далее в таблице 1.4.

1.2.5 Кодирование видеоинформации

Видеоинформация – наиболее сложный вид для хранения, обработки и воспроизведения. Впервые движущиеся изображения были сохранены на кинопленке в виде большого количества отдельных кадров изображения, заснятых через небольшие промежутки времени (24 кадра в секунду). Позднее на ту же пленку стала записываться и звуковая дорожка (в последующем несколько дорожек для многоканального звука). Далее появилось телевидение с аналоговой записью движущегося изображения на магнитные ленты (системы телевидения **PAL** и **SECAM** используют 25 кадров в секунду, система **NTSC** – 29,97 кадров в секунду). С появлением компьютеров широкое распространение получили цифровые методы записи и кодирования видеоинформации, которые постоянно совершенствуются. В настоящее время каждый может записать видео с использованием мобильных телефонов, цифровых фото- и видеокамер и выполнить монтаж видеофильма на персональных компьютерах, производительности которых достаточно для перекодирования видео высокого разрешения объемом в несколько гигабайт (но продолжительность кодирования может составлять несколько часов).

Компьютерные цифровые методы кодирования видео могут использовать частоту телевизионных стандартов **PAL/SECAM** или **NTSC**, т. к. видеозаписи многих цифровых форматов могут воспроизводиться как специальными компьютерными программами, так и бытовыми DVD-плеерами, а также путем подключения телевизора к компьютеру (для передачи видео и звука следует использовать порт HDMI).

Качество видеоизображения в цифровых методах постоянно улучшается. Широкое распространение цифрового видео было связано с появление вначале CD-дисков, затем DVD, далее Blu-Ray дисков, на которых, в основном, и распространялись кинофильмы, и емкостью которых ограничивались качественные возможности. В таблице 1.4 приведены характеристики некоторых видеоформатов.

Стандарты кодирования видео разрабатываются группой экспертов в области цифрового видео **MPEG** (Moving Picture Experts Group) Международной Организацией по Стандартизации (ISO). Первый стандарт **MPEG-1** был представлен в 1992 г., последние стандарты в этой области – **MPEG-7** и **MPEG-21**.

Алгоритмы кодирования видео очень сложны, их описания можно найти в специальной литературе или на сайте *http://www.mpeg.org*.

Таблица 1.4. Сравнение форматов записи видео на диски

Формат	Разрешение, PAL / NTSC	Стандарт кодирования		Совместимость с DVD-плеером
		видео	аудио	
VCD	352×288 352×240	MPEG-1	MPEG-1	всегда
SVCD	480×576 480×480	MPEG-2	MPEG-1	иногда
DVD	720×576 720×480	MPEG-2	MPEG-1, AC3	всегда
XVCD	720×576 720×480	MPEG-1 или MPEG-2	MPEG-1	иногда
DivX	640×480	MPEG-4	MP3, WMA	иногда
HDTV 720p	1280×720	MPEG-4 H.264	MP3, WMA, AC3 или др.	BD-плеер
HDTV 1080i	1920×1080 (i – чересстрочная развертка)	MPEG-4 H.264	MP3, WMA, AC3 или др.	BD-плеер
AVCHD 720p	1280×720 (p – прогрессивная развертка)	MPEG-4 v.10 (AVC/H.264)	PCM (7.1) или AC3 (5.1)	нет
AVCHD 1080i	1920×1080	MPEG-4 v.10 (AVC/H.264)	PCM (7.1) или AC3 (5.1)	нет

Все форматы сжатия семейства **MPEG** (MPEG-1, MPEG-2, MPEG-4, MPEG-7) используют высокую избыточность информации в изображениях, разделенных малым интервалом времени. Между двумя соседними кадрами обычно изменяется только малая часть сцены – например, происходит плавное смещение небольшого объекта на фоне фиксированного заднего плана. В этом случае полная информация о сцене сохраняется выборочно – только для **опорных** кадров. Для остальных кадров достаточно передавать разностную информацию: о положении объекта, направлении и величине его смещения, о новых элементах фона, открывающихся за объектом по мере его движения. Причем эти разности можно формировать не только по сравнению с предыдущими изображениями, но и с последующими (поскольку именно в них по мере движения объекта открывается ранее скрытая часть фона).

Алгоритмы MPEG сжимают только опорные кадры – I-кадры (Intra frame – внутренний кадр). В промежутки между ними включаются кадры, содержащие только изменения между двумя соседними I-кадрами – P-

кадры (Predicted frame – прогнозируемый кадр). Для того чтобы сократить потери информации между I-кадром и P-кадром, используются B-кадры (Bidirectional frame – двунаправленный кадр). В них содержится информация, которая берется из предшествующего и последующего кадров.

MPEG-4 использует технологию фрактального сжатия изображений. Фрактальное (контурно-основанное) сжатие подразумевает выделение из изображения контуров и текстур объектов. Контуры представляются в виде сплайнов (полиномиальных функций) и кодируются опорными точками. Текстуры могут быть представлены в качестве коэффициентов пространственного частотного преобразования (например, дискретного косинусного или вейвлет-преобразования).

Новые версии **MPEG-4** – **AVC/H.264** (Advanced Video Codec, называемый также **H.264**) – стандарт, предназначенный для значительного сжатия видеопотока при сохранении высокого качества и **AVCHD** (Advanced Video Codec High Definition – улучшенный видеокодек для видео высокого разрешения) – цифровой формат записи видеоданных в форматах 720p или 1080i и многоканального звука. Стандарт **AVCHD** был разработан совместно компаниями Sony и Panasonic в 2006 году. За основу был взят кодек AVC/H.264.

В октябре 1996 года группа **MPEG** приступила к разработке формата сжатия **MPEG-7**, призванным определить универсальные механизмы описания аудио и видео информации. Этот формат получил название «Мультимедиа-интерфейс для описания содержимого» (Multimedia Content Description Interface). В отличие от предыдущих форматов сжатия семейства **MPEG**, **MPEG-7** описывает информацию, представленную в любой форме (в том числе в аналоговой) и не зависит от среды передачи данных.

Формат сжатия **MPEG-7** использует развитую многоуровневую структуру описаний аудио и видео информации га основе языка этих описаний. Существуют различные типы информации, для которых разработаны схемы описания базовых структур: низкоуровневые аудио-визуальные характеристики, такие как цвет, текстура, движение, уровень звука и т.д.; высокоуровневые семантические объекты, события и абстрактные принципы; описание содержимого, навигации и доступа к аудио-визуальному материалу и т. д. Одной из отличительных особенностей **MPEG-7** является его способность к определению типа сжимаемой информации. Если это аудио или видео файл, то он сначала сжимается с помощью алгоритмов **MPEG-1, MPEG-2, MPEG-4**, а затем описывается при помощи **MPEG-7**.

Разработка формата **MPEG-21** – долговременный проект, который называется «Система мультимедийных средств» (Multimedia Framework).

Над разработкой этого формата эксперты начали работать в 2000 г. Задача разработки **MPEG-21** может быть сформулирована следующим образом: определение технологии, необходимой для поддержки пользователей при обмене, доступе, продаже и других манипуляциях цифровыми объектами. При этом предполагается обеспечить максимальную эффективность и прозрачность этих операций.

Форматы файлов Microsoft **AVI** и **MKV** – контейнеры, предназначенные для хранения видеоинформации, синхронизованной с аудиоинформацией. **AVI** может содержать в себе потоки 4 типов – Video, Audio, MIDI, Text. Причем видеопоток может быть только один, тогда как аудио – несколько.

Контейнер **MKV** (Matroska, матрёшка) разрабатывался с учётом современных тенденций и возможных тенденций будущего. Он универсален, так как построен на принципе EBML (то же самое, что и XML, но для двоичных данных). В **MKV** можно поместить любое количество аудио-видеорядов, меню, как на DVD, главы, субтитры, шрифты, постеры, тексты, комментарии, описания, фотоальбомы и проч. Ограничений практически нет. Максимальная совместимость со всеми возможными требованиями к видеоконтейнеру на данный момент и на ближайшее будущее. Используется в настоящее время для переноса информации DVD и Blu-Ray дисков в один файл *.mkv с сохранением меню, выбора языка воспроизведения, показа субтитров на выбранном языке, показа сцен-фрагментов основного фильма, рекламных роликов диска и пр.

1.2.6 Сжатие (архивация) различных видов информации

Дискретное двоичное представление информации обычно имеет некоторую избыточность. Часто в информации присутствуют последовательности одинаковых битов или их групп. Объём информации имеет большое значение не только для хранения, но также непосредственно влияет на скорость передачи информации по компьютерным сетям. Поэтому были разработаны специальные методы (алгоритмы) ***сжатия информации*** (data compression), с помощью которых можно существенно уменьшить ее объём. Существуют как универсальные алгоритмы, которые рассматривают информацию, как простую последовательность битов, так и специализированные, которые предназначены для сжатия информации определённого типа (изображений, текста, звука и видео).

Все алгоритмы сжатия оперируют входным потоком информации, минимальной единицей которой является, бит, а максимальной – несколько бит, байт или несколько байт.

Основными техническими характеристиками процессов сжатия и результатов их работы являются:

степень сжатия (compress rating) или отношение (ratio) объемов исходного и результирующего потоков;

скорость сжатия – время, затрачиваемое на сжатие некоторого объема информации входного потока, до получения из него эквивалентного выходного потока;

качество сжатия – величина, показывающая, насколько сильно упакован выходной поток, при помощи применения к нему повторного сжатия по этому же или иному алгоритму.

Все способы сжатия можно разделить на две категории: **обратимое** и **необратимое** сжатие.

Необратимое сжатие – такое преобразование входного потока информации, при котором выходной поток, основанный на определенном формате информации, представляет собой объект, достаточно похожий по внешним характеристикам на входной поток, однако отличается от него объемом.

Степень сходства входного и выходного потоков определяется степенью соответствия некоторых свойств объекта (до сжатия и после), представляемого данным потоком информации. Такие подходы и алгоритмы используются для сжатия информации растровых графических файлов, видео и звука. При таком подходе используется свойство структуры данного формата файла и возможность представить информацию приблизительно схожую по качеству для восприятия человеком. Поэтому, кроме степени или величины сжатия, в таких алгоритмах возникает понятие качества, т.к. исходная информация в процессе сжатия изменяется. Под качеством можно понимать степень соответствия исходной и результирующей информации, оцениваемое субъективно, исходя из формата информации. Для графических файлов такое соответствие определяется визуально, хотя имеются и соответствующие интеллектуальные алгоритмы и программы. Необратимое сжатие невозможно применять в областях, в которых необходимо иметь точное соответствие информационной структуры входного и выходного потоков. Данный подход реализован в популярных форматах представления фотоинформации – **JPEG, TIFF, GIF, PNG** и др., аудио информации – **MP3**, видео информации – **MPEG-4.**

Обратимое сжатие всегда приводит к снижению объема выходного потока информации без изменения его информативности, т.е. без потери информационной структуры.

Из выходного потока, при помощи восстанавливающего или декомпрессирующего алгоритма, можно получить входной, а процесс восстановления называется декомпрессией или распаковкой и только после процесса распаковки информация пригодна для использования в соответствии с их внутренним форматом.

Способы обратимого сжатия информации

1) Сжатие способом кодирования серий (RLE)

Наиболее известный простой подход и алгоритм сжатия информации обратимым путем – это кодирование серий последовательностей (Run Length Encoding – RLE).

Суть методов данного подхода состоит в замене цепочек или серий повторяющихся байтов или их последовательностей на один кодирующий байт и счетчик числа их повторений.

Например:
44 44 44 11 11 11 11 11 01 33 FF 22 22 – исходная последовательность
03 44 04 11 00 03 01 33 FF 02 22 – сжатая последовательность.

Первый байт во второй последовательности указывает, сколько раз нужно повторить следующий байт.

Если первый байт равен 00, то затем идет счетчик, показывающий, сколько за ним следует неповторяющихся байт информации (00 03).

Данные методы, как правило, достаточно эффективны для сжатия растровых графических изображений (BMP, PCX, TIF, GIF), т.к. последние содержат достаточно много длинных серий повторяющихся последовательностей байтов. Недостатком метода RLE является достаточно низкая степень сжатия.

2) Алгоритм Хаффмана

Сжимая файл по алгоритму Хаффмана, первое, что необходимо сделать – прочитать файл полностью и подсчитать сколько раз встречается каждый символ из расширенного набора **ASCII.**

Если учитывать все 256 символов, то не будет разницы в сжатии текстового и EXE файла.

После подсчета частоты вхождения каждого символа, необходимо сформировать бинарное дерево для кодирования с учетом частоты вхождения символов.

Пример сжатия по алгоритму Хаффмана приведен ниже.

Пусть файл имеет длину 100 байт и в нем присутствуют 6 различных символов. Подсчитаем вхождение каждого из символов в файл и получим следующую таблицу:

Символ	A	B	C	D	E	F
Число вхождений	10	20	30	5	25	10

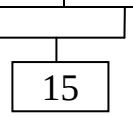
Отсортируем символы по частоте вхождения:

Символ	C	E	B	F	A	D
Число вхождений	30	25	20	10	10	5

Далее возьмем из последней таблицы 2 символа с наименьшей частотой. В нашем случае это D (5) и F (10) или A (10), можно взять любой из них, например A.

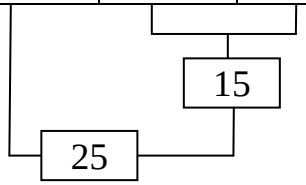
Сформируем из «узлов» D и A новый «узел», частота вхождения для которого будет равна сумме частот D и A:

Символ	C	E	B	F	A	D
Число вхождений	30	25	20	10	10	5



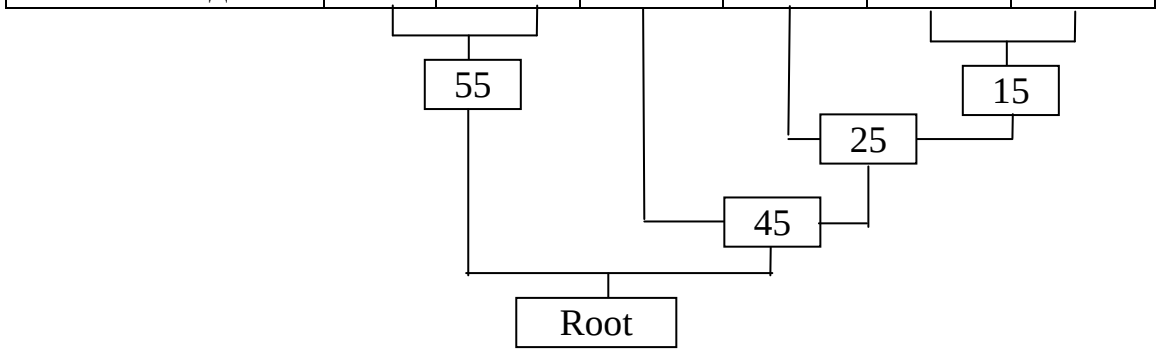
Номер в рамке – сумма частот символов A и D. Теперь мы снова ищем два символа с самыми низкими частотами вхождения. Исключая из просмотра D и A и рассматривая вместо них новый «узел» с суммарной частотой вхождения. Самая низкая частота теперь у F и нового «узла». Снова сделаем операцию слияния узлов:

Символ	C	E	B	F	A	D
Число вхождений	30	25	20	10	10	5



Просматриваем таблицу снова для следующих двух символов (B и E). Продолжаем этот режим пока все «дерево» не сформировано, т.е. пока все не сведется к одному узлу.

Символ	C	E	B	F	A	D
Число вхождений	30	25	20	10	10	5



Теперь, когда наше дерево создано, можно кодировать файл. Мы должны всегда начинать из корня (Root). Кодируя первый символ (лист дерева C с наибольшей частотой), прослеживаем вверх по дереву все повороты ветвей и если делаем левый поворот, то запоминаем бит = 0, и аналогично бит = 1 для правого поворота. Так для C, мы будем идти влево к 55 (и запомним 0), затем снова влево (0) к самому символу. Код Хаффмана для нашего символа C – 00. Для следующего символа (E) получается – лево, право, что выливается в последовательность 01. Выполнив эту процедуру для всех символов, получим:

C = 00 (2 бита)
E = 01 (2 бита)
B = 10 (2 бита)
F = 110 (3 бита)
A = 1101 (4 бита)
D = 1111 (4 бита)

При кодировании заменяем символы на новые коды, при этом те символы, которые встречаются наиболее часто, имеют самые короткие коды. Таблицу кодирования запоминаем в том же архивном файле для последующей разархивации.

3) Арифметическое кодирование

Совершенно иное решение предлагает так называемое арифметическое кодирование. Арифметическое кодирование является методом, позволяющим упаковывать символы входного алфавита без потерь при условии, что известно распределение частот этих символов и является наиболее оптимальным, т.к. достигается теоретическая граница степени сжатия.

Каждый символ после кодирования представляется как некоторое дробное число из интервала [0, 1) с весом, пропорциональным вероятности его появления.

4) Алгоритм Лемпеля-Зива-Велча (Lempel-Ziv-Welch – LZW)

Данный алгоритм отличают высокая скорость работы как при упаковке, так и при распаковке, достаточно скромные требования к памяти и простая аппаратная реализация.

Недостаток – низкая степень сжатия по сравнению со схемой двухступенчатого кодирования.

5) Двухступенчатое кодирование. Алгоритм Лемпеля-Зива

Гораздо большей степени сжатия можно добиться при выделении из входного потока повторяющихся цепочек – блоков, и кодирования ссылок на эти цепочки с построением хеш-таблиц от первого до n-го уровня с последующим кодированием Хаффмена или арифметическим кодированием.

Метод принадлежит Лемпелю и Зиву и обычно называется LZ-compression.

Перечень программ сжатия с кратким указанием алгоритмов их работы

PKPAK:
Метод Packed – алгоритм RLE.
Метод Crunched – алгоритм LZW.
Метод Squashed – двухпроходное статическое кодирование Хаффмена.
ZIP:
Метод Shrinked – модифицированный алгоритм LZW с частичной очисткой словаря и переменной длиной кода.
Метод Imploded – модифицированный алгоритм Лемпеля-Зива и статическое кодирование Хаффмена.
LHArc:
Алгоритм Лемпедя-Зива и динамическое кодирование Хаффмена.
LHA:
Алгоритм Лемпедя-Зива и статическое кодирование Хаффмена.
ARJ:
Алгоритм Лемпеля-Зива и оригинальный метод кодирования
WinRar:
Использование высокоэффективного оригинального алгоритма автора (А. П. Рошал).

1.3 Законодательство Российской Федерации о защите компьютерной информации

Ограничимся здесь цитированием нескольких наиболее важных статей законов, действующих в Российской Федерации, имеющих отношение к данной теме. Текст законов можно найти в правовых информационных системах «Консультант+», «Гарант» и др., а также в Интернете. Законодательство РФ достаточно часто модифицируется и дополняется. Например, вместо Закона об авторском праве в 2006 г. была принята часть четвертая Гражданского кодекса, последняя редакция которого по состоянию на сентябрь 2008 г. опубликована в редакции Федерального закона от 30.06.2008 N 104-ФЗ.

1.3.1 Уголовный Кодекс РФ о преступлениях в сфере компьютерной информации

(http://www.consultant.ru/online/base/?req=doc;base=LAW;n=93082)

Глава 28. Преступления в сфере компьютерной информации

Статья 272. Неправомерный доступ к компьютерной информации

1. Неправомерный доступ к охраняемой законом компьютерной информации, то есть информации на машинном носителе, в электронно-вычислительной машине (ЭВМ), системе ЭВМ или их сети, если это деяние повлекло уничтожение, блокирование, модификацию либо копирование информации, нарушение работы ЭВМ, системы ЭВМ или их сети, – наказывается штрафом в размере до двухсот тысяч рублей или в размере заработной платы или иного дохода осужденного за период до восемнадцати месяцев, либо исправительными работами на срок от шести месяцев до одного года, либо лишением свободы на срок до двух лет (в ред. Федерального закона от 08.12.03 № 162-ФЗ).

2. То же деяние, совершенное группой лиц по предварительному сговору или организованной группой либо лицом с использованием своего служебного положения, а равно имеющим доступ к ЭВМ, системе ЭВМ или их сети, – наказывается штрафом в размере от ста тысяч до трехсот тысяч рублей или в размере заработной платы или иного дохода осужденного за период от одного года до двух лет, либо исправительными работами на срок от одного года до двух лет, либо арестом на срок от трех до шести месяцев, либо лишением свободы на срок до пяти лет (в ред. Федерального закона от 08.12.03 № 162-ФЗ).

Статья 273. Создание, использование и распространение вредоносных программ для ЭВМ

1. Создание программ для ЭВМ или внесение изменений в существующие программы, заведомо приводящих к несанкционированному уничтожению, блокированию, модификации либо копированию информации, нарушению работы ЭВМ, системы ЭВМ или их сети, а равно использование либо распространение таких программ или машинных носителей с такими программами – наказываются лишением свободы на срок до трех лет со штрафом в размере до двухсот тысяч рублей или в размере заработной платы или иного дохода осужденного за период до восемнадцати месяцев (вред. Федерального закона от 08.12.03 № 162-ФЗ).

2. Те же деяния, повлекшие по неосторожности тяжкие последствия, – наказываются лишением свободы на срок от трех до семи лет.

Статья 274. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети

1. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети лицом, имеющим доступ к ЭВМ, системе ЭВМ или их сети, по влекшее уничтожение, блокирование или модификацию охраняемой законом информации ЭВМ, если это деяние причинило существенный вред, – наказывается лишением права занимать определенные должности или заниматься определенной деятельностью на срок до пяти лет, либо обязательными работами на срок от ста восьмидесяти до двухсот сорока часов, либо ограничением свободы на срок до двух лет.

2. То же деяние, повлекшее по неосторожности тяжкие последствия, – наказывается лишением свободы на срок до четырех лет.

1.3.2 Законодательство РФ о защите программ для ЭВМ, как объекта авторского права

Часть 4 Гражданского кодекса РФ определяет права на результаты интеллектуальной деятельности.

Статья 1225. Охраняемые результаты интеллектуальной деятельности и средства индивидуализации

1. Результатами интеллектуальной деятельности и приравненными к ним средствами индивидуализации юридических лиц, товаров, работ, услуг и предприятий, которым предоставляется правовая охрана (интеллектуальной собственностью), являются:

1) произведения науки, литературы и искусства;

2) программы для электронных вычислительных машин (программы для ЭВМ);

3) базы данных; …

Статья 1256. Действие исключительного права на произведения науки, литературы и искусства на территории Российской Федерации

1. Исключительное право на произведения науки, литературы и искусства распространяется:

1) на произведения, обнародованные на территории Российской Федерации или необнародованные, но находящиеся в какой-либо объективной форме на территории Российской Федерации, и признается за авторами (их правопреемниками) независимо от их гражданства;…

Статья 1261. Программы для ЭВМ

Авторские права на все виды программ для ЭВМ (в том числе на операционные системы и программные комплексы), которые могут быть выражены на любом языке и в любой форме, включая исходный текст и объектный код, охраняются так же, как авторские права на произведения литературы. Программой для ЭВМ является представленная в объективной форме совокупность данных и команд, предназначенных для функционирования ЭВМ и других компьютерных устройств в целях получения определенного результата, включая подготовительные материалы, полученные в ходе разработки программы для ЭВМ, и порождаемые ею аудиовизуальные отображения.

Кроме того, в части 4 Гражданского кодекса присутствуют специальные статьи, посвященные программам для ЭВМ:

Статья 1262. Государственная регистрация программ для ЭВМ и баз данных

Статья 1280. Свободное воспроизведение программ для ЭВМ и баз данных. Декомпилирование программ для ЭВМ

Статья 1296. Программы для ЭВМ и базы данных, созданные по заказу

Статья 1297. Программы для ЭВМ и базы данных, созданные при выполнении работ по договору.

Многие положения других статей относятся ко всем объектам авторского права, которыми являются и программы для ЭВМ и базы данных.

Ответственность за нарушение авторских прав изложена в статье 146 Уголовного кодекса РФ.

1.3.3 Федеральный закон РФ «Об информации, информационных технологиях и о защите информации»

Статья 1. Сфера действия настоящего Федерального закона

1. Настоящий Федеральный закон регулирует отношения, возникающие при:

1) осуществлении права на поиск, получение, передачу, производство и распространение информации;

2) применении информационных технологий;

3) обеспечении защиты информации.

2. Положения настоящего Федерального закона не распространяются на отношения, возникающие при правовой охране результатов интеллектуальной деятельности и приравненных к ним средств индивидуализации.

Дальнейшие статьи содержат многочисленные положения правового регулирования в сфере информации, информационных технологий и защиты информации. Большое внимание уделяется вопросам ограничения доступа к информации, создания государственных информационных систем, защиты информации.

1.4 Требования к организации рабочих мест пользователей ПК

Главным документом, определяющим данные требования, является СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

Документ содержит разделы:

I. Общие положения и область применения

II. Требования к ПЭВМ

III. Требования к помещениям для работы с ПЭВМ

IV. Требования к микроклимату, содержанию аэроионов и вредных химических веществ в воздухе на рабочих местах, оборудованных ПЭВМ

V. Требования к уровням шума и вибрации на рабочих местах, оборудованных ПЭВМ

VI. Требования к освещению на рабочих местах, оборудованных ПЭВМ

VII. Требования к уровням электромагнитных полей на рабочих местах, оборудованных ПЭВМ

VIII. Требования к визуальным параметрам ВДТ, контролируемым на рабочих местах

IX. Общие требования к организации рабочих мест пользователей ПЭВМ

X. Требования к организации и оборудованию рабочих мест с ПЭВМ для взрослых пользователей

XI. Требования к организации и оборудованию рабочих мест с ПЭВМ для обучающихся в общеобразовательных учреждениях и учреждениях начального и высшего профессионального образования

XII. Требования к оборудованию и организации помещений с ПЭВМ для детей дошкольного возраста

XIII. Требования к организации медицинского обслуживания пользователей ПЭВМ

XIV. Требования к проведению государственного санитарно-эпидемиологического надзора и производственного контроля.

Документ содержит также 14 приложений.

Некоторые самые существенные требования по организации рабочих мест пользователей ПЭВМ (так здесь называют ПК; ВДТ – видеодисплейный терминал), о которых не должен забывать простой пользователь, приведены в следующих разделах документа:

3.4. Площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе электронно-лучевой трубки (ЭЛТ) должна составлять не менее 6 $м^2$, в помещениях культурно-развлекательных учреждений и с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) – 4,5 $м^2$.

При использовании ПЭВМ с ВДТ на базе ЭЛТ (без вспомогательных устройств – принтер, сканер и др.), отвечающих требованиям международных стандартов безопасности компьютеров, с продолжительностью работы менее 4-х часов в день допускается минимальная площадь 4,5 $м^2$ на одно рабочее место пользователя (взрослого и учащегося высшего профессионального образования).

4.4. В помещениях, оборудованных ПЭВМ, проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ПЭВМ.

6.1. Рабочие столы следует размещать таким образом, чтобы видеодисплейные терминалы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева.

9.1. При размещении рабочих мест с ПЭВМ расстояние между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора) должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов – не менее 1,2 м.

9.4. Экран видеомонитора должен находиться от глаз пользователя на расстоянии 600 – 700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

9.6. Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ПЭВМ, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип

рабочего стула (кресла) следует выбирать с учетом роста пользователя, характера и продолжительности работы с ПЭВМ.

Рабочий стул (кресло) должен быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию.

9.7. Поверхность сиденья, спинки и других элементов стула (кресла) должна быть полумягкой, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием, обеспечивающим легкую очистку от загрязнений.

Полный текст документа можно найти в Интернете по адресу *http://lib.herzen.spb.ru/postuplenia/gigienich_trebov.pdf*.

Контрольные вопросы к главе 1

1) В чем заключается задача изучения курса «Информатика» в вузе?
2) Что такое «информация», какие виды ее Вы знаете?
3) Как измеряется объем компьютерной информации?
4) В какой системе счисления представлена информация при ее обработке и хранении на компьютере?
5) Какие кодировки используются для работы с кириллицей? Что такое UTF-8?
6) В чем разница при кодировании целых и вещественных чисел?
7) Что такое RGB и CVYK?
8) Что такое частота дискредитации и глубина кодирования звука? В каких единицах измеряется битрейт?
9) Какие форматы и стандарты кодирования видеоинформации Вы знаете?
10) В чем разница необратимого и обратимого сжатия информации, где они используются?
11) Как наказывается неправомерный доступ к компьютерной информации и распространение вредоносных программ по законодательству РФ?
12) Какие основные положения по организации рабочих мест пользователей ПК Вы должны выполнять в соответствии с СанПиН 2.2.2/2.4.1340-03?

Глава 2. Аппаратное обеспечение персональных компьютеров

Современный компьютер – это система, построенная на базе электронных микросхем, и предназначенная для хранения, обработки и передачи любых видов информации.

Простые пользователи обычно имеют дело с персональными компьютерами (**ПК**), поэтому в дальнейшем речь пойдет именно о них.

Наиболее известны и распространены персональные компьютеры на базе процессоров фирмы Intel и их аналогов (ранее их называли IBM-совместимые ПК) и ПК фирмы Apple – Macintosh**.**

В последние годы все более широкое распространение получает ноутбук – переносной персональный компьютер, с размерами, например, 33 × 24 × 3,5 см, масса его 2 – 2,5 кг (см. рисунок 2.1).

Рисунок 2.1. Настольный ПК и ноутбук

Историю развития средств вычислительной техники можно найти во многих учебниках по информатике или в Интернете, например на сайте Свободной энциклопедии – Википедии (см. *http://ru.wikipedia.org/*wiki/*Компьютер*.

Аппаратное обеспечение – все те компоненты, из которых состоит компьютер, а также оборудование для организации локальных и глобальных сетей – так называемое компьютерное «железо» (hardware).

Прежде всего, это процессор, материнская плата и ее главные микросхемы – чипсет, определяющий всю архитектуру компьютера, возмож-

ные типы основной оперативной памяти, видеокарт, дисковых устройств, мониторов, принтеров и других периферийных устройств.

Знание современных технических средств, а также истории их развития, необходимо каждому пользователю ПК. Обновление всех компонентов ПК идет стремительными темпами. Не так давно появились многоядерные процессоры для настольных компьютеров и ноутбуков; новые шины для высокоскоростной работы с памятью, видеоподсистемой, жесткими дисками; интегрированные контроллеры для работы с гигабитными проводными сетями и с беспроводными сетями. Жидкокристаллические мониторы вытеснили с рынка мониторы на электронно-лучевых трубках. Продвигается новая концепция домашнего компьютера – медиацентра, который не только работает с DVD-дисками и имеет качественную многоканальную аудиосистему, но и работает с радио- и телевизионными трансляциями Интернета, а также с FM и TV-тюнерами, которыми комплектуется ПК, с программируемым управлением этими средствами.

Умение подобрать необходимую конфигурацию ПК или осознанно выбрать ноутбук позволит избежать покупки морально устаревшего оборудования, цена которого может ненамного отличаться от новой, более совершенной техники. Так, в компьютерных магазинах часто можно встретить типовые конфигурации ПК с морально устаревшими процессорами и с жесткими дисками не самых оптимальных объемов (часто новый диск с емкостью в 2 раза большей, чем более старый, бывает дороже всего лишь на 10-20%).

Чтобы Ваш ПК всегда быстро работал и оперативно выполнял все необходимые функции, необходимо самому пользователю разбираться в основном аппаратном обеспечении и соответствующем ему программном обеспечении.

2.1 Процессор

Процессор – специальная микросхема, которая выполняет операции по обработке информации в компьютере.

На небольшой кремниевой пластине размещены сотни миллионов транзисторов-переключателей и каналов передачи данных. Кроме центрального процессора (**CPU**) в современных компьютерах значительную роль играет процессор видеокарты, который занимается обработкой видеоинформации. В 2010 году при переходе на 32 нм технологию корпора-

ция Intel разместила в одном корпусе центрального процессора также вторую микросхему – видеоядро HD Graphics (GPU).

От процессора в значительной степени зависит скорость работы **ПК** (лимитирующим фактором также может быть объем оперативной памяти). Процессор имеет сложную архитектуру, свою высокоскоростную буферную память (кэш), использует специальные технологии обработки информации.

Простейшая принципиальная схема микропроцессора Intel представлена на рисунке 2.2 (по данным фирмы Intel).

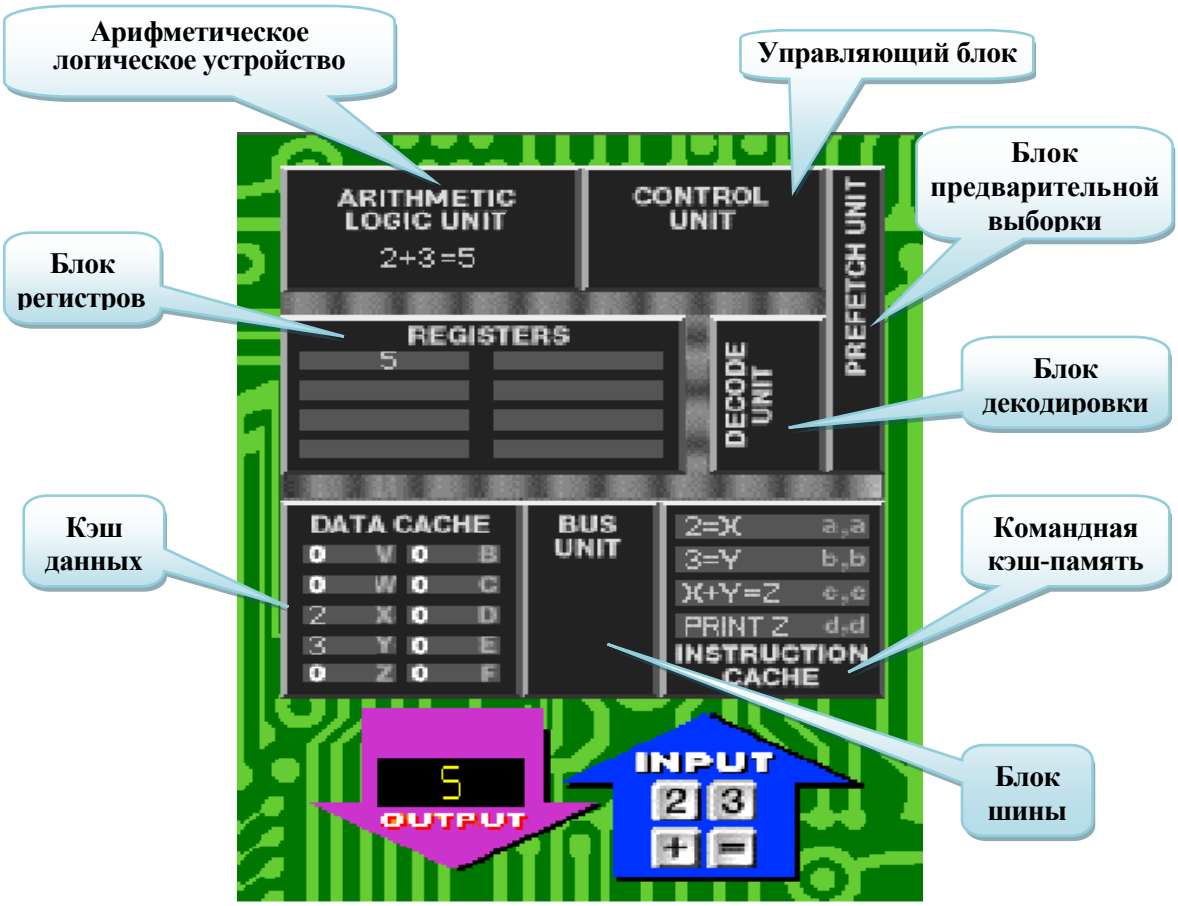


Рисунок 2.2. Принципиальная схема процессора по данным фирмы Intel

Принцип работы центрального процессора можно представить следующим образом. Информация для обработки под управлением блока предварительной выборки поступает из системной памяти через блок шины в кэш данных процессора, команды обработки информации – в командную кэш-память. Блок декодировки раскодирует команды, преобразуя их в двоичный код, который пересылается в управляющий блок и в кэш данных, давая им указание о том, как с полученной командой посту-

пать дальше. Арифметическое логическое устройство выполняет готовые к исполнению команды и заносит результаты в блок регистров. Далее содержимое регистров передается в системную память или на внешние устройства. Более подробное описание выполнения процессором простой операции сложения 2+3 можно найти по адресу *http://www.intel.com/plt/cd/corporate/emea/rus/museum/mpuworks.swf*, здесь же в глоссарии приведено краткое описание функций всех устройств, показанных на рисунке 2.2.

Скорость работы процессора зависит в настоящее время, прежде всего от типа и архитектуры процессора, а также от его тактовой частоты и объема кэш-памяти. Процессоры Intel Pentium и Core используют технологию конвейерной обработки данных, в результате чего за один такт выполняется несколько машинных операций. Многоядерные процессоры позволяют увеличить производительность **ПК** за счет одновременного выполнения нескольких программ пользователя на разных ядрах или выполнения одной программы на нескольких ядрах, если она предусматривает параллельную многопроцессорную обработку данных. Скорость обработки информации процессором может также лимитироваться скоростью поступления этой информации из оперативной памяти.

Процессоры выпускаются различными фирмами, для различных типов компьютеров и для другой электроники. Так, фирма Intel (www.intel.com) выпускает процессоры не только для настольных ПК, но и для ноутбуков, серверов, коммуникаторов и другого оборудования (см. таблицу 2.1). Другой наиболее известный производитель – фирма AMD (www.amd.com).

Таблица 2.1 Процессоры фирмы Intel различного назначения

Для настольных ПК	Для мобильных ПК	Для серверов	Прикладные	Сетевые	Ввода/вывода
Core™ i7	Core™ i3 - i7	Xeon®	PXA270	IXP465	IOP332
Core™ i5	Core™2 Duo	Itanium® 2	PXA255	IXP460	IOP331
Core™ i3	Core™ Solo			IXP455	IOP321
Core™2 Duo	Atom™			IXP425	IOP303

В 1990 году прогнозировалось – тактовая частота процессора Intel возрастёт к 2000 году до 900 Мгц, количество транзисторов в нём – до 40 млн. штук, к 2005 году – 10 Ггц, 1 млрд. транзисторов. Однако в 2005 г. стало ясно, что прогресс пошел другим путем – развитием многоядерных процессоров без увеличения их рабочей частоты и количества транзисто-

ров на чипе (но с увеличением частоты системной шины FSB до 1333 МГц и с поддержкой 64-разрядных вычислений и памяти).

Графики реального увеличения тактовой частоты процессора и количества транзисторов в нем с момента их появления по сегодняшний день приведены на рисунке 2.3.

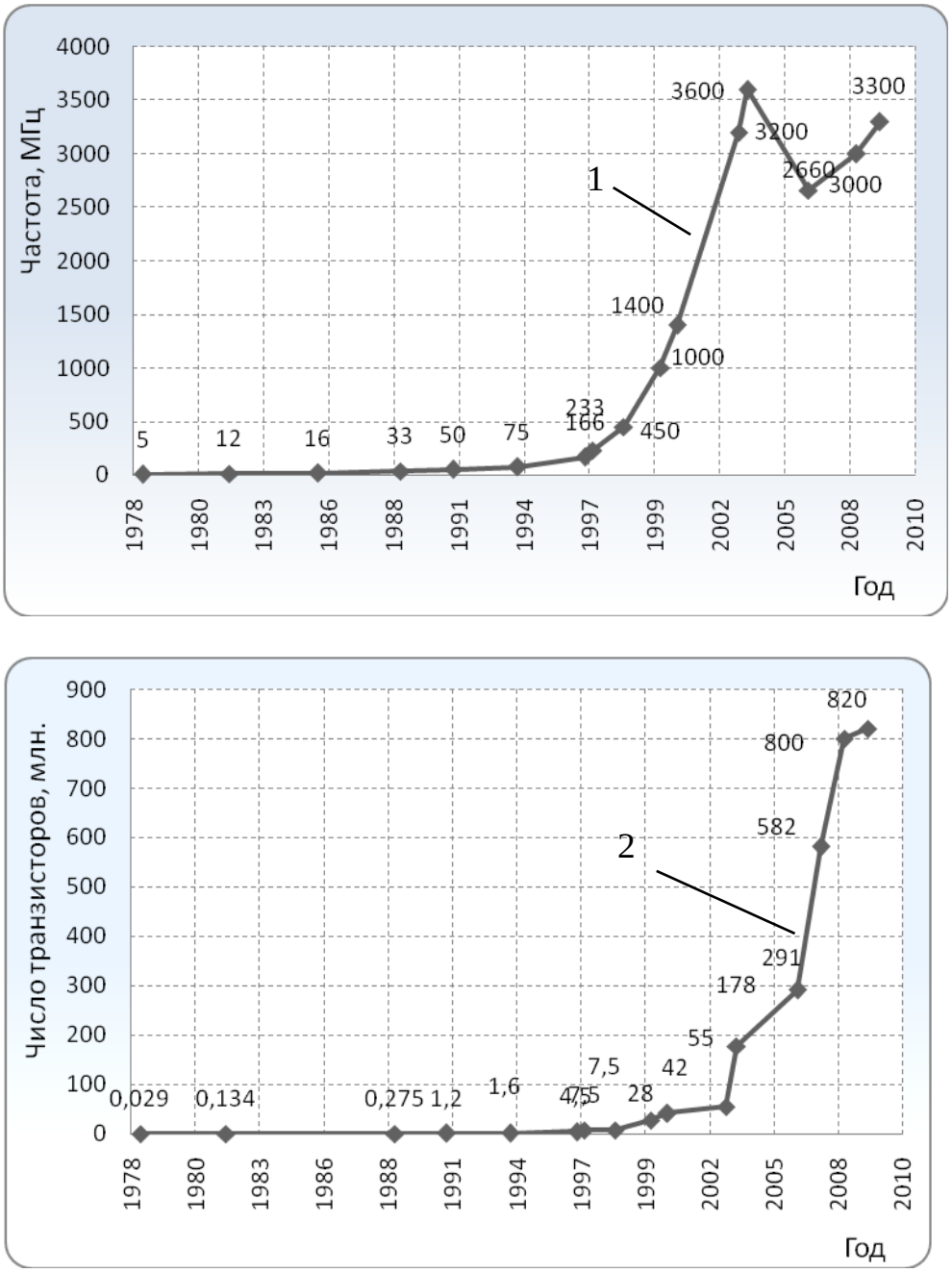


Рисунок 2.3. Графики изменения тактовой частоты (1) количества транзисторов (2) в процессорах фирмы Intel

Некоторое снижение тактовой частоты и уменьшение числа транзи-

сторов в 2005 – 2006 годах было вызвано появлением двухъядерных процессоров *Core Duo* и *Core 2 Duo*, а позднее четырехъядерного *Core Quad*, суммарная производительность двух или четырех ядер которых выше, чем одноядерных процессоров с более высокой тактовой частотой.

Историю развития ПК можно проследить по основным этапам развития процессоров фирмы Intel.

Полную таблицу выпущенных этой фирмой процессоров для настольных ПК, ноутбуков и серверов можно посмотреть на сайте фирмы Intel (*http://www.intel.com/corporate/europe/emea/rus/country/museum/history/*), краткая справка по истории развития процессоров для настольных ПК приведена в таблице 2.2.

Таблица 2.2. История развития процессоров Intel (для настольных ПК)

Начало выпуска	Процессор	Тактовая частота	Технология; число транзисторов
2010 г.	Intel® Core™ i7-970 (6 ядер, 12 потоков, HT)	3,2 - 3,46 ГГц, Turbo Boost	32-нм
2010 г.	Intel® Core™ i3-560, i3-530 (2 ядра, 4 потока, HT)	3,33, 2,93 ГГц	32-нм
2010 г.	Intel® Core™ i5-680, ™ i5-670, i5-661, i5-660, i5-650 (2 ядра, 4 потока, HT)	от 3,46 ГГц до 3,73 ГГц, технология Turbo Boost	32-нм
2010 г.	Pentium® G6950 (2 ядра)	2,8 ГГц	32-нм
2009 г.	Intel® Core™ i5-750 (4 ядра)	2,66 ГГц	45-нм
2008 г.	Intel® Core™ i7-960 – i7-860 (4 ядра, 8 потоков, технология HT)	от 3,2 до 3,46 ГГц технология Turbo Boost	45-нм 731 млн.
2008 г.	Intel® Atom™ (для субноутбуков)	2,0 - 0,8 ГГц	45-нм 47 млн.
2007 г.	Intel® Pentium® Dual-Core	2,8 - 1,60 ГГц	45 или 65-нм
2007 г.	Intel® Core™ 2 Quad (4 ядра) Q9650 … Q6600	3,00 - 2,40 ГГц	45 или 65-нм 820 млн.
2007 г.	Intel® Core™ 2 Extreme QX9775, QX6700 (4 ядра)	3.20 - 2.66 ГГц	45 или 65-нм 410 млн.

Продолжение таблицы 2.2

Начало выпуска	Процессор	Тактовая частота	Технология; число транзисторов
2006 г.	Intel®Core™2 Duo (2 ядра) E6700, E6750 … E4300	2,66 – 1.80 ГГц	65-нм; 291 млн.,167 млн.
2006 г.	Core™ Duo (2 ядра)	1,66 - 2,33 ГГц	65-нм 152 млн.
2005 г.	Intel® Pentium® Extreme Edition № 840 (2 ядра, технология HT)	3,20 ГГц	90-нм
2005 г.	Intel® Pentium® D (2 ядра) №№ 960 … 805	3,60 - 2,66 Гц	90-нм; 376 млн.
2004 г.	Intel® Celeron® D №№ 351, 346, ... 320, 315	3,20 - 2,53ГГц	90-нм 188 млн.
2004 г.	Intel® Pentium® 4 Extreme Edition	3.73, 3,40 ГГц	90 нм, 0,13 мкм 169 млн.
2003 г.	Intel® Pentium® 4 с технологией Intel® HT	3,60 - 2,40 ГГц	0,09 мкм, 0,13 мкм; 178 млн.
2000 г.	Intel® Pentium® 4	1,40 ГГц	0,18 мкм; 42 млн.
1999 г.	Intel® Pentium® III	1 ГГц - 450 МГц	0,18 мкм; 28 млн. 0,25 мкм; 9.5 млн.
1998 г	Intel® Celeron®	2,80 ГГц, 266 МГц	0,13 мкм; 0,25 мкм; 7,5 млн.
1997г.	Intel® Pentium® II	450 – 350 МГц 300 – 233 МГц	0,25 мкм; 0,35 мкм; 7,5 млн.
1995 г.	Intel® Pentium® Pro	200 -166 МГц, 50 МГц	0,35 мкм; 5,5 млн. 0,6 мкм
1997 г.	Intel® Pentium® с технологией MMX™	233 – 166 МГц	0,35 мкм; 4,5 млн.
1993 г.	Intel® Pentium®	66 – 60 МГц	0,8 мкм; 3,1 млн.
1992 г.	Intel486™ SL	33 – 20 МГц	0,8 мкм; 1,4 млн.
1994 г.	IntelDX4™	100, 75 МГц	0,6 мкм; 1,6 млн.

Продолжение таблицы 2.2

Начало выпуска	Процессор	Тактовая частота	Технология; число транзисторов
1989 г.	Intel486™ DX	50, 33, 25 МГц	1 – 0,8 мкм; 1,2 млн.
1992 г.	Intel386™ SX	33, 25, 20, 16 МГц	1,5 – 1 мкм; 275 тыс.
1985 г.	Intel386™ DX	33, 25, 20, 16 МГц	1,5 – 1 мкм; 275 тыс.
1982 г.	80286	12, 10, 6 МГц	1,5 мкм; 134 тыс.
1978 г.	8086	10, 8, 5 МГц	3 мкм; 29 тыс.

Процессор Core является переработанной версией процессора Pentium M, который был основой платформы Intel Centrino для ноутбуков.

Процессор Pentium M представлял собой продолжение Pentium III, адаптированный под актуальные потребности, с широкой шиной, поддержкой новых наборов инструкций и направленностью на минимальное энергопотребление.

Двухъядерные процессоры Intel® Core® Duo, основанные на принципиально иной по сравнению с Pentium 4 архитектуре, появились в начале 2006 года, во второй половине года появились процессоры Intel® Core™2 Duo, в конце 2008 г. – Core™ i7, в 2009 г. – Core™ i5 (см. рисунок 2.4).

Рисунок 2.4. Процессоры Intel®: Core™2 Duo, Core™ i7 и Core™ i5

Процессор Pentium 4, преобладая на рынке почти 6 лет (с 2000 по 2005 год), получил отставку. На смену пришли многоядерные процессоры: двухъядерный Core 2 Duo, двухъядерные четырехпоточные Core i3 и Core

i5 (32-нм), четырехъядерный восьмипоточный Core i7. В дальнейшем планируется выпуск процессоров для обычных ПК с еще большим количеством ядер.

Эволюционный ряд процессоров Pentium и Core показан на рисунке 2.5.

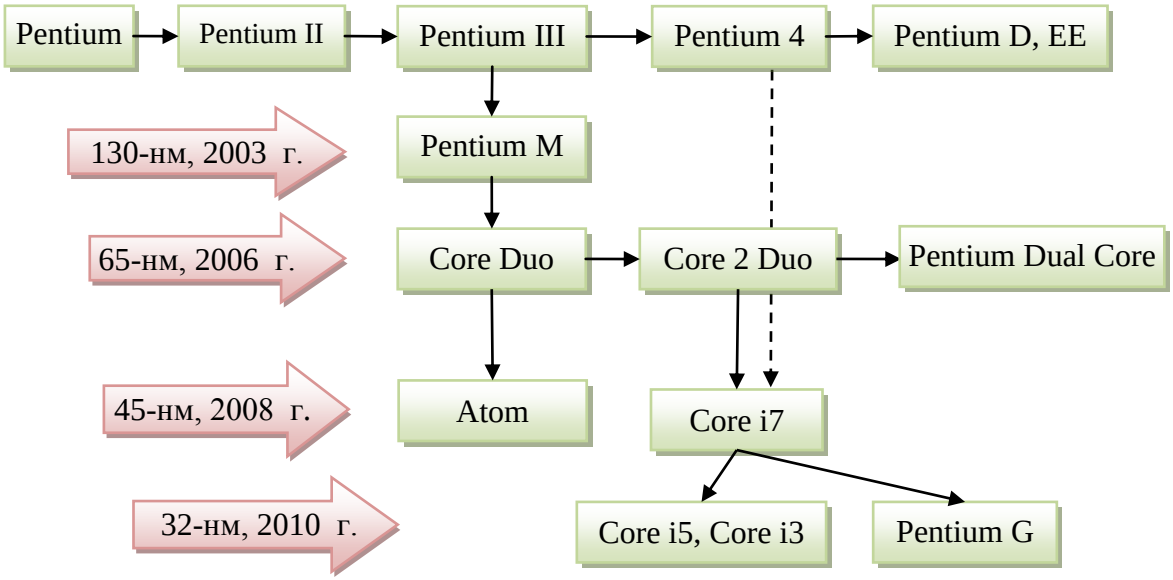


Рисунок 2.5. Эволюционный ряд процессоров Pentium и Core фирмы Intel

В 2008 году в производстве процессоров произошел переход с технологического процесса 65 нм на 45. Особенность новых процессоров – более 410 млн. транзисторов для двухъядерного и более 820 млн. для четырёхъядерного процессоров Intel при уменьшившейся до 110 мм² площади кристалла (у 65-нм – 143 мм²), поддержка порядка 50 новых инструкций Intel SSE4, нацеленных на повышение возможностей и производительности при работе с мультимедийным контентом.

В конце 2008 г. Intel представила четырехъядерные восьмипоточные процессоры **Core i7** на микроархитектуре Nehalem со встроенным контроллером памяти (выпускается в корпусе LGA1366). Пропускная способность шины памяти компьютеров с процессорами Intel **Core i7** выросла более чем в два раза по сравнению с процессорами Intel серии Extreme благодаря новому интерфейсу **Intel QuickPath Interconnect** (Intel® QPI), заменившему привычную системную шину **FSB** (Front-Side Bus, см. далее в разделе 2.2).

В 2010 году произошел переход на 32-нм технологию для процессоров **Core i7** (для ноутбуков), **Core i5, Core i3** и **Pentium G6950** архитектуры Clarkdale.

В двухъядерных и четырехъядерных процессорах Intel используются все новейшие технологии Intel, обеспечивающие экстремальную производительность, например, для **Core i5**:

- Технология **Intel® Turbo Boost** – максимально повышает производительность ресурсоемких приложений, динамически увеличивая производительность в соответствии с нагрузкой.
- Технология **Intel® Virtualization** (Intel® VT) – управление эффективной и масштабируемой виртуальной инфраструктурой, с сохранением оптимальной производительности при низких затратах.
- Технология **Intel® Enhanced Speedstep** – управления энергопотреблением процессора в зависимости от нагрузки.
- Функция **Execute Disable Bit** позволяет процессору выделять области памяти, где допускается выполнение кода приложений, и где оно не допускается. Когда вредоносная программа-червь пытается установить свой код в буфер памяти, процессор отключает выполнение кода.
- Архитектура **Intel 64** – обеспечивает поддержку 64-разрядных вычислений, предоставляет процессору доступ к большему объёму памяти
- Технология **Intel® Hyper-Threading** – обеспечивает обработку 2 потоков команд каждым физическим ядром, повышая общую пропускную способность процессора до 4 потоков. Она позволяет каждому ядру процессора одновременно выполнять две задачи.
- Технология **Intel® Smart Cache** – использование общей кэш-памяти процессора 2 уровня, которая динамически распределяется между ядрами в зависимости от нагрузки..
- **Интегрированный контроллер памяти** поддерживает 2 канала памяти DDR3 1333 МГц.
- Графическая система **Intel® Graphics Media Accelerator HD,** расположенная в одном корпусе с процессором, обеспечивает высокое качество воспроизведения видео HD и графики в трехмерных играх без необходимости использования дополнительных видеокарт или декодеров. Графическое ядро HD Graphics оснащено 14 исполнительными блоками (14 EU, Execution Units) с потоковыми процессорами Intel 3-го поколения.

Блок-схема процессоров семейства Clarkdale (Core i5, i3, Pentium G6950) показана на рисунке 2.6.

Начиная с **Core**, в модельном ряду **Intel** произошло изменение способа нумерации процессоров. Если раньше использовались трехцифровые обозначения, то теперь модельный номер представляет собой комбинацию из пяти символов – одной буквы и четырёх цифр. Например: **Core 2 Duo**

E6700, Core 2 Quad Q6600 (у экстремальных процессоров добавляется еще одна буква X: Core2 Extreme quad-core QX6800).

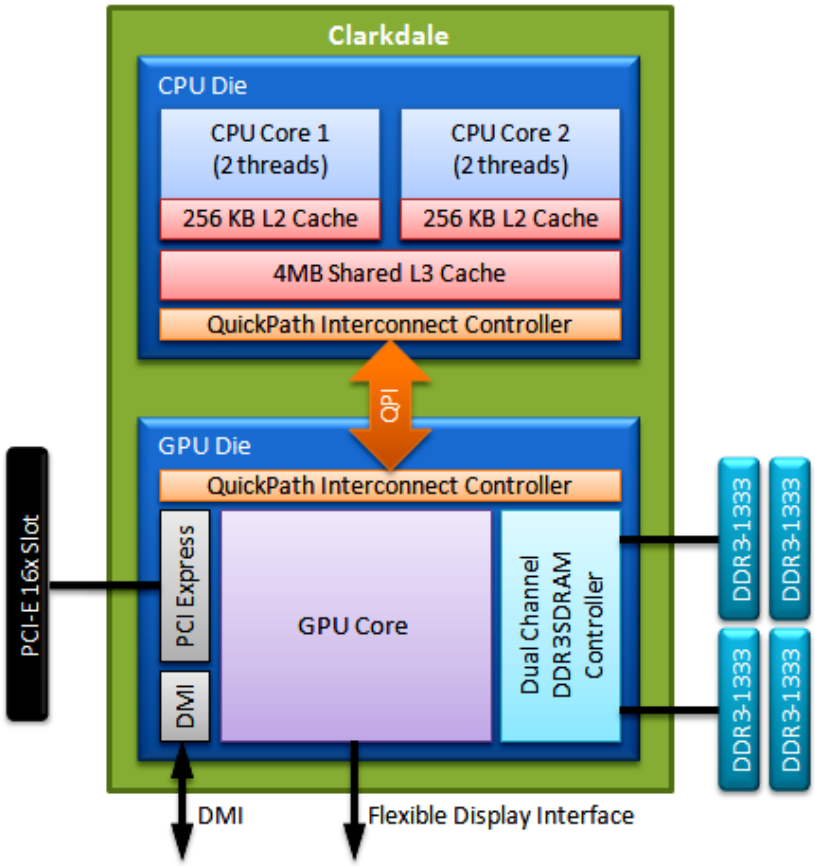


Рисунок 2.6. Блок-схема процессоров семейства Clarkdale

Первая буква символизирует энергопотребление/тепловыделение и, таким образом, тип процессора. (например, **E –** более 50 Вт, процессор для настольных ПК). Четыре цифры после буквы показывают производительность в пределах одной линейки.

У процессоров **Core i7, Core i5** и **Core i3** используется трехцифровое обозначение. В рамках каждого класса или семейства процессоров больший номер соответствует большему количеству различных функций, в том числе: объем кэш-памяти, тактовая частота, частота системной шины, поддержка новых технологий, новых команд и пр.

Рейтинг процессоров Intel можно посмотреть на сайте *http://www.intel.com/ru_RU/consumer/rating.htm*.

В таблице 2.3 показаны некоторые процессоры Intel и их цены (январь 2010 г.), в Приложении 2 приведены более подробные сведения.

Таблица 2.3. Некоторые процессоры Intel и их цены (http://price.ru)

Модель (ядер/потоков)		Тактовая частота, ГГц	Кэш, Мб	Цена, руб. (сентябрь 2010 г.)
Intel® Core™ i3-530 (2/4)	32 нм	2,93	4	3686
Intel® Core™ i5-680 (2/4)	32 нм	3,20	4	5973
Intel® Core™ i5-750 (4/4)	45 нм	3,20	8	6410
Intel® Core™ i7-960 (4/8)	45 нм	2,80	8	9900
Intel® Core™ i7-970 (6/12)	32 нм	3,20	12	32670
Intel® Pentium® G6950 (2/2)	32 нм	2,80	3	2710
Core™2 Quad Q8400 (4 ядра)	45 нм	2,66	4	5325
Core™2 Duo E8300 (2 ядра)	45 нм	2,83	6	4999

Фирма **AMD** также выпускает многоядерные процессоры для настольных ПК – AMD Athlon™ и Phenom™ (см. таблицу 2.4).

Таблица 2.4. Некоторые модели процессоров AMD

Модель	Тактовая частота, ГГц	Кэш L2, всего, Мб	Цена, руб. (ноябрь 2009 г.)
Phenom II X4 955 (4 ядра)	3,2	2	6500
Phenom II X4 620 (4 ядра)	2,6	2	4070
Phenom II X3 720 (3 ядра)	2,8	7,5	3820
Phenom II X2 550 (2 ядра)	3.1	7	3200
Athlon II X2 250 (2 ядра)	3.0	2	2520

Особенность процессоров AMD Athlon и Phenom – контроллер памяти интегрирован в процессор (у процессоров Intel до Core i7 он находится в чипсете – в контроллере-концентраторе памяти MCH – см. далее в разделе 2.2). Каждое ядро имеет свой кэш второго уровня и работает со своей оперативной памятью, но при необходимости может обращаться и к памяти другого ядра (технология HyperTransport™).

Сравнению процессоров AMD и Core посвящено большое количество статей на сайтах *www.thg.ru, www.ixbt.com* и пр. На конец 2009 г. по производительности процессоры Core i5 и Core i7 превосходили Phenom II X4 (см., например, *http://www.thg.ru/cpu/cpu_gamer_november_2009*).

2.2 Чипсет

Чипсет (chipset) – набор микросхем материнской платы для обеспечения работы процессора с памятью и внешними устройствами.

Раньше компьютер имел до 2-х сотен микросхем на материнской плате. Современные компьютеры содержат две основные большие микросхемы чипсета, либо одну микросхему :

1. Трехчиповая компоновка (третий чип – процессор):
   - ♦ северный мост (North Bridge) – один из двух вариантов:
     - – ***контроллер-концентратор памяти*** (*MCH*), который обеспечивает работу процессора с памятью и с видеоподсистемой или
     - – ***контроллер ввода-вывода*** (*IOH*) – для процессоров с интегрированным контроллером памяти;
   - ♦ южный мост (South Bridge) – ***контроллер-концентратор ввода-вывода*** (*ICH*), обеспечивающий работу с внешними устройствами.
2. Двухчиповая компоновка для некоторых процессоров Core i7 и Core i5 – кроме процессора используется только одна большая микросхема чипсета (PCH – Platform Controller Hub), обеспечивающая работу с внешними устройствами.

Разработкой чипсетов для настольных ПК занимаются 5 компаний: Intel, AMD, NVIDIA, VIA и SIS.

Фирма Intel делит чипсеты для настольных ПК на группы по назначению:

- производительные – X58, X48, X38…
- распространенные – Q45, Q43, P55, P45, G45…

Выбор типа чипсета зависит от процессора, с которым он работает, и определяет вид других устройств ПК (оперативной памяти, видеокарты, винчестера и пр.). Для процессоров Intel можно найти список рекомендованных материнских плат, построенных на определенных чипсетах.

Например, для процессоров Core 2 Duo E8600 в список совместимых входят чипсеты Intel® G31, G33, G35, G45, Q35, Q43, Q45, P55, X38, X48 (http://processormatch.intel.com/CompDB/SearchResult.aspx?ProcNbr=E8600) и материнские платы на их основе.

Помимо поддержки двух- и четырехъядерных процессоров Intel Core, большинство чипсетов 3-ей, 4-ой и 5-ой серий поддерживают тип памяти ***DDR3*** (наряду с традиционной ***DDR2***), новое поколение интерфейса ***PCI Express 2.0*** с удвоенной пропускной способностью графики, а так-

же работают с новой технологией ***Intel Turbo Memory*** для ускорения загрузки приложений. Чипсеты с буквой G и Q в обозначении имеют интегрированную графику с полноценной аппаратной поддержкой DirectX10, все чипсеты этих серий имеют интегрированную аудиоподсистему High Definition Audio.

Принципиальная схема чипсета Intel® G45 Express приведена на рисунке 2.7 (информация с сайта фирмы Intel).

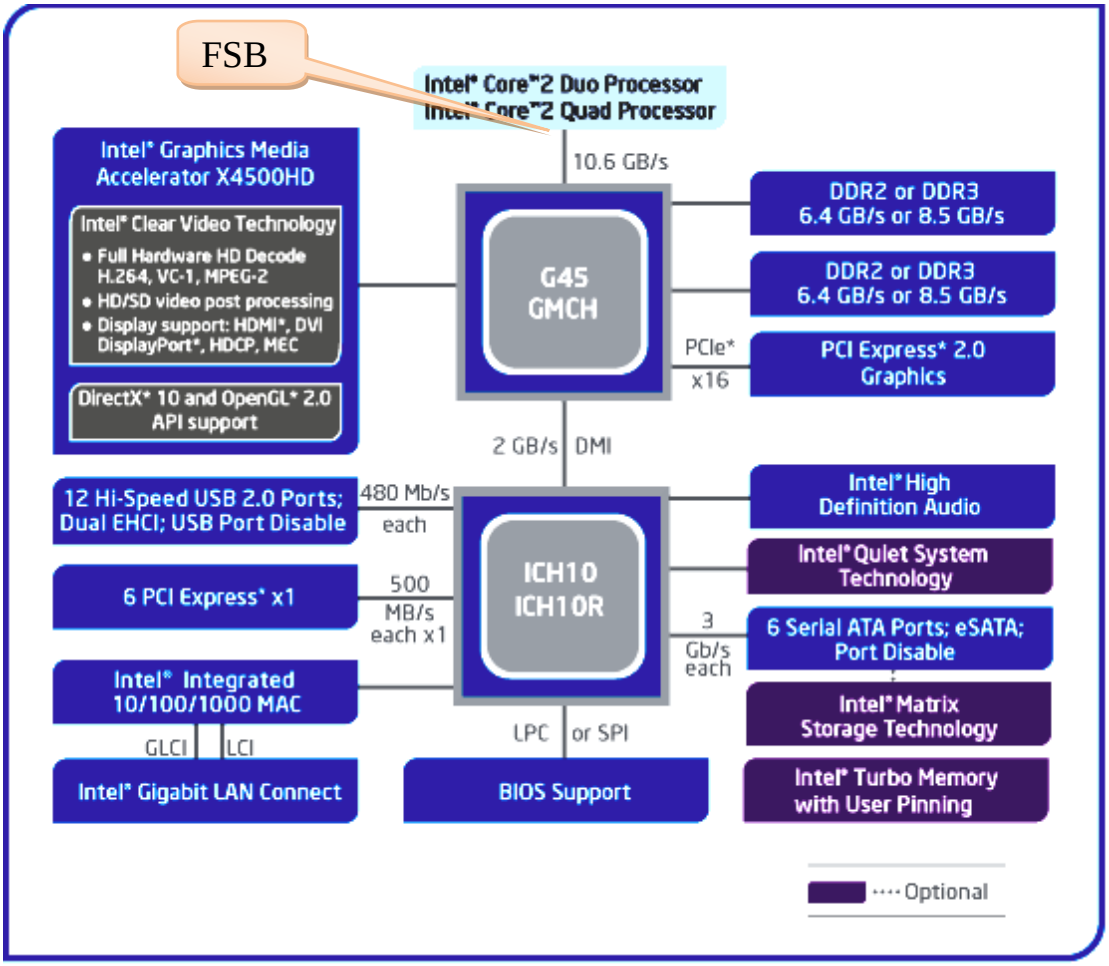


Рисунок 2.7. Принципиальная схема чипсета Intel® G45 Express

На схеме чипсета Intel® G45 Express мы видим:

– системная шина (FSB – front side bus) может работать с частотой 1333/1066/800 МГц и поддерживает скорость работы процессора (Intel® Core™2 Quad, Intel® Core™2 Duo и др.) с контроллером-концентратором памяти (G45 GMCH) 10,6 Гбит/с;
– MCH имеет 2 контроллера для работы с памятью типа DDR3 (скорость 8,5 Гбит/с) или DDR2 (6,4 Гбит/с) – двухканальная память,

скорость работы увеличивается при использовании двух модулей памяти;

- MCH имеет интегрированную видеоподсистему ***Intel® Graphics Media Accelerator X4500HD*** с поддержкой Microsoft DirectX 10, Shader Model 4.0 и OpenGL 2.0 (поддерживают высокий уровень качества Windows Vista Aero). Поддерживаются интерфейсы с оборудованием HDMI, DVI, HDCP, MEC;
- поддержка портов HDMI (интерфейс мультимедиа высокого разрешения), DisplayPort (с разрешением до 2560 × 1600) и DVI;
- MCH поддерживает интерфейс работы с видеокартами PCI Express 2.0 Graphics x16, скорость работы 16 Гбит/с;
- MCH через межчипсетную шину DMI обменивается информацией с контроллером-концентратором ввода-вывода (ICH10) со скоростью 2 Гбит/с;
- ICH поддерживает до 12-ти портов USB 2.0 для подключения внешних устройств (скорость работы 480 Мбит/с каждый);
- ICH поддерживает работу с шестью слотами PCI Express x1 для подключения плат различных внешних устройств (скорость работы 500 Мбит/с);
- ICH имеет интегрированный сетевой адаптер Intel Gigabit LAN Connect;
- ICH обеспечивает работу с интегрированной 8-ми канальной (7.1) аудиоподсистемой типа High Definition Audio;
- ICH поддерживает до 6-ти контроллеров Serial ATA (SATA) (скорость работы 3 Гбит/с каждый) для подключения дисковых устройств.

Таким образом, для материнских плат, созданных на основе данного чипсета, характерно использование памяти типа DDR3 или DDR2, винчестеров SATA, они имеют интегрированные видеоподсистему, аудиоподсистему и один или два гигабитных сетевых адаптера.

Принципиальная схема двухчипового набора микросхем Intel® P55 для процессоров Core i7 и Core i5 показана на рисунке 2.8. Этот чипсет предназначен для процессоров, в которых интегрированы контроллер памяти и контроллер графического интерфейса.

Модель P55 пятой серии чипсетов предназначен для производительных компьютеров, оснащенных дискретными графическими адаптерами; H57 и H55 разработаны для систем с интегрированной графикой, Q57 – для бизнес-систем и компьютеров масштаба предприятия (в данном слу-

чае реализован ряд специализированных функций для управления парком компьютеров).

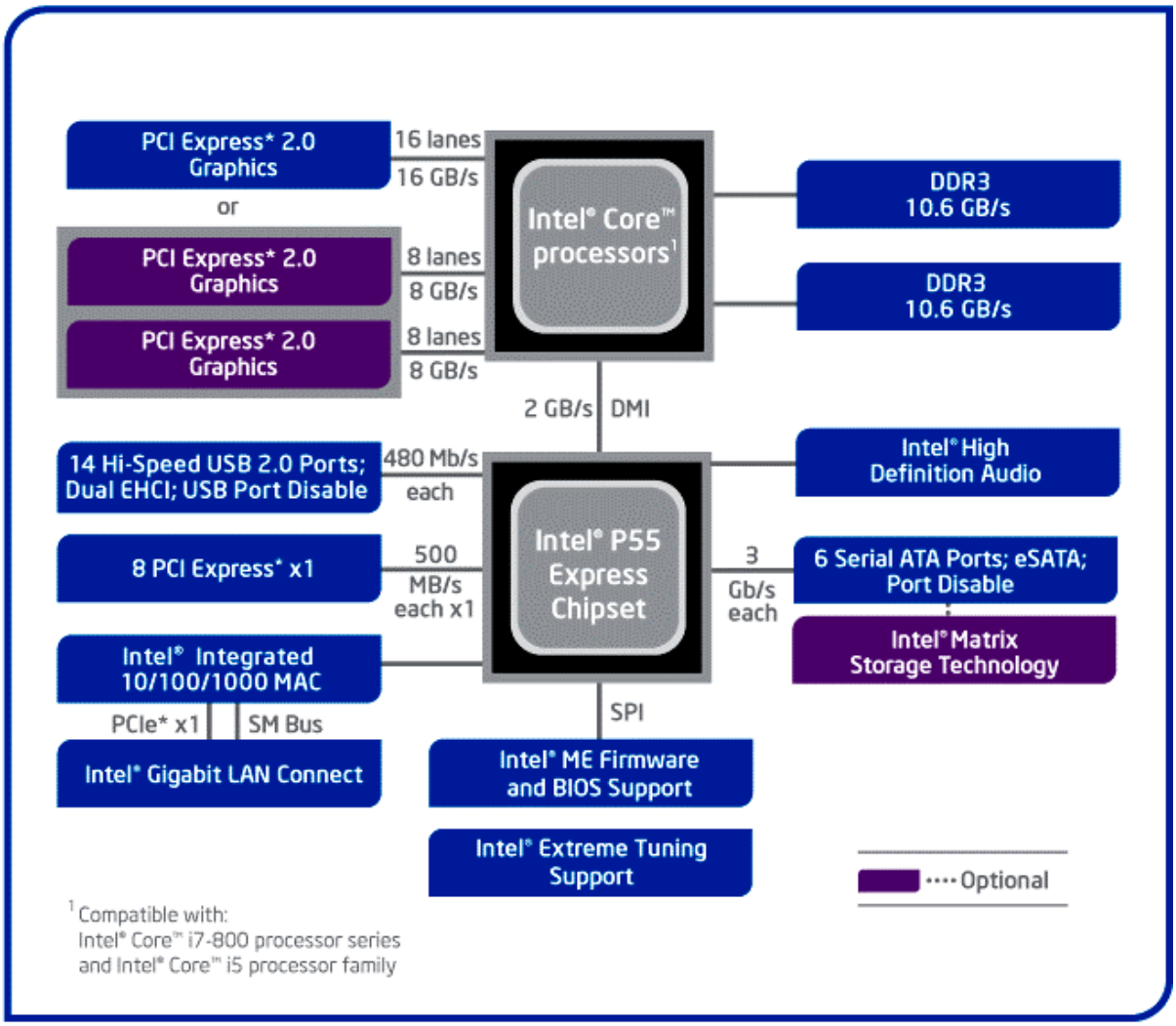


Рисунок 2.8. Блок-схема чипсета Intel® P55 Express

Принципиальная схема чипсета AMD 780G для процессоров AMD Phenom и Athlon показана на рисунке 2.9.

Особенности этого чипсета: северный мост не имеет контроллеров для работы с памятью (которые интегрированы в процессор), а отвечает за работу с шиной PCI Express 2.0 для подключения нескольких видеокарт и работу видеоинтерфейсов DVI, HDMI, VGA/ Версия 690G поддерживает два независимых видеоконтроллера – с аналоговым выходом VGA и цифровыми DVI и HDMI.

Южный мост содержит:

- контроллер SATA 2.0 на 6 портов (до 3 Гбит/с);

- поддержка режимов RAID для всех SATA-портов: 0, 1, 5 и 10;
- один канал PATA (IDE) на 2 устройства ATA133;
- 12 портов USB 2.0;
- шину PCI (до 6 слотов);
- звуковую подсистему HD Audio;

Более подробную характеристику чипсетов можно найти на сайтах фирм-производителей.

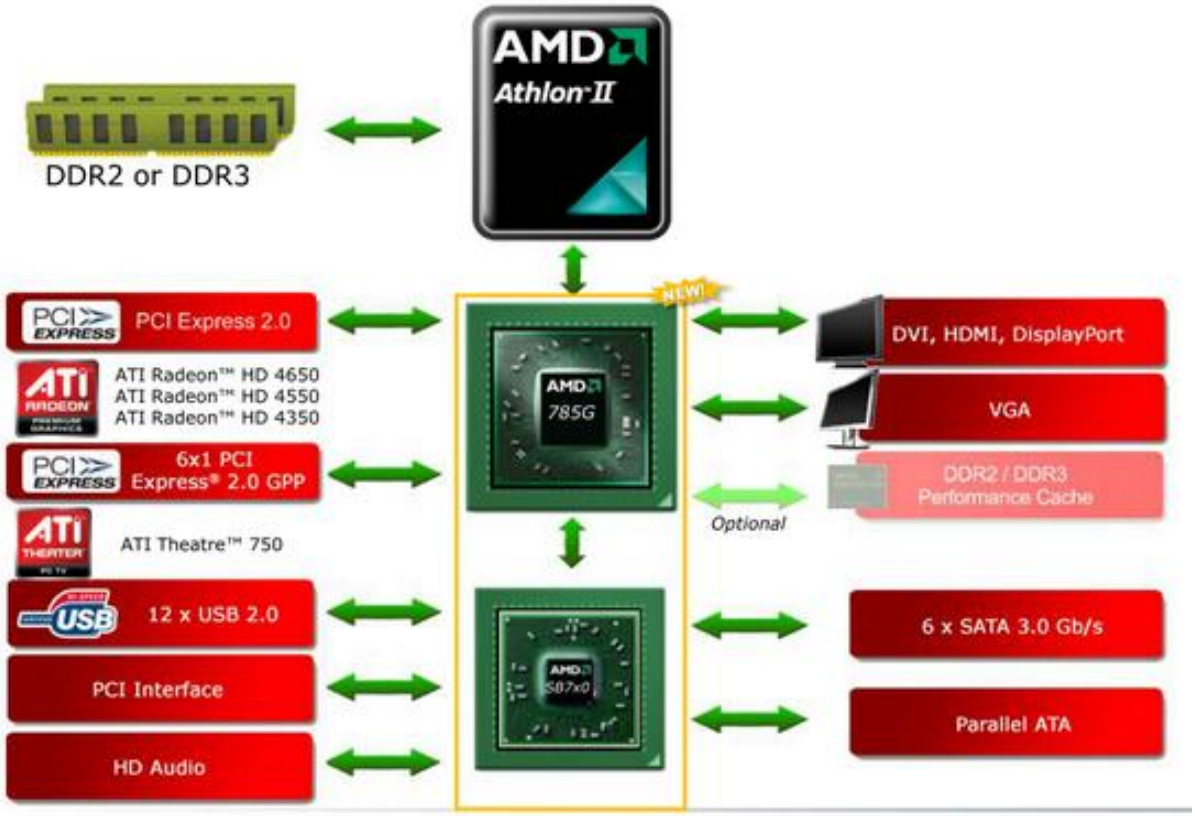


Рисунок 2.9. Чипсет AMD 780G

2.3 Материнская плата

Материнская плата – печатная плата, на которой осуществляется монтаж микросхем и других компонентов компьютерной системы.

Название происходит от английского *motherboard*, иногда используется сокращение *MB* или слово *mainboard* – главная плата, в русских источниках используется также название *системная плата*.

На материнской плате располагаются микросхемы чипсета, разъемы для подключения центрального процессора, оперативной памяти, диско-

вых устройств, графической платы, звуковой платы и дополнительных внешних устройств.

Существуют различные спецификации (форм-факторы), описывающие конструктивные особенности материнских плат.

Форм-фактор – стандарт (спецификация), определяющий размеры материнской платы, расположение крепежных отверстий, сокета центрального процессора, слотов оперативной памяти, интерфейсов шин, портов ввода/вывода, разъемов для подключения питания.

Наиболее известен форм-фактор ATX (Advanced Technology eXtended), с достаточно большим размером, что позволяет производителям размещать на материнской плате наибольшее количество устройств. Самые распространенные спецификации форм-факторов показаны в таблице 2.5. Полные спецификации приведены на сайте *http://www.formfactors.org/formfactor.asp*.

Таблица 2.5. Некоторые форм-факторы материнских плат

Форм-фактор	Размеры, мм	Примечание
ATX	305 × 244	для корпусов типов MiniTower, FullTower
MicroATX	244 × 244	меньше слотов, чем у ATX
MiniATX	284 × 208	для системных блоков типа Tower и Desktop
Mini-ITX	171,45× 171,45	новый форм-фактора Intel и VIA (2007 г.)
BTX	325 × 267	до 7 слотов и 10 отверстий для монтажа платы
MicroBTX	264 × 267	до 4 слотов и 7 отверстий для монтажа платы
PicoBTX	203 × 267	1 слот и 4 отверстия для монтажа платы
WTX	355,6 × 425,4	для высокопроизводительных рабочих станций и серверов среднего уровня

Известны и другие форм-факторы, редко используемые в настоящее время (LPX, MiniLPX, NLX и пр.). От форм-фактора зависит тип системного блока ПК.

Пример расположения портов ввода/вывода на задней панели материнской платы, приведенный в спецификации ATX, показан на рисунке 2.10.

Наиболее важными микросхемами материнской платы являются северный и южный мост чипсета. Именно чипсет определяет, в значитель-

ной степени, особенности материнской платы и то, кие устройства могут подключаться к ней.

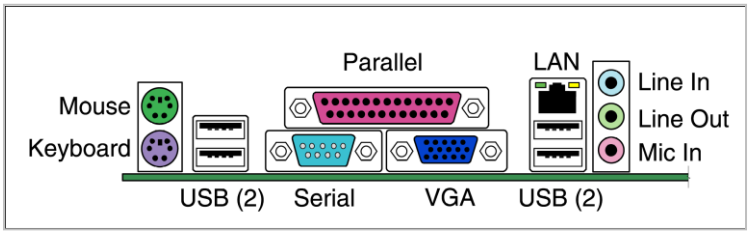


Рисунок 2.10. Пример портов внешних устройств на задней панели

В настоящее время фирма Intel ввела классификацию своих материнских плат по их назначению (см. таблицу 2.6), полный список приведен по адресу *http://www.intel.com/cd/products/services/emea/rus/motherboards/desktop/323255.htm*

Таблица 2.6. Серии материнских плат фирмы Intel для настольных ПК

Характеристика	Серия				
	Extreme	**Media**	**Executive**	**Classic**	**Essential**
Назначение	Extreme-процессоры	Медиа-центры	Для бизнеса	Классика	Экономичные
Плата, пример	***DP55KG***	***DP55WG***	***DQ43AP***	***DG41TY***	***DG41RQ***
Чипсет	Intel® P55	Intel® P55	Intel® Q43	Intel® G41	Intel® G41
Разъем процессора	LGA1156	LGA1156	LGA775	LGA775	LGA775
Форм-фактор	ATX	ATX	MicroATX	MicroATX	MicroATX

В соответствии с описанием фирмы-производителя материнская плата Intel® Desktop Board DG41TY имеет следующие основные характеристики:

Форм-фактор	MicroATX
Процессор	Поддержка Intel® Core®2 Quad в корпусе LGA775 Поддержка Intel® Core®2 Duo в корпусе LGA775 Поддержка Intel® Pentium® Dual-Core процессоров в корпусе LGA775 Поддержка Intel® Celeron® в корпусе LGA775
Память	Два 240-контактных разъемов DIMM для двухканальных модулей DDR2 SDRAM Поддержка модулей памяти DIMM DDR2 800 или DDR2 667 МГц

	Поддержка до 8 ГБ системной памяти
Чипсет	Intel® G41 Express
Аудио	Подсистема *Intel® High Definition Audio* в следующей конфигурации: 6-канальная (5.1) аудиоподсистема с поддержкой аудиокодека Realtek ALC888VC
Видео	Встроенная графическая подсистема *Intel® Graphics Media Accelerator X4500*
Сеть	Гигабитная (10/100/1000 Мбит/сек) сетевая подсистема на базе устройства Realtek RTL 8111D
Периферийный интерфейс	8 портов *USB 2.0* Один последовательный порт Четыре интерфейса IDE *Serial ATA* Один интерфейс Parallel ATA IDE с поддержкой протоколов UDMA 33 и ATA-66/100 Интерфейс флоппи-дисковода Порты PS/2 для клавиатуры и мыши
Возможности расширения	Один разъем для дополнительной платы *PCI Express* x16 Один разъем PCI Express x1 Два разъема PCI Conventional

Внешний вид материнской платы Intel DG41TY показан на рисунке 2.11.

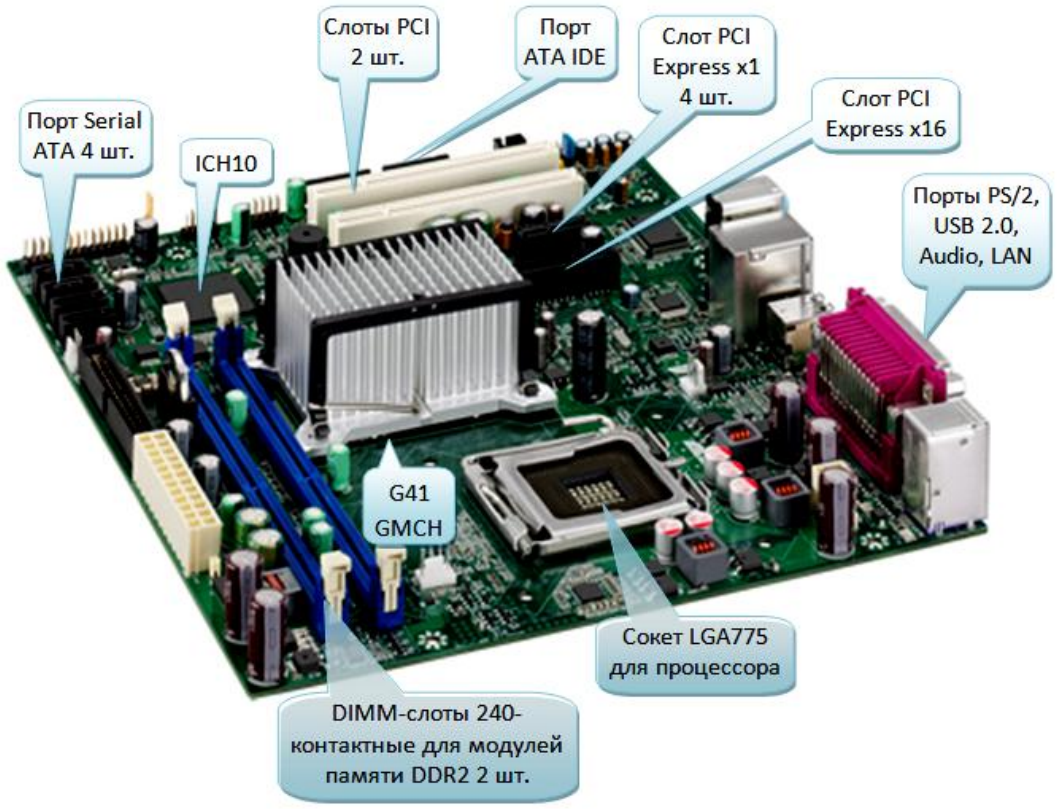


Рисунок 2.11. Материнская плата Intel DG41TY

Материнская плата имеет радиатор пассивного охлаждения для микросхем чипсета MCH, кроме того, имеет возможность подключения дополнительных вентиляторов для поддержания необходимого температурного режима работы микросхем.

Схематичное изображение основных компонентов материнской платы Gigabyte GA-P35-DQ6 (на чипсете Intel P35) приведено на рисунке 2.12.

На этом рисунке показана еще одна маленькая, но очень важная микросхема материнской платы – **BIOS** (Basic Input-Output System – базовая система ввода-вывода), а также устройства для ее питания – микросхема BP_BIOS и аккумуляторная батарейка (BATTERY). Назначение этой микросхемы с постоянным питанием и наличием собственной энергонезависимой памяти (перепрограммируемая Flash EEPROM):

- конфигурирование компьютера (программа BIOS Setup) и сохранение параметров конфигурации;
- выполнение программы часов и календаря компьютера;
- при включении компьютера:
  - выполнение программы самотестирования;
  - поиск главной загрузочной записи (MBR) на дисках;

- далее MBR выполняет сканирование дисков для поиска загрузочного раздела, и, если он найден, запускает в работу операционную систему.

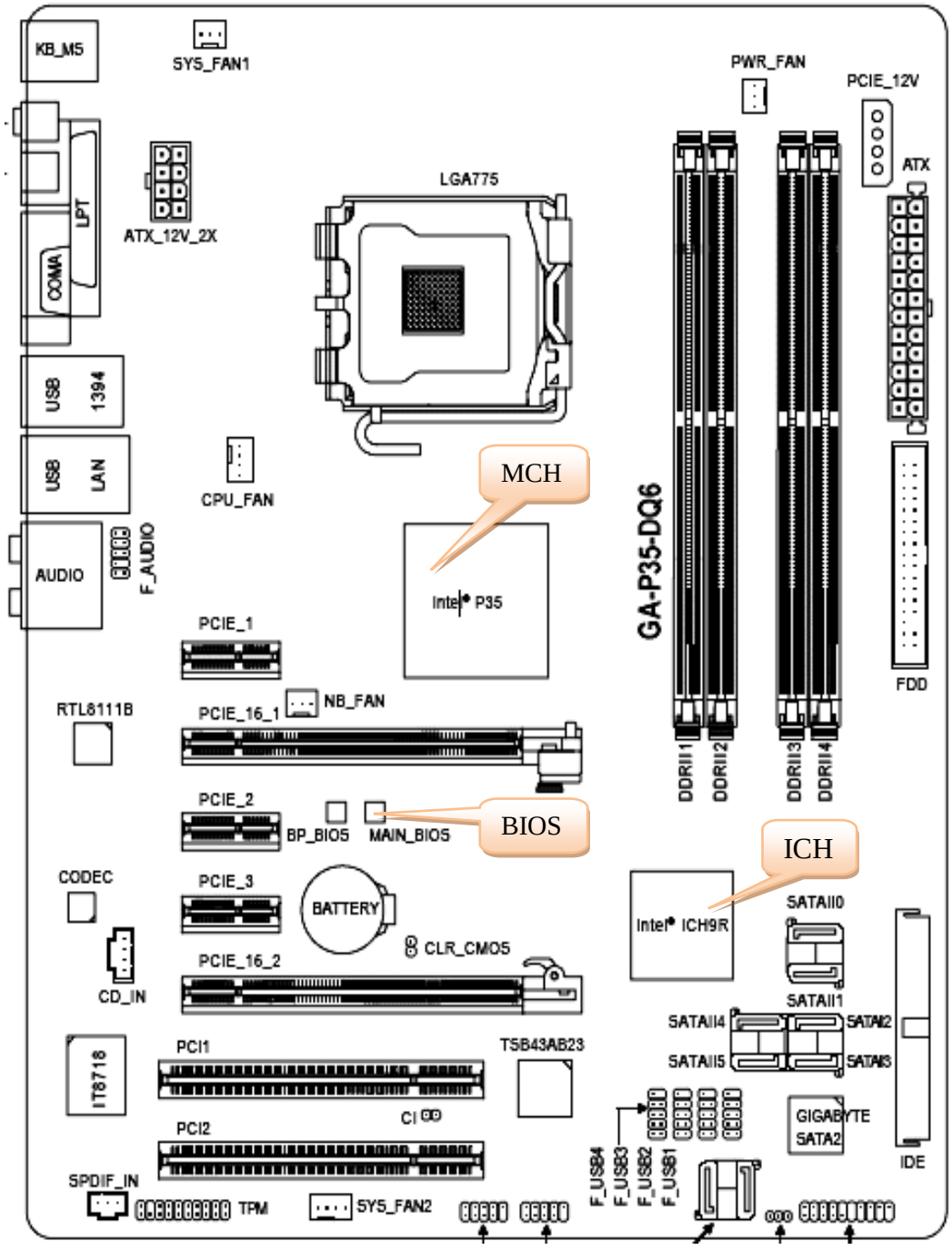


Рисунок 2.12. Схема материнской платы Gigabyte P35-DQ6

В программе BIOS Setup (которая запускается нажатием клавиши Delete или другой в момент включения компьютера) есть возможность на-

строить порядок устройств для поиска загрузчика операционной системы – если необходимо, ее можно загрузить с CD, DVD диска или даже через USB-порт. Эта возможность необходима для установки новой операционной системы на компьютер (или ее переустановки).

Назначение **слотов** (разъемов) на материнской плате – подключение различных плат сопряжения с внешними устройствами и реализация дополнительных возможностей:

- **PCI Express x16** – только для соответствующих видеоплат;
- **PCI Express x1** – новые звуковые, сетевые платы;
- **PCI** – старые звуковые, сетевые платы, внутренние модемы, FM-, TV-тюнеры.

PCI (Peripheral Component Interconnect bus) – шина для подсоединения периферийных устройств. Впервые массово применяться начала для Pentium-систем.

Шина PCI Express x16 пришла на смену шине AGP, ранее использовавшейся для подключения видеокарт.

Внутренние порты, расположенные на материнской плате, служат для подключения устройств:

- **Serial ATA** – дисковые устройства (HD, DVD) соответствующего типа;
- **ATA IDE** – дисковые устройства (HD, CD, DVD) соответствующего типа.

ATA (Advanced Technology Attachment) – спецификация интерфейса для подключения дисковых устройств. **IDE** (Integrated Drive Electronics) – диск со встроенным контроллером управления для подключения с помощью параллельного интерфейса спецификации ATA.

Serial ATA – новый последовательный интерфейс подключения дисков. На рисунке 2.13 показаны кабели подключения для дисков параллельного и последовательного интерфейсов.

Внешний порт – разъем на задней или передней панели системного блока, используется для подключения различных устройств:

- **USB 2.0** – принтеры, сканеры, внешние модемы, TV-тюнеры, клавиатура, мышь, внешние дисковые накопители, Flash Drive и многое другое;
- **IEEE-1394** – цифровые видеокамеры, дисковые устройства;

Рисунок 2.13. Кабели подключения IDE и SATA

- **PS/2** – клавиатура и мышь;
- при наличии интегрированной видеоподсистемы разъем **DVI** – цифровой выход на монитор или **VGA** – аналоговый (см. рисунок 2.8);
- при наличии интегрированной аудиоподсистемы разъемы для подключения колонок и микрофона.

Кроме того, могут присутствовать устаревшие на сегодняшний день порты:

- **COM** – мышь, модем и пр.;
- **LPT** – принтер, сканер.

Новые интерфейсы подключения внешних устройств – USB, SATA, IEEE-1394 (FireWire), DVI – последовательные шины (Serial Bus) с небольшим количеством проводов (4 – 7) в кабеле.

Параллельные шины ATA IDE, LPT имеют значительно большее количество проводников в кабеле (IDE 40 или 80 проводов, LPT – 18) и, соответственно, контактов на разъемах.

Скорость работы современных последовательных шин выше, чем параллельных.

Наиболее часто используется в настоящее время порт USB (Universal Serial Bus – универсальная последовательная шина), которая имеет 2 типа: USB и USB 2.0.

Особенности этой шины:

- высокая скорость работы – до 480 Мб/с для USB 2.0;
- длина кабеля подключения – до 5 м;
- количество подключаемых устройств (включая размножители) – до 127;
- возможно подключение устройств с различными скоростями обмена;
- возможность подключать и отключать устройства при работающем компьютере;
- возможность подавать питание напряжением 5 V на маломощные периферийных устройства (ток до 500 mA);
- кабель подключения имеет различные разъемы для подключения к компьютеру (тип «А») и к внешнему устройству (тип «В»).

Шина USB предоставляет в полной мере возможность использования технологии Plug&Play для периферийных устройств, т. е. при подключении нового устройства к работающему компьютеру (наиболее часто в настоящее время – флэш-накопителей, внешних жестких дисков, «мышь» на ноутбуках и т. п.) происходит его автоматическое распознавание и установка соответствующих драйверов, необходимых для работы с этим

устройством. Для использования большого количества USB портов применяются USB-концентраторы (хабы), как внешние, так и встроенные в монитор, клавиатуру и пр.

В прайс-листах фирм по продаже компьютеров обычно приводится краткое обозначение материнской платы примерно следующего вида:

MB Intel DG45FC LGA775, G45, DDR2 800, SATA2 RAID, PCI-E, 8ch Audio, GBL, DVI+HDMI, mini-ITX,

что означает:

материнская плата **Intel DG45FC**, разъем процессора **LGA775**, чипсет **G45**, память **DDR2** с частотой **800** МГц, дисковый интерфейс **SATA2** с аппаратной поддержкой дисковых массивов (**RAID),** шина и слоты **PCI Express**, звуковой адаптер 7.1 (**8ch Audio**), интегрированные сетевой адаптер Gb Ethernet (**GBL**), видеоадаптер с выходами **DVI** и **HDMI**, форм-фактор **mini-ITX** (171,45 мм x 171,45 мм).

Стоимость материнских плат на чипсетах G31 – P45 от 2000 до 6000 руб.

Корпус системного блока ПК подбирается в зависимости от форм-фактора материнской платы и типа процессора. Бывает двух видов – TOWER (вертикальный) и DESKTOP (горизонтальный), второй тип часто имеют ПК ведущих фирм-производителей (IBM, Dell, Hewlett-Packard/Compaq, Acer и др.). Обычно в комплекте с корпусом идет блок питания, мощность которого тоже может быть выбрана в соответствии с типом процессора, видеокарты и количеством периферийных устройств и лежит в настоящее время обычно в диапазоне 350 – 450 Ватт. Стоимость обычных корпусов 1500 – 3000 руб., однако существуют и корпуса стоимостью более 10000 руб. (например, для домашних медиацентров).

Выпускаются также ПК-моноблоки (фирмы HP, Apple и др.) – в этом случае все устройства, размещаемые обычно в системном блоке, располагаются в одном корпусе с жидкокристаллическим монитором (может быть сенсорным). Толщина такого моноблока лишь немного больше толщины обычного монитора.

2.4 Оперативная память

Оперативная память – один из основных компонентов компьютера, предназначенный для хранения информации (программ и обрабатываемых программами данных) во время исполнения программ на компьютере.

Кроме основной оперативной памяти (RAM – Random Access Memory, память с произвольным доступом) современные персональные компьютеры имеют память видеокарты (с которой работает процессор видеокарты), а также сверхоперативную память центрального процессора (кэш первого, второго и иногда третьего уровня).

Для долговременного хранения информации используются дисковые и пр. устройства, которые иногда называют внешней памятью (описаны далее в разделе 2.5).

Оперативная память, так же, как и процессоры, имеют длительную историю своего развития – от первых микросхем памяти, запаянных на материнской плате первых персональных компьютеров до современных сменных модулей емкостью в несколько Гбайт.

Чипсет, используемый на компьютере, определяет тип оперативной памяти, который можно использовать. Ранее при описании чипсета Intel® G45 Express (раздел 2.2) было указано, что он может работать с двухканальной памятью типа DDR3 (скорость 8,5 Гбит/с) или DDR2 (6,4 Гбит/с). Полное наименование такой памяти – DDR3 SDRAM.

Современная оперативная память – динамическая (DRAM), содержимое ее остается неизменным в течение очень короткого промежутка времени, поэтому память должна периодически обновляться (регенерироваться). Запоминающим элементом динамической памяти является конденсатор, который может находиться в заряженном или разряженном состоянии. Если конденсатор заряжен, то в ячейку памяти записана логическая 1, если разряжен – логический 0. В идеальном конденсаторе заряд может сохраняться неограниченное время, в реальном конденсаторе существует ток утечки, поэтому информация, записанная в динамическую память, со временем будет утрачена в результате разрядки конденсаторов.

Единственным способом регенерации хранимой в памяти информации является выполнение операций чтения и повторной записи данных, выполняемых через определенные промежутки времени (например, каждые 2 мс) всех ячеек памяти. В эти моменты процессор находится в состоянии ожидания. Регенерация памяти происходит также при выполнении каждой операции чтения данных.

Ячейки памяти организованы в матрицу, состоящую из строк и столбцов. Полный адрес ячейки данных включает два компонента – адрес строки и адрес столбца. При чтении информации из памяти вначале считывается информация всех ячеек памяти для строки с заданным адресом и помещается в буфер ввода/вывода. Далее в соответствии с адресом столб-

ца данные выбираются из буфера ввода/вывода и поступают на выход динамической памяти.

Микросхемы памяти состоят из отдельных матриц, каждая из которых имеет собственную линию чтения/записи. В этом случае одновременно считывается или записывается несколько бит информации. Количество линий определяет разрядность шины ввода/вывода. Современные микросхемы памяти 64-х разрядные.

Емкость матрицы памяти (глубиной адресного пространства, address depth) и их количество (разрядность) определяет общий объем микросхемы памяти, которые размещаются на модулях памяти (DIMM – модули). DIMM-модуль (Dual In-Line Memory Module) имеет с двух сторон электрически независимые контакты для подключения к шине контроллера памяти через слот, в который они вставляются.

Наиболее распространенными являются 240-контактные 64-разрядные модули DIMM, имеющие по 120 контактов с каждой стороны (см. рисунок 2.14).

Хотя число контактов у модулей DDR2 и DDR3 одинаково, слоты, предназначенные для памяти разного типа, отличаются расположением «ключа» - прореди между контактами. Поэтому, установить DDR3 SDRAM в DDR2 DIMM и наоборот не удастся.

SDRAM – микросхема синхронной динамической памяти (Synchronous DRAM), в которой процессы записи и считывания информации строго привязаны к тактам системной шины.

Рисунок 2.14. Модули памяти DDR3

DDR SDRAM (Double Date Rate – удвоенная скорость передачи данных) является следующим поколением SDRAM. В отличие от SDRAM память этого типа дает возможность выполнять два обращения к памяти за время одного такта системной шины.

DDR2 и DDR3 – последующие модификации памяти DDR SDRAM, позволяющие за один такт записывать или считывать 4 (DDR2) или 8 (DDR3) блоков данных, имеют пониженное напряжение питание (DDR - 2,5 В, DDR2 - 1,8 В, DDR3 - 1,5 В) и энергопотребление.

В настоящее время память DDR2 работает на частоте от 533 МГц (обозначаемая, как PC2-4200 по скорости передачи данных 4200 Мб/с) до 1066 МГц (PC2-8500).

Память DDR3 выпускается для работы с частотой от 1066 МГц до 2000 МГц, маркировка DDR3 в зависимости от частоты: PC3-8500 … PC3-16000.

В обозначениях модулей памяти встречаются также обозначения:

- ECC – память с коррекцией ошибок (Error Correct Code) – имеет дополнительную 8-разрядную микросхему памяти, в результате модули с ECC являются 72- разрядным (в отличие от стандартных 64-разрядых модулей);
- Full Buffered (FB) – память с буферизацией данных, имеет дополнительные микросхемы регистров для буферизации данных, наиболее часто используются в регистровых модулях (Registered) для серверов.

Стоимость памяти для ПК достаточно высока, данные таблицы 2.7 позволяют получить сравнительные данные о различных модулях памяти.

Таблица 2.7. Модули памяти и их примерная цена (декабрь 2009 г.)

Наименование модуля	Цена, руб.
Kingston DDR2 1GB (PC2-5300) 667MHz	680
Kingston DDR2 1GB (PC2-6400) 800MHz	960
Kingston DDR2 2GB (PC2-5300) 667MHz	1640
Kingston DDR2 2GB (PC2-8500) 1066MHz (2 x 1Gb)	1766
Kingston DDR3 1GB (PC3-8500) 1066MHz	1020
Kingston DDR3 1GB (PC3- 10660) 1333MHz	1054
Kingston DDR3 2GB (PC3- 10667) 1333MHz (2 x 1Gb)	1720
Kingston DDR3 4GB (PC3-8500) 1066MHz (2 x 2Gb)	4640
Kingston DDR3 4GB (PC3-10667) 1333MHz (2 x 2GB)	4730

Необходимый объем оперативной памяти для настольных персональных компьютеров зависит в первую очередь от операционной системы и от приложений пользователя, которые используются на ПК.

До недавнего времени наибольшее применение находили модули памяти DDR2 емкостью 1 Гбайт (2 модуля в комплекте для двухканальной памяти), для семейства процессоров Core i3/i5 используют 2 модуля по 2 Гб DDR3.

2.5 Устройства хранения информации

Устройства хранения информации (внешняя память) – компоненты компьютера, позволяющие практически неограниченное время сохранять большие объемы информации без потребления электроэнергии (энергонезависимые).

Первыми такими устройствами для ПК были **Floppy**-дисководы (FDD) и сменные дискеты – вначале пятидюймовые (5,25") емкостью 360 Кб и 1,2 Мб, затем трехдюймовые (3,5") емкостью 1,44 Мб. В настоящее время применяются редко в связи с широким распространением устройств флэш-памяти емкостью в несколько гигабайт.

Примечание. На больших ЭВМ в эпоху до появления ПК для хранения информации использовались бумажные перфокарты, перфоленты, магнитные ленты и диски.

2.5.1 Винчестер

Следующий этап в развитии устройств хранения информации для ПК – винчестеры (или жесткие диски – Hard Disk Drive, HDD), которые в настоящее время являются обязательным компонентом каждого настольного персонального компьютера. В принципе, могут существовать бездисковые сетевые рабочие станции и терминалы, с загрузкой операционной системы с сервера и работой с дисковыми устройствами сервера, но на практике такие ПК встречаются достаточно редко. В субноутбуках винчестер может отсутствовать, вместо него используется флэш-память большого объема.

Первый винчестер с обозначением 30/30 был создан корпорацией IBM в 1973 г. и позволял на двухстороннем алюминиевом диске сохранять 30 + 30 Мбайт данных. Название 30/30 совпадает с обозначением американской винтовки типа «винчестер», откуда и произошло наименование устройств хранения информации на жестких дисках.

Современные винчестеры имеют емкость 500 Гбайт и более. Винчестер представляет собой плоскую, герметически закрытую металлическую

коробку, внутри которой на общей оси находятся несколько жестких алюминиевых или стеклянных пластинок круглой формы (см. рисунок 2.15).

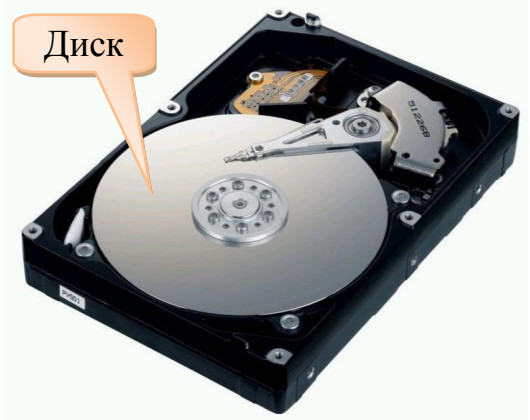


Рисунок 2.15. Винчестер SATA 1000 GB Samsung HD103UJ, 32MB кэш, 7200 об/мин (со снятой верхней крышкой)

Выпускаются винчестеры в основном двух размеров дисков (форм-фактор) – 3.5″ и 2.5″ (последние преимущественно для использования в ноутбуках). Поверхность каждого диска покрыта тонким ферромагнитным слоем (веществом, способным длительное время сохранять состояние намагниченности). Для чтения и записи данных в винчестерах используются плавающие на воздушной подушке магнитные головки. Зазор между магнитным слоем и головкой составляет десятые доли микрон. Для перемещения головок используется электромагнит, а для отслеживания местоположения применяется автоматическая система регулирования, которая по магнитным меткам на поверхности опорного диска позиционирует головки в нужном месте.

Пакет дисков вращается непрерывно с большой скоростью (7200 – 15000 об/мин). Запись данных осуществляется следующим образом. При изменении силы тока, проходящего через записывающую головку, происходит изменение напряженности магнитного поля в щели между поверхностью и головкой, что приводит к изменению напряженности магнитного поля на небольшом участке ферромагнитного покрытия диска. Намагниченные участки и не намагниченные соответствуют двоичным кодам 1 и 0. Для чтения данных используется магниторезистивный принцип, когда сопротивление полупроводниковой пленки, из которой сделана головка чтения, меняется в зависимости от напряженности магнитного поля на поверхности ферромагнитного диска. Электрический сигнал с головки усиливается и передается на обработку в контроллер жесткого диска, расположенный в чипсете.

Вся электроника, предназначенная для управления двигателем привода дисков и электромагнитом головок, а также для чтения и записи данных, располагается на небольшой печатной плате, укрепленной снаружи корпуса.

Винчестер может иметь до десяти дисков. На поверхности дисков размечаются траектории записи информации – концентрические окружности, которые называются дорожками или треками (track) и сектора на каждой дорожке. Каждая дорожка имеет свой номер. Дорожки с одинаковыми номерами, расположенные одна над другой на разных дисках образуют цилиндр. Каждый сектор на дорожке также имеет номер, нумерация начинается с единицы. В один сектор может быть записано 571 байт двоичной информации: 512 байт отведено для записи нужной информации, остальные под заголовок (префикс), определяющий начало и номер секции и окончание (суффикс), где записана контрольная сумма, нужная для проверки целостности хранимых данных. Секторы и дорожки образуются во время низкоуровневого форматирования диска на заводе-изготовителе. В новых винчестерах число секторов на внешних дорожках больше, чем на внутренних, и для них используют линейный способ адресации блоков LBA (Linear Block Addressing), когда все секторы нумеруются последовательно от первого сектора нулевой дорожки нулевого цилиндра до последнего на крайней внутренней дорожке.

К современным винчестерам неприменима операция низкоуровневого форматирования пользователем с помощью специальных программ (возможна для гибких дисков и старых винчестеров).

Несколько секторов образуют кластер – наименьшую адресуемую единицу на диске для используемой файловой системы. Форматирование с разметкой кластеров и созданием файловой системы на диске выполняется с помощью специальной программы, входящей в состав операционной системы (иногда на стадии установки этой системы). Неформатированный диск не может быть использован для записи файлов.

До форматирования жесткий диск можно разбить на логические диски (раздел винчестера или том). Это удобно, поскольку наличие нескольких логических дисков упрощает структуризацию данных, хранящихся на жестком диске.

Существует огромное количество разных моделей жестких дисков многих фирм, таких как Seagate, Maxtor, Western Digital, Quantum, Fujitsu, Hitachi, Samsung, Toshiba и пр. Для обеспечения совместимости винчестеров, разработаны стандарты на их характеристики, определяющие номенклатуру соединительных проводников, их размещение в переходных разъемах, электрические параметры сигналов.

Самый распространенный до недавнего времени тип винчестера по интерфейсу подключения был IDE (Integrated Drive Electronics). Винчестер типа IDE использует спецификацию параллельного интерфейса ATA (Ad-

vanced Technology Attachment). В настоящее время новые чипсеты поддерживают новую спецификацию подключения HDD – последовательный интерфейс SerialATA и SerialATA II. Соответственно новые винчестеры имеют тип интерфейса SerialATA. Особняком стоит SCSI (Small Computer System Interface – интерфейс малых компьютеров), так этот интерфейс используется, в основном, в серверах и устройства SCSI довольно дороги.

ATA – один из старейших стандартов обмена данными, был разработан в 1989 году тремя компаниями: Imprimus – подразделением Control Data Corporation, Western Digital и Compaq. Представляет собой параллельный шинный интерфейс. Первый стандарт ATA был утвержден в 1994 году национальным комитетом по стандартам информационных технологий (NCITS). ATA и его дальнейшие варианты, такие как ATA-2, ATA-3, и т. д. – официальные названия стандарта. IDE, EIDE, UltraATA, и т.д. – маркетинговые термины, используемые производителями винчестеров. IDE-интерфейс позволяет обмениваться данными со скоростью до 2 Мбайт/с.

Serial ATA – последовательный шинный интерфейс для передачи данных, пришел на замену ATA. Был разработан компанией Intel совместно с IBM, Dell, Seagate, Quantum, Maxtor, APT Technologies и др. в 2000 году. Обеспечивает пропускную способность до 3 Гбит/с.

Среди других параметров, которые влияют на быстродействие HDD следует отметить следующие:

- скорость вращения дисков – в наше время обычно 7200 об/мин, накопители SCSI 10000 – 15000 об/мин;
- объем кэш-памяти – во всех современных дисковых накопителях устанавливается кэш-буфер, ускоряющий обмен данными; чем больше его емкость, тем выше вероятность того, что в кэш-памяти будет необходимая информация, которую не надо считывать с диска (этот процесс в тысячи раз медленней); емкость кэш-буфера в современных винчестерах 8 – 16 Мбайт;
- среднее время доступа – время (в миллисекундах), на протяжении которого блок головок смещается с одного цилиндра на другой. У современных винчестеров составляет 5 – 10 миллисекунд;
- время задержки – это время от момента позиционирования блока головок на нужный цилиндр до позицирования конкретной головки на конкретный сектор, другими словами, это время поиска нужного сектора (около 4 мс);
- скорость обмена – определяет объем данных, которые могут быть переданы из накопителя к микропроцессору и в обратном направле-

нии за определенные промежутки времени; максимальное значение этого параметра равно пропускной способности дискового интерфейса и зависит от того, какой режим используется: PIO (Programmed I/O) или DMA; в режиме PIO обмен данными между диском и контроллером происходит при непосредственном участии центрального процессора, чем больше номер режима PIO, тем выше скорость обмена; работа в режиме DMA (Direct Memory Access) разрешает передавать данные непосредственно в оперативную память без участия процессора; скорость передачи данных в современных жестких дисках до 300 Мбайт/с.

Несмотря на внешнюю простоту и наличие всего двух разъемов – информационного и питания, винчестеры являются очень сложными устройствами, которые требуют бережного и правильного обращения – следует избегать тряски и резкого перемещения работающего винчестера, это может привести к разрушению внутренних механических узлов и повреждению магнитных головок и поверхности дисков.

Современные винчестеры могут безотказно работать более 100 000 часов, система автоматической дефектации (S.M.A.R.T., Self-Monitoring, Analysis and Reporting Technology) незаметно для пользователя заменяет плохие кластеры на резервные (если в BIOS включен режим S.M.A.R.T.). Если на винчестере появляются плохие сектора и их количество постоянно увеличивается, то такой винчестер лучше заменить на новый. В настоящее время на винчестеры производителями устанавливается 1 – 3-летний гарантийный срок.

Винчестеры фирм Seagate и Hitachi объемом 500 Гб состоят из 5 дисков с плотностью записи 100 Гб на диск. Hitachi и Samsung успешно эксплуатируют в своих винчестерах диски объемом 125 Гб. Seagate наряду с WD имеют в составе своих продуктовых линеек винчестеры с плотностью записи 133 Гб на диск.

Лидеры рынка дисков класса 7200/3.5″, компании Seagate, Maxtor и WD, выпускают также **внешние** жесткие диски, выполненные в отдельном корпусе с блоком питания, интерфейсом USB или IEEE1394 (FireWire).

Некоторые модели выпускаемых в настоящее время винчестеров показаны в таблице 2.8.

Данные таблицы по стоимости единицы хранения информации свидетельствуют, что на текущий момент наиболее выгодным приобретением будет винчестер объемом 500 Гбайт.

Таблица 2.8. Сравнение стоимости винчестеров (сентябрь 2010 г.)

Марка жесткого диска	Цена, руб.	Цена за 1 Гб, руб.
IDE 500 Gb WESTERN DIGITAL WD5000AAKB (7200rpm) 16 Mb	2360	4,72
SATA 320Gb WD WD3200AAJS 3.5" Caviar SE SATA II/300 7200rpm 8Mb 3GB/S	1280	2,56
SATA 500Gb Seagate ST3500410SV 3.5" SV35.5 7200rpm 16Mb SATA II/300 27 дБ 70G	1820	3,64
SATA 1.5Tb Seagate ST31500341AS 3.5" Barracuda 7200.11 7200rpm 32Mb	2740	5,48
SATA 1.5Tb WD WD15EARS 3.5" Caviar Green 5400rpm 64Mb 29дБ	2490	4,98
Внешний жесткий диск 3.5" Seagate ST310005FDD2E1-RK 1Tb "FreeAgent Desk" USB2.0	3510	7,02
Внешний жесткий диск 3.5" Seagate ST315005EXD101-RK 1,5Tb, "Expansion" USB2.0 7200rpm	3760	7,52

2.5.2 Дисководы оптических дисков

Дисковод CD-ROM

В 1995 году в базовой конфигурации ПК появился первый привод оптических дисков – **CD-ROM** (Compact Disk Read Only Memory, постоянное запоминающее устройство компакт-дисков). Устройство использовало многослойные компакт-диски диаметром 120 мм и толщиной 1,2 мм, емкость диска 650 – 700 Мбайт.

CD-диск состоит из 4-х слоев (сверху-вниз):

1) защитный слой из поликарбоната;
2) слой для записи информации;
3) отражающий слой;
4) основа из поликарбоната.

Процесс изготовления диска состоит из операций напылением серебра или золота отражающего слоя на основу, нанесения на него прозрачного слоя для записи информации и выдавливании на нем углублений, образующих спиральную дорожку, идущую от центра диска к его краю. Для штамповки диска используют матрицу-прототип (мастер-диск) буду-

щего диска. После этого на поверхность диска наносят защитный слой из прозрачного пластика.

CD-ROM считывает информацию с диска с помощью лазерного луча с длиной волны 780 нм, который по разному отражается от поверхности диска (land – ленд) и углублений на поверхности (pit – пит, см. рисунок 2.16). Минимальный раз мер пита составляет 0,88 мкм, шаг дорожек – 1,5 мкм.

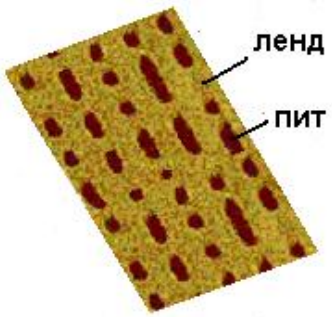


Рисунок 2.16. Поверхность CD

Информация при подготовке к записи на компакт-диске подвергается ряду сложных преобразований. Вначале используется ***помехоустойчивое кодирование***, затем осуществляется так называемое ***канальное кодирование***. В компакт-дисках применяется канальный код EFM (Eight to Fourteen Modulation), разработанный фирмой Philips для лазерной звукозаписи. Суть канального кодирования EFM заключается в том, что каждый байт информации заменяется 14-разрядным словом из специальной таблицы преобразования. К полученному таким образом 14-разрядному слову по определённому правилу добавляется ещё три так называемых соединительных разряда. В результате канального кодирования получается непрерывная последовательность бит, причём между двумя единицами никогда не может быть меньше двух или больше десяти нулей.

Именно эта последовательность бит и представлена на компакт-диске питами. Причём питы и ленды чередуются в моменты, соответствующие биту, равному 1 (см. рисунок 2.17). Если до этого был, например, ленд, то становится пит, и наоборот. Таким образом, питы и ленды с точки зрения представления информации равнозначны и характеризуют только временные интервалы между двумя единицами в последовательности бит.

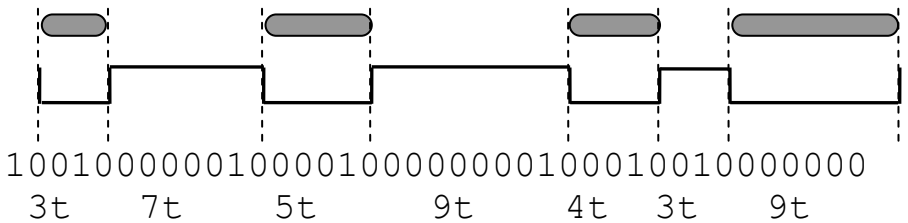


Рисунок 2.17. Схема преобразования при считывании информации с CD

При считывании информации луч лазера отражается от лендов и попадает в фотоприёмник, от питов свет тоже отражается, но из-за расфо-

кусировки луча в фотоприёмник не попадает. Далее высокочастотный информационный сигнал преобразуется компаратором в последовательность прямоугольных импульсов. Длительность каждого из полученных таким способом прямоугольных импульсов преобразуется в двоичный код. Каждые 14 последовательных битов декодируются в соответствии с таблицей кодирования EFM, соединительные разряды при этом отбрасываются.

Система кодирования обеспечивает также работу системы автослежения (автотрекинга), которой оснащены все приводы CD-ROM. Размеры каждого пита очень малы – порядка 0,6 мкм, а шаг спирали - всего 1,6 мкм. Назначение системы автослежения – точное позиционирование считывающей головки. Слежение осуществляется за дорожкой с данными по расположенным на ней питам, для нормальной работы системы питы не должны отсутствовать более 10 периодов тактовой частоты, что и обеспечивается системой кодирования.

Вторая задача, которая возникает при считывании информации с компакт-диска, – точное выделение временных интервалов, соответствующих периоду следования символов последовательного кода. Поддерживать скорость вращения диска с требуемой точностью практически невозможно. Поэтому для решения этой задачи временные интервалы определяются путем подстройки частоты тактового генератора под реальную скорость поступления информации с помощью инерционной системы автоподстройки, что опять-таки возможно выполнить благодаря использованию канального кода EFM при записи.

Дисковод CD-ROM содержит:

- электродвигатель, который вращает диск;
- оптическую систему, состоящую из лазерного излучателя, оптических линз и датчиков и предназначенную для считывания информации с поверхности диска;
- микропроцессор, который руководит механикой привода, оптической системой и декодирует прочитанную информацию в двоичный код.

Основные характеристики CD-ROM:

- скорость передачи данных – измеряется в кратных долях скорости проигрывателя аудио компакт-дисков (150 Кбайт/сек) и характеризует максимальную скорость с которой накопитель пересылает данные в оперативную память компьютера, например, 2-скоростной CD-ROM (2x CD-ROM) будет считывать данные с скоростью 300 Кбайт/сек., 50-скоростной (50x) - 7500 Кбайт/сек.;
- время доступа – время, нужное для поиска информации на диске, измеряется в миллисекундах.

Дисковод CD-RW

Устройство используется для записи информации на диски CD-R (однократная запись) и CD-RW (CD-ReWritable – перезаписываемый диск).

Внешне похож на CD-ROM и совместим с ним по размерам дисков и форматам записи. Запись данных осуществляется с помощью специального программного обеспечения или средств операционной системы. Скорость записи современных накопителей CD-R составляет 2x-48x.

Диск CD-R или CD-RW имеет 4 слоя (сверху-вниз):

1. защитный слой из поликарбоната;
2. активный слой для записи информации;
3. отражающий слой;
4. основа из поликарбоната.

Может присутствовать также этикетка диска, нанесенная краской на защитный слой.

Во время записи информации активный слой CD-R диска нагревается лазерным лучом (длина волны 780 нм) и в месте нагрева перестает пропускать свет. В результате отражающая способность на участках, подвергнутых действию луча лазера, уменьшается. Одним из типов активных слоев CD-R диска, широко используемых на сегодняшний день, является цианин (cyanine).

Активный слой CD-RW диска под воздействием лазерного луча при записи информации изменяет свое агрегатное состояние с переходом из кристаллического состояния в аморфное. Участки диска с аморфным состоянием активного слоя имеют меньшую степенью отражения лазерного луча при считывании информации. Процесс стирания информации с CD-RW заключается в обработке поверхности диска лучом лазера меньшей интенсивности, чем при записи, в результате чего активный слой полностью возвращается в кристаллическое состояние.

Основа CD-R и CD-RW имеет не гладкую поверхность: при изготовлении она получает разметку – сплошную спиральную канавку (Pregroove), необходимую для работы системы автослежения, заполненную органическим красителем, который немного хуже отражает свет, чем пластик основы. Для синхронизации частоты тактового генератора со скоростью вращения диска в CD-R канавка выполнена не в виде ровной спирали, а с микроскопическими отклонениями – вобуляцией (см. рисунок 2.18).

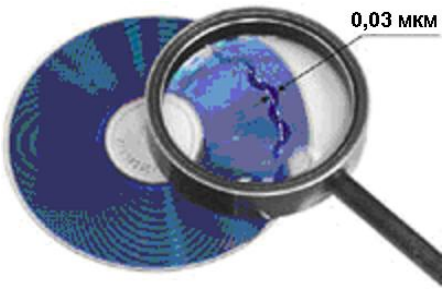


Рисунок 2.18. Разметка на слое-основе CD-R

Частота колебаний канавки относительно спиральной траектории составляет 22,05 кГц (для скорости вращения диска 1x). Соответственно, один период этих колебаний занимает 60 мкм спиральной траектории. Амплитуда колебаний всего 0,03 мкм, значительно меньше ширины самой канавки, но этого достаточно, чтобы выделить колебания с частотой 22,05 кГц и синхронизировать этими колебаниями частоту своего тактового генератора. Кроме того, вся спиральная разметка разбивается на фреймы, каждый из которых по длительности соответствует одному кадру информации. Информация о номере фрейма (будущего кадра) представлена на разметке путем сдвига частоты вобуляции на 1 кГц от значения 22,05 кГц, т.е. реально частота вобуляции принимает значения 21,05 кГц или 23,05 кГц.

Диск CD-RW (CD-ReWritable) используется для многоразовой записи данных, причем можно как просто дописать новую информацию на свободное пространство, так и полностью перезаписать диск новой информацией (после предварительной очистки всего диска). Скорость записи современных накопителей CD-RW составляет 2x-24x.

Дисководы DVD-ROM и DVD±RW

В 1996 году DVD Forum (Hitachi, Matsushita (Panasonic), Mitsubishi, Philips, Pioneer, Sony, Thomson, Time Warner, Toshiba и JVC) принял стандарт на диски DVD-R, предусматривающий два размера: «нормальный» (12 см) и уменьшенный (8 см). Диск с диаметром 8 см предназначен для использования в цифровых малогабаритных видеокамерах. Расшифровывалась аббревиатура DVD как Digital Video Disk – «цифровой видеодиск», однако позже стандарт новых дисков переименовали в Digital Versatile Disk – «цифровой многофункциональный диск». При разработке DVD были учтены проблемы защиты информации на этих дисках (метка региональной зоны на диске, использование специальных методов защиты видео- и аудио-информации).

Для работы с DVD дисками используется лазер с длиной волны 635 нм. Минимальная длина пита уменьшилась в 2 раза по сравнению с CD – до 0,4 мкм, шаг дорожек – до 0,74 мкм.

Емкость диска DVD первого поколения составила 4,7 Гб, и он получил официальное наименование **DVD-5**, стандарт **DVD-9** предусматривает

использование двухслойных дисков. Первый слой может быть получен механической прессовкой, а дополнительный слой нанесен напылением. Дополнительный верхний слой является полупрозрачным и не препятствует чтению нижнего слоя. Диск стандарта **DVD-9** способен хранить до 8,54 Гб данных. Дальнейшим развитием стандартов DVD-5 и DVD-9 стали стандарты на двухсторонние диски **DVD-10** (9,4 Гб) и **DVD-18** (17,08 Гб).

Позже стандарт DVD-дисков был дополнен спецификацией на записываемые и перезаписываемые диски DVD-R и DVD-RW.

Стандарт на DVD-R разбит на три так называемых «слоя» или «уровня».

Первый из них – «физический» («Physical layer»), определяет требования к физическим размерам информационных углублений (пит) и их расположению на диске, размерам самого диска и его основных зон, длине волны лазера в режимах записи и воспроизведения, мощность его излучения и т.д.

Второй, «системный» («File system layer»), описывает файловую систему данных.

Третий – пользовательский («Application layer») определяет формат и параметры цифрового потока, включая управляющие команды переключения режимов кодера и декодера MPEG-2, данные о размещении файлов и доступе к ним и т.д.

Первые два уровня базируются на технологии записи/воспроизведения компьютерных DVD-R/DVD-RW, а третий – на стандарте DVD-Video.

Существуют также диски **DVD-RAM**, которые представляет собой одно- или двусторонний диск, помещенный в пластиковый картридж. Для работы с ними необходим специальный дисковод.

В 1999 году был также разработан формат дисков **DVD+RW**. Разницы в формате представления информации на **DVD+RW** нет. Особенность формата – более высокая точность позиционирования лазерного луча позволяет осуществлять коррекцию данных «на лету», в реальном времени переписывая отдельные сбойные сектора диска, т.е. в **DVD+RW** реализован более совершенный алгоритм коррекции ошибок. В дисках **DVD+R** применяется специальный отражающий слой с повышенной отражающей способностью.

Устройства для работы с DVD-дисками по внешнему виду не отличаются от привода CD-ROM.

Принцип записи информации на DVD-R и DVD-RW тот же, что и для CD-R и CD-RW. Процесс воспроизведения ничем не отличается от

штампованного DVD, с той лишь разницей, что у последнего от поверхности отражается 45 – 85% световой энергии, а у DVD-RW всего 18 – 30%. Поэтому не все ранее выпущенные DVD-приводы могут надежно их считывать.

Дисководы Blu-Ray и HD

В 2002 году представители девяти лидирующих высокотехнологических компаний Sony, Matsushita (Panasonic), Samsung, LG, Philips, Thomson, Hitachi, Sharp и Pioneer на совместной пресс-конференции объявили о создании и продвижении нового формата оптических дисков большой емкости под названием **Blu-Ray Disc**. Согласно объявленной спецификации Blu-Ray Disc (или **BD-R** и **BD-RE**) – перезаписываемый диск следующего поколения со стандартным CD/DVD размером 12 см с максимальной емкостью записи на один слой и одну сторону до 27 Гб.

Формат **HD DVD** был предложен компаниями Toshiba и NEC на сессии DVD Forum в августе 2003 года. В феврале 2008 года стало известно о фактической победе **Blu-Ray** над **HD DVD**: компания Toshiba сообщила о полном сворачивании работ в этом направлении. Производство фильмов и других программ на HD DVD также прекращено.

Назвать Blu-Ray и HD принципиально новыми дисками для записи нельзя – это скорее эволюция формата DVD. В новых устройствах для записи и воспроизведения информации на диск вместо красного лазера (635 нм), который используется в DVD, применен синий лазер (blue-violet laser) с длиной волны 405 нанометров. Меньшая длина волны – соответственно меньшая интерференция отраженного луча, что позволяет сделать толщину дорожки данных у Blu-Ray и HD дисков в два раза меньше, чем у DVD (см. рисунок 2.19).

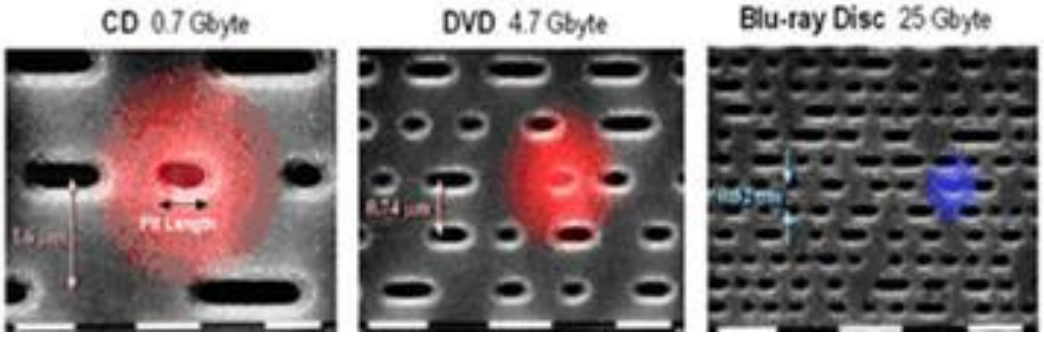


Рисунок 2.19. Поверхность CD, DVD и Blu-Ray дисков

Покрытие дисков Blu-Ray, на которое записываются данные (optical transmittance protection layer), очень тонкое – 0.1 мм. Из этого факта следуют 3 вывода:

1) чем тоньше слой, тем меньше рассеяние отраженного луча и больше данных можно вместить на квадратный дюйм, то есть тонкий слой – это необходимость для достижения большой емкости диска;
2) тонкий слой позволяет без проблем сделать диск многослойным (по крайне мере двухслойным, как DVD), так как уменьшается рефракция луча отраженного от более глубокого слоя;
3) тонкий слой легко повредить, он требует защиты, поэтому у Blu-Ray дисков над активным слоем присутствуют 3 слоя: защитный, покрывающий и особо прочный, как показано на рисунке 2.20 (у HD-DVD такие же слои, как у обычного DVD).

Технологии Blu-Ray и HD создавались в первую очередь для записи, хранения и воспроизведения видео и аудио информации, однако на эти диски можно записать и просто данные. Формат Blu-ray предполагает работу с видеопотоком разрешения до 1080p, звуком вплоть до 7.1 и поддержкой протокола защиты информации HDCP. Поддерживаются алгоритмы кодирования видео – MPEG-2 HD, VC1 (Video Codec 1, базируется на Windows Media Video 9) и H.264/MPEG-4 AVC, форматы звука – AC3, MPEG1, MPEG Layer2. Для цифровых видеоплееров формата Blu-Ray декодирование будет осуществляться аппаратно, для компьютерных приводов – программно.

Blu-Ray устройства имеют высокую скорость пересылки данных. Согласно спецификации максимальная скорость пересылки данных между Blu-Ray приводом и целевым устройством (MPEG-2 декодером или компьютером) может достигать 36 Мбит/с.

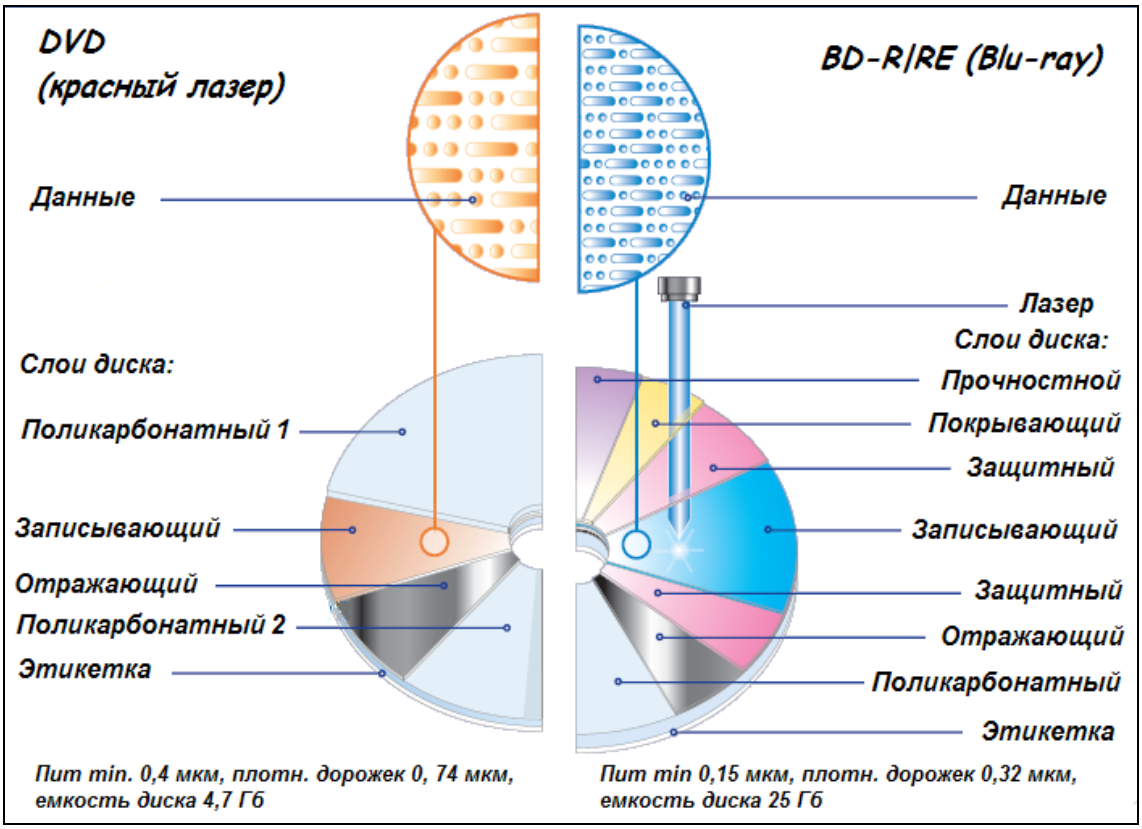


Рисунок 2.20. Диски DVD и BD-R/RE (*www.verbatim.ru*)

На рисунке 2.21 показан внутренний дисковод с интерфейсом SATA, который читает и записывает BD-RE, BD-R и HD DVD диски, а также показаны диски этих типов.

Рисунок 2.21. Привод BRD-SH6B и диски BD-RE и HD DVD-R

В настоящее время выпускаются BD-дисководы, которые читают и записывают также обычные DVD и CD диски, например **DVD±RW/CD-RW/BD/HD LG GGC-H20L**, CD: R40×/W40×/RW24×, HD-DVD: R3×,

DVD: R16×/W16×(4×/RW8× двухслойный), DVD-RAM: R3×/W5×, BD: R6×, Serial ATA 1.0, стоимостью 4600 руб.

В таблицах 2.9 – 2.12 приведены некоторые сравнительные данные по оптическим дискам и дисководам.

Таблица 2.9. Сравнительная характеристика оптических дисков

Показатель	Тип диска		
	CD	DVD	BD
Длина волны лазера, нм	780	635	405
Шаг дорожек, мкм	1,5	0,74	0.32
Минимальный размер пита, мкм	0,88	0,4	0,149
Объем, Гб (один слой)	0,7	4,7	25

Таблица 2.10. Спецификации BD и HD дисков

Показатель	Тип диска							
	BD-R		BD-RE		HD DVD-R		HD DVD-RW	
Количество записываемых слоев	1	2	1	2	1	2	1	2
Объем, Гб	25	50	25	50	15	30	15	30
Записываемый слой	Неорганическое вещество		Вещество, изменяющее агрегатное состояние		Неорганическое вещество		Вещество, изменяющее агрегатное состояние	
Длина волны лазера, нм	405		405		405		405	
Скорость передачи данных, Мбит/с	36		36		36.55		36.55	
Диаметр диска, мм	120		120		120		120	
Толщина диска, мм	1,2		1,2		1,2 (0,6 + 0,6)		1,2 (0,6 + 0,6)	
Покрывающий слой, мм	0,1		0,1		-		-	
Шаг дорожек, мкм	0.32		0.32		0.40		0.40	
Минимальный размер пита, мкм	0.149		0.149		0.204		0.204	

Таблица 2.11. Сравнительная характеристика стоимости оптических дисков разных типов

Марка	Цена, руб.
CD-R LG 700Mb 52× 10 шт.	66
CD-RW LG 700Mb 4-10× 10 шт.	160
DVD-R VERBATIM 4.7Gb 16× 10 шт.	97
DVD-R VS 4.7Gb 16× 10 шт.	74
DVD+RW LG 4.7Gb 4× 10 шт.	205
DVD-RW VS 4.7Gb 4× 10 шт.	137
BD-R Verbatim 25ГБ 4×	227
BD-RE PHILIPS 25Gb 2×	220

Таблица 2.12. Сравнительная характеристика стоимости DVD-дисководов

Тип привода	Марка	Цена, руб.
DVD-ROM	LG DH16NSxx Black	743
DVD-RW	NEC AD-7240S-0B SATA	994
DVD-RW	LG GH22NS40 SATA	1050
BD, HD, DVD-RW, CD-RW	LG GGC-H20L	4200
BD-RW, DVD±RW, CD-RW	LITE-ON IHOS104-06 SATA	2628
BD-RW	NEC BR-5100S-01 SATA	3203

2.5.3 Флэш-память

Флэш-память появилась довольно давно (первые образцы были разработаны компанией Toshiba еще в 1984 г.), однако ее массовое использование началось с широким распространением цифровых фотокамер. Сегодня производители выпускают флэш-память нескольких типов:

– **флэш-карты** Compact Flash (CF), SmartMedia (SM), MultiMedia Card (MMC), SecureDigital (SD), Memory Stick PRO (MS PRO), Memory Stick (MS) и xD-Picture (xD) – для работы с ними необходимо устройство чтения флэш-карт;

– **USB-флэш-память** (так называемые «флэшки»); последняя самодостаточна и не требует применения дополнительных устройств для записи и чтения информации, имеет разъем для подключения к USB-порту ПК (см. рисунок 2.22).

Рисунок 2.22. Apacer AH324 16 Gb

Флэш-память – разновидность ЭСППЗУ, ее полное название Flash Erase EEPROM (Electronically Erasable Programmable ROM) можно перевести как «быстро электрически стираемое программируемое постоянное запоминающее устройство». Другими словами, флэш-память – это энергонезависимая (не потребляющая энергии при хранении данных) перезаписываемая (данные можно стереть и записать заново при помощи электрического тока) память, содержимое которой можно быстро стереть (Flash Erase).

Флэш-память – это полупроводниковая память особого типа. Ее элементарная ячейка, в которой хранится один бит информации, представляет собой не конденсатор, а полевой транзистор со специальной электрически изолированной областью, которую называют «плавающим затвором» (floating gate). Электрический заряд, помещенный в эту область, способен сохраняться многие годы. При записи одного бита данных ячейка заряжается, заряд помещается на плавающий затвор, при стирании он с затвора снимается и ячейка разряжается.

Быстродействие флэш-памяти достаточно сильно варьируется в зависимости от ее типа. Самая быстрая – USB-флэш-память – скорость чтения 8 Мбайт/с, записи 7 Мбайт/с; у некоторых производителей до 20 Мбайт/с. За ней следуют флэш-карты CF (6,4 и 4,2 Мбайт/с), SD (5,8 и 4,9 Мбайт/с), SM и xD (4,0 и 0,7 Мбайт/с), MMC (2,0 и 1,9 Мбайт/с), MS PRO (1,4 и 0,5 Мбайт/с), MS (1,2 и 0,6 Мбайт/с).

Разброс по ценовым характеристикам флэш-памяти меньше, но также существует. Более дешевые – SD и CF-карты, дороже – MS-карты (см. таблицу 2.13).

В настоящее время емкость USB-флэш-память и флэш-карт может доходить до 64 Гбайт.

Таблица 2.13. Карты флэш-памяти и USB-флэш-память

Тип карты	Марка	Цена, руб.
CF	8GB Transcend 266x (TS8GCF133)	970
CF	16GB Transcend (TS16GCF133)	1766
CF	32GB Transcend (TS32GCF133)	2992
SD	2GB Transcend 150x (TS2GSD150)	238
SD	4GB Transcend 150x (TS4GSD150)	381
SD	8GB Transcend (TS8GSDHC6)	646
SD	32GB SanDisk, класс 4	2799
microSD	2GB Transcend (TS2GUSD-S3)	265

Продолжение таблицы 2.13

Тип карты	Марка	Цена, руб.
microSD	8GB Kingston MicroSD SDHC Class 4	732
microSD	16GB Kingston MicroSD SDHC Class 2	2110
MS	4 GB Transcend (TS4GMSM)	756
MS	2GB Transcend (TS2GMSD)	517
USB-флэш-память	2GB Transcend (JetFlashT3W) USB2.0	338
USB-флэш-память	4GB Kingston DataTraveler 112 USB2.0	366
USB-флэш-память	8GB Kingston DataTraveller I, 40X (DTI/8GB)	608
USB-флэш-память	16GB Kingston DataTraveler (DT112/16GB)	1085
USB-флэш-память	32GB Kingston DataTraveller 112 (DT112/32GB)	2540
USB-флэш-память	64 GB Silicon Power "LuxMini 920 200X"	6142

В качестве скоростного и универсального (не требующего применения дополнительных устройств) накопителя для переноса достаточно большого объема данных (на сегодня до 64 Гбайт) удобно использовать USB-флэш-память.

В настоящее время некоторые ПК и ноутбуки комплектуются устройствами чтения флэш-карт. Можно также приобрести переносное устройство – внешний карт-ридер для всех типов карт (например, All-In-One REKi CF/MD/SM/MMC/SD/MS/SMC/TransFlash USB2.0, цена 346 руб.), поэтому флэш-карты получают все большее распространение, тем более что они же используются и в фотоаппаратах, плеерах и других бытовых устройствах.

Преимущества флэш-памяти по сравнению с другими средствами переноса и хранения данных очевидны – высокая надежность и ударопрочность (в результате отсутствия движущихся компонентов и простоты механической конструкции носителей и накопителей), малое энергопотребление и компактность. Однако у нее есть и недостатки: ограниченное число циклов перезаписи (в настоящее время до 1 млн.) и относительно медленная работа.

Последнее обстоятельство связано с принципиальной особенностью работы флэш-памяти NAND-архитектуры: нельзя перезаписать содержимое одной отдельно взятой ее ячейки – можно только стереть содержимое достаточно большого блока ячеек памяти и потом записать туда новую информацию. Изменение содержимого одного бита (байта) данных во флэш-памяти происходит поэтапно: сначала с микросхемы флэш-памяти в буфер считывается блок данных, потом этот блок стирается в микросхеме,

затем в буфере изменяется нужный бит (байт) и блок данных снова записывается в микросхему.

На базе флэш-памяти разработаны массивы хранения данных (например, Sun Storage F5100 Flash Array). Они предназначены для повышения быстродействия баз данных и оптимизации архитектур сетей хранения данных с целью увеличения производительности при снижении затрат. Объем хранения данных в Sun F5100 Flash Array на базе твердотельных флэш-накопителей может достигать двух терабайт. Sun F5100 обладает производительностью 1,6 млн операций чтения и 1,2 млн операций записи в секунду, имеет пропускную способность 12,8 Гбайт/с. Устройство потребляет 300 Вт. По информации Sun, его рабочие параметры сопоставимы с системой хранения данных класса предприятия с 3 тыс. жестких дисков, которые занимают более 14 стоек и потребляют в десять раз больше электроэнергии (40 кВт).

2.5.4 Стример

Стример – устройство для резервного копирования данных (backup) с жестких дисков на магнитные ленты.

Считается, что жесткие диски не обеспечивают должного уровня безопасности длительного хранения информации. Для более надежного архивного хранения данных рекомендуется применять стримеры.

Основной задачей резервного копирования является обеспечение сохранности оперативной информации, представляющей большое значение для любых предприятий. Потери организаций, работающих с банковскими счетами клиентов, от одного часа простоя, вызванного временным сбоем в работе компьютерного оборудования, могут исчисляться миллионами долларов. Потери, вызванные полной утратой коммерческой информации, могут привести к полному краху предприятия.

Ежедневное проведение резервного копирования (а лучше два, три раза в день), хоть и является наилучшим выходом с точки зрения безопасности данных, но для хранения нескольких копий требуется слишком много места на магнитных лентах. Поэтому, кроме полного резервного копирования, применяется инкрементальное и дифференциальное копирование. В этом случае после первоначального полного копирования всей необходимой информации выполняется копирование только новой и измененной информации. Для инкрементального метода полное копирование периодически повторяется.

Для небольших локальных сетей (до 20 компьютеров) возможно применение одного стримера (ленточного привода) со сменными ленточными картриджами. Такие устройства характеризуются относительно небольшими скоростными и емкостными характеристиками (вместительность одного картриджа от 40 до 400 Гб, в зависимости от длины ленты и степени сжатия информации при копировании) и невысокой стоимостью. Скорость чтения/записи таких устройств находится в пределах 6 – 12 Мб/сек. Недостатком таких устройств является необходимость вручную заменять картриджи по мере их заполнения.

Для средних предприятий (до 100 компьютеров в сети) существуют автоматические ленточные библиотеки, способные вмещать до 10 картриджей одновременно. Такая библиотека самостоятельно выбирает необходимый картридж и вставляет его в ленточный привод для дальнейшей работы. Чтение/запись информации производится только с одного картриджа одновременно со скоростью от 21 до 43 Гб/час. Общая емкость загруженной библиотеки составляет до 1.6 Тб данных. Такие устройства могут быть встроены в серверный шкаф, являются полностью автономными и не требуют внимания со стороны администратора.

В ленточные библиотеки более высокого уровня имеется возможность установить сразу несколько приводов для обеспечения необходимой скорости чтения/записи информации. Например, в ленточную библиотеку Scalar100 могут быть установлены 8 ленточных приводов для обеспечения теоретической скорости чтения/записи информации до 560 Мб/сек с общей емкостью хранимой информации до 28.8 Тб. На практике скорость чтения/записи таких систем ограничивается только пропускной способностью существующих каналов связи.

Наиболее крупные распределенные системы (более 250 компьютеров) при объемах информации, подлежащей резервному копированию более 1Тб/сутки используют виртуальные ленточные библиотеки. В таких системах используется схема **D2D2T** (Disk To Disk To Tape), а не Диск – Лента (**D2T** – Disk To Tape). Такая система представляют собой дисковый массив, напрямую подключенный к собственной ленточной системе. И массив жестких дисков и ленточная система объединены в одном корпусе и соединяются между собой быстродействующими каналами связи (обычно SCSI).

В этой системе процесс резервного копирования происходит следующим образом: информация, подлежащая резервному копированию, передается с устройств своего непосредственного хранения на дисковую часть гибридной системы хранения, что происходит достаточно быстро, а,

следовательно, меньше загружает каналы передачи данных. Полученная информация далее переписывается на ленты, не загружая внешние дисковые системы и обеспечивая должный уровень надежности хранения информации.

К наиболее распространенным форматам стримеров среднего класса относятся **DDS, DAT, DLT, SuperDLT, DLT VS80, Ultrium** и пр.

Технология DDS – в накопителях этой группы используется 4-миллиметровая лента формата DDS (Digital Data Storage), часто ошибочно называемого DAT (Digital Audio Tape). Термин «DDS» одобрен DDS Manufacturers Group и относится к исходному формату DAT, допускающему хранение до 4 Гбайт данных со сжатием или 2 Гбайт без сжатия. В новых четырехмиллиметровых накопителях используется стандарт DDS-4, обеспечивающий емкость в 20 Гбайт без сжатия или около 40 Гбайт со сжатием и скорость передачи данных 40 Мбит/с.

Технология DAT развивается с 1989 г. Носитель – картридж величиной чуть меньше аудиокассеты (DAT-картриджи используются и для высококачественной записи музыки, что следует из их названия – Digital Audio Tape). Запись производится в одном направлении по всей ширине ленты. Современный стандарт DAT72 позволяет хранить на одном картридже DDS 36/72[*] ГБ информации и производить чтение/запись со скоростью 20/40[*] МБ/с. Стримеры DAT поддерживают технологию One Button Disaster Recovery (восстановление состояния системы после полной потери содержимого дисков без необходимости форматирования новых дисков, установки ОС и т. п.). Технология DAT является абсолютным чемпионом: свыше половины используемых сейчас в мире накопителей на магнитной ленте – это стримеры того или иного формата DAT.

Технология DLT появилась в 1985 г. – корпорация DEC выпустила стример TK-50, в результате развития которого появился формат Digital Linear Tape. В настоящее время обладателем прав и основным разработчиком технологии DLT является компания Quantum. Носитель – квадратный картридж 10×10 см. Запись производится в двух направлениях последовательно челночным способом (сначала вперед, затем назад, затем опять вперед, и т. д. до полного использования ширины ленты). Современный стандарт DLT8000 позволяет хранить на одном картридже DLT IV 40/80[*] ГБ информации и производить чтение/запись со скоростью 6/12[*] МБ/с. Стримеры DLT8000 могут также читать/писать картриджи предыдущих форматов DLT IIIXT и DLT III. Среднее время доступа к файлу – 50 с.

[*] Вторая цифра – объем сжатой информации

Стримеры могут работать со 100% загрузкой (24 часа в сутки), что позволяет строить на их основе библиотеки магнитных лент.

Технология SuperDLT – позволяет записывать на SDLT-картридж 160/320* ГБ при скорости обмена 42/85* МБ/с. Устройство может читать DLT IV картриджи, записанные на DLT8000. Среднее время доступа к файлу – 70 с. Стример может работать со 100% загрузкой.

Технология DLT VS80 позволяет хранить на одном картридже 320/640* ГБ информации.

Технология Ultrium является первой реализацией стандарта **LTO** (Linear Tape Open), совместного проекта HP, IBM и Seagate. Картридж Ultrium Generation 1 по размеру почти не отличается от DLT. В картридж встроен микрочип, куда записывается информация о записанных файлах и их местонахождении, а также статистика использования картриджа. Ultrium 800 позволяет записывать на картридж 400/800* ГБ при скорости обмена 40/80* МБ/с. Стример может работать со 100% загрузкой.

Итак, максимальная емкость картриджей для разных технологий составляет, Гбайт:

DDS – 40,
DAT – 72,
DLT – 80,
SuperDLT – 320,
DLT VS80 – 640,
Ultrium – 800.

2.6. Устройства ввода информации

К устройствам ввода информации персональных компьютеров относятся прежде всего клавиатура и «мышь», которыми любой пользователь постоянно пользуется. Реже используют сканер, дигитайзер, web-камера и музыкальные устройства ввода.

2.6.1. Клавиатура и мышь

Клавиатура – 101 – 105 клавишная, имеет латинские и русские буквы, цифры и другие символы (переключение латинских и русских букв в Microsoft Windows сочетанием клавиш Левый Alt+Shift или Ctrl+Shift). Мультимедийная клавиатура (см. рисунок 2.23, клавиатура эргономичная) имеет дополнительные клавиши – запуск Интернет-браузера, программы

для работы с электронной почтой, запуск универсального проигрывателя, клавиши регулирования громкости и пр.

Существует множество программ для обучения слепому десятипальцевому методу быстрой печати, например, "СОЛО на клавиатуре" (*http://ergosolo.ru/products/)*. В слепом методе печати каждый из пальцев двух рук используется для нажатие на определенные клавиши (см. рисунок. 2.24).

Рисунок. 2.23. Беспроводной набор Logitech Cordless Desktop Comfort Laser

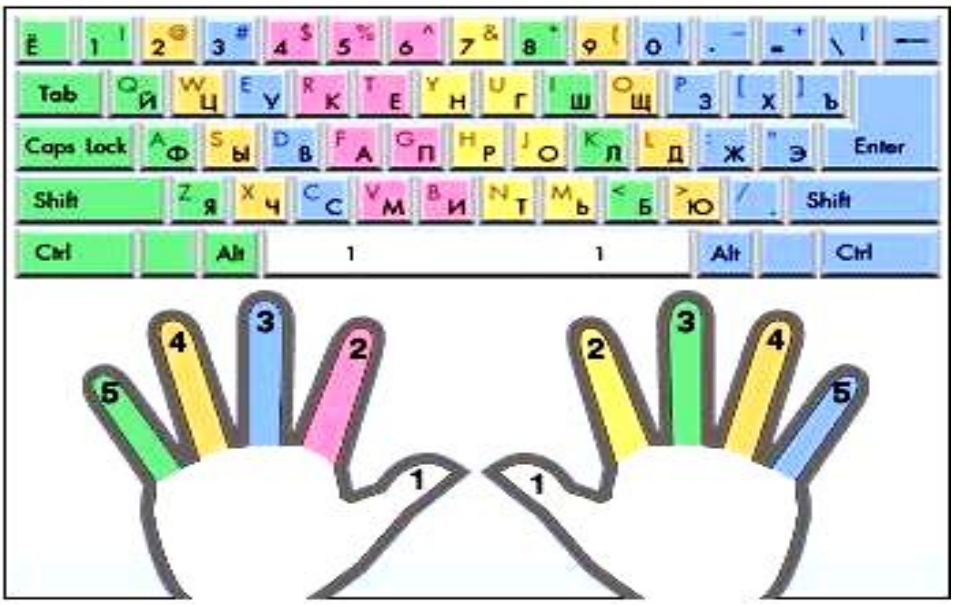


Рисунок. 2.24. Клавиатура ПК и десятипальцевый метод

Манипулятор мышь – инструмент для работы с объектами на рабочем столе операционной системы. Используется для выбора нужного окна, файла, позиции в меню, в тексте, поля на экранной форма и пр., для открытия файлов (двойной или одиночный щелчок левой кнопки), вызова контекстного меню (правая кнопка), прокрутки списка или текста (колесико мышки). В настоящее время используются мыши двух типов – оптические и лазерные, проводные (с подключением к USB-порту) или беспроводные (к USB подключается приемо-передатчик радиосигнала). Главная техническая характеристика мыши – разрешение оптического сенсора, ко-

торое измеряется в точках на дюйм (dpi или cpi – count per inch.) и может составлять 600 – 3200 dpi.

Существуют беспроводные комплекты клавиатура + мышь (см. рисунок. 2.23), такой комплект комплектуется одним приемником радиосигнала. Часто он же служит зарядным устройством для аккумуляторных батареек, которые являются источниками питания для беспроводных устройств.

2.6.2. Сканер

Сканер – устройство для автоматизированного считывания (сканирования) графической информации.

Принцип действия сканера заключается в следующем: установленный в нем источник света облучает сканируемый объект, а оптическая система сканера воспринимает отраженный от объекта световой поток, который с помощью программы сканирования преобразуется в цифровую форму.

Сканеры разделяются на два основных типа:

- настольные,
- ручные.

Существует три разновидности настольных сканеров:

- планшетные
- рулонные
- проекционные.

Основной тип сканера для небольших форматов (А4) в настоящее время – настольный планшетный.

Характеристикой качества сканеров является разрешение, измеряемое числом точек на дюйм – dpi. Хорошие сканеры имеют 1200-2400 dpi. В основном, современные планшетные сканеры цветные с 24 – 36 битным представлением цвета.

В **планшетном сканере** сканирующая каретка – блок, объединяющий в себе источник света и оптическую систему (ширина ее обычно 210-220 мм), перемещается относительно бумаги с помощью шагового двигателя. Оптическая система состоит из одного или нескольких зеркал, линзы (либо призмы) и светочувствительной матрицы. Матрица содержит три линейки светоприемных элементов (по одной линейке на каждый из базовых цветов – красный, зеленый, синий) и оптико-электронный преобразователь. За один такт работы сканера матрица «фотографирует» оче-

редной горизонтальный фрагмент (линию, строку) оригинала. При этом три «разноцветных» элемента матрицы обеспечивают формирование одного пикселя будущего отсканированного изображения. Для сканирования изображения необходимо открыть крышку сканера, положить сканируемый лист на основание изображением вниз (или вверх для прозрачных сканеров HP scanjet 4600, см. рисунок 2.25), после чего опустить крышку. Дальнейшее управление процессом сканирования осуществляется с клавиатуры компьютера – с использованием программы сканирования, поставляемой со сканером. В программе возможна настройка многочисленных параметров сканирования. Конструкция планшетного сканера позволяет сканировать не только отдельные листы, но и страницы толстого журнала или книги.

Рисунок 2.25. Сканер HP scanjet 4600

Для примера в таблице 2.14 приведена характеристики сканера HP scanjet 4600.

В **рулонных сканерах** отдельные листы документов протягиваются вращающимся валиком мимо закрепленной неподвижно сканирующей головки. К рулонному типу относятся широкоформатные сканеры с шириной области сканирования от 635 до 1372 мм, например Contex Cougar SX 25" или CalComp ScanPlus IV 954 C (1372мм).

Проекционные сканеры напоминают собой своеобразный проекционный аппарат. Вводимый документ кладется на поверхность сканирования изображением вверх, блок сканирования находится при этом также наверху. Перемещается только сканирующее устройство. Основной особенностью данных сканеров является возможность сканирования проекций трехмерных предметов.

Для того чтобы ввести в компьютер какой-либо документ при помощи **ручного (handheld)** сканера, надо без резких движений провести его сканирующей головкой по изображению. Равномерность перемещения сканера существенно сказывается на качестве вводимого в компьютер изображения, которое обычно значительно ниже, чем у других типов сканеров.

Для превращения отсканированной текстовой картинки в текстовый документ, состоящий из отдельных символов, служат программы распо-

знавания текста (OCR-системы). Из русских систем наиболее хорошо себя зарекомендовала система FineReader.

Таблица 2.14. Техническая характеристика сканера HP scanjet 4600

Наименование	Характеристика
Типа сканера	планшетный
Режимы ввода	кнопки на лицевой панели: «сканирование», «копирование», «эл. почта», «share-to-web», «запись документа на диск» ПО от HP для создания и обработки фотоизображений
Скорость предварительного просмотра	6 секунд
Скорость сканирования	цветное фотоизображение 10x15 см: 27 секунд черно-белый рисунок: 22 секунды
Разрешение при сканировании	оптическое: 2400 т/д аппаратное: 2400×2400 т/д улучшенное: до 999 999 т/д (зависит от памяти компьютера, свободною места на диске и других параметров системы)
Разрядность	48 бит
Масштабирование	от 10 до 2000% с шагом 1%
Максимальный размер документа	216×297 мм
Интерфейс соединения	1 высокоскоростной интерфейс USB 2.0
Совместимость с операционными системами	Microsoft® Windows® (98, 2000, Me, XP Home и Professional Edition), Mac OS X 9.1 или 10.1 и выше
Габариты	488×340×83 мм
Вес нетто	сканер: 1,4 кг., держатель: 0,5 кг.
Форматы файлов	Bitmap (BMP), TIFF, GIF, PDF, HTML, JPEG, FlashPix (FPX), сжатый TIFF, PCX, PNG, RTF, TXT

2.6.3. Дигитайзер, графический планшет

Дигитайзер – устройство, которое используется для оцифровки конструкторской документации (чертежей).

Графический планшет и интерактивный перьевой дисплей – устройства, которые используются в компьютерной графике и анимации и взаимодействует с графическими приложениями типа Adobe Photoshop.

Дигитайзер может иметь размер рабочей области А1, рабочая область его выглядит, как чертежная доска. Применяется дигитайзер для поточечного координатного ввода графических изображений в системах автоматизированного проектирования (САПР).

Дигитайзер и графический планшет обычно содержит рабочую плоскость, рядом с которой находятся кнопки управления. Для ввода информации служит специальное перо или координатное устройство с «прицелом», подключенное кабелем к планшету или с беспроводной передачей данных. Сам дигитайзер подключается к компьютеру кабелем через USB-порт. Разрешающая способность таких графических планшетов не менее 100 dpi (точек на дюйм).

В некоторых дигитайзерах ввод информации может происходить без специальных перьев или прицелов, так как рабочая поверхность планшета обладает чувствительностью к нажатию пером, основанной на использовании пьезоэлектрического эффекта. При нажатии на точку, расположенную в приделах рабочей поверхности планшета, под которой проложена сетка из тончайших проводников, на пластине пьезоэлектрика возникает разность потенциалов. Координаты этой точки счиываются программой, сканирующей сетку проводников. Эта программа запоминает координаты точек и показывает графические объекты на мониторе.

В графических планшетах и интерактивных мониторах (см. рисунок 2.26) для рисования используется устройство ввода (беспроводное перо), нажатие которого на поверхность рабочей области считывается с использованием технологии электромагнитного резонанса.

Основные задачи, решаемые с использованием этих устройств:

- рисование;
- редактирование фотографий и изображений;
- создание коллажей, открыток и календарей;
- компьютерная анимация и 3D-моделирование;
- работа в мультимедийных приложениях;

- аудио- и видеомонтаж;
- распознавание рукописного текста пр.

Рисунок 2.26. Графический планшет и интерактивный перьевой дисплей

В таблице 2.15 приведены некоторые технические характеристики дигитайзера и планшета.

Таблица 2.15. Техническая характеристика дигитайзера и планшета

Наименование	Дигитайзера CalComp DrawingBoard V 2436	Планшет Wacom Intuos3 A4
Формат	A1	A4
Драйверы	Microsoft Windows 95/98/NT/Me/2000/XP	
Интерфейс	USB, RS-232	USB
Точность	± 0.002 дюйм (± 0.051 мм)	± 0,25 мм
Разрешение	До 500 линий/мм	0,005 мм
Технология	Электромагнитная	Электромагнитный резонанс
Режим работы	Point, line, run, track, incremental, prompt	Программы Adobe Photoshop, Corel GS и др.
Курсор	16-и кнопочный курсор, проводной	8 кнопок и 2 сенсорные полоски

2.6.4. Сенсорный монитор

Сенсорные мониторы отличаются от обычных мониторов тем, что они имеют на поверхности экрана специальное покрытие, которое чувствительно к прикосновениям и передает координаты этого прикосновения через специальный драйвер в операционную систему. Наиболее часто

драйвер эмулирует манипулятор мышь компьютера, либо взаимодействует со специальным программным обеспечением. Сенсорные экраны используются также в мобильных телефонах, КПК, ПК-моноблоках, в интерактивных учебных досках и других устройствах. Сенсорные мониторы в настоящее время наиболее часто можно встретить в терминалах по оплате различных услуг, информационных системах вокзалов, в ресторанах при оформлении заказов официантами и т. п. Они могут использоваться как мониторы обычных настольных ПК. Существуют ПК-моноблоки с сенсорными экранами (например, фирмы HP) с эффектным управлением папками и файлами с помощью руки (похоже на управление тач-фонами).

Существует несколько видов сенсорных технологий:

- резистивная,
- поверхностно-акустическая,
- емкостная,
- инфракрасная.

Резистивная сенсорная панель имеет многослойную структуру, состоящую из двух проводящих поверхностей, разделенных специальным изолирующим составом, распределенным по всей площади активной области экрана. При касании наружного слоя, выполненного из тонкого прозрачного пластика, его внутренняя проводящая поверхность совмещается с проводящим слоем основной стеклянной пластины, играющей роль каркаса конструкции, благодаря чему происходит изменение сопротивления всей системы. Это изменение фиксируется микропроцессором контроллера, передающим координаты точки касания управляющей программе компьютера.

Поверхностно-акустическая волновая технология ПАВ (SAW – Surface Acoustic Wave) основана на использовании сверхзвукового сигнала, вырабатываемого пьезоэлектрическими преобразователями для осей X и Y. Волны направлены перпендикулярно друг другу и образовывают координатную сеть всего поля панели. Прикосновение к экрану поглощает часть волн, проходящих в этот момент через место контакта. Полученные датчиками сигналы отличаются от эталонного, что приводит к автоматическому определению места касания по осям X и Y. Уникальной особенностью является возможность определять силу прикосновения - координату Z.

Емкостная технология основана на использовании сенсорных панелей, покрытыж материалом, содержащим электрический заряд. Когда к панели прикасаются, в точке прикосновения заряд изменяется, что фиксируется измеряющей системой. При работе с емкостными панелями ис-

пользуется прикосновение пальцев, в отличие от резистивных и ПАВ панелей, при работе с которыми могут быть использованы прикосновения пальцем или пером.

Инфракрасная технология используется в основном в экранах больших размеров (школьные интерактивные доски, рекламные демонстрации, большие информационные панели).

Цена настольных 19" сенсорных мониторов от 13000 руб. (www.touchgames.ru).

2.6.5. Музыкальные устройства ввода

Рисунок 2.27. MIDI-клавиатура

К музыкальным устройствам ввода прежде всего относится MIDI-клавиатура, подключаемая к ПК через его звуковую плату. Например, клавиатура M-Audio Pro Keys 88 (www.m-audio.com, рисунок 2.27) – полноразмерное (88 клавиш с профессиональной рояльной механикой) цифровое пианино и MIDI-контроллер. Существуют и другие музыкальные устройства, подключаемые к ПК, например, M-Audio Black Box – мощный эффект-процессор, способный моделировать 12 известных гитарных предусилителей со встроенной drum-машиной и гитарным тюнером.

2.6.6. WEB-камера

Рисунок 2.28. Web-камера

Современная web-камера (рисунок 2.28) представляет собой цифровое устройство, производящее видеосъемку, оцифровку, сжатие, сохранение и передачу цифрового видео по компьютерной сети.

В состав web-камеры входят следующие компоненты: ПЗС-матрица, объектив, оптический фильтр, плата видеозахвата, блок компрессии (сжатия) видеоизображения, центральный процессор и встроенный web-сервер, ОЗУ, флэш-память, сетевой интерфейс, последовательные порты, тревожные входы/выходы.

Некоторые модели камер имеют поворотные устройства и могут выполнять круговое видеонаблюдение, а также имеют варифокальный объектив, которым можно управлять по сети.

Подключаться камеры могут к сетевому коммутатору или концентратору, либо простейшие – к компьютеру (см. рисунок 2.29).

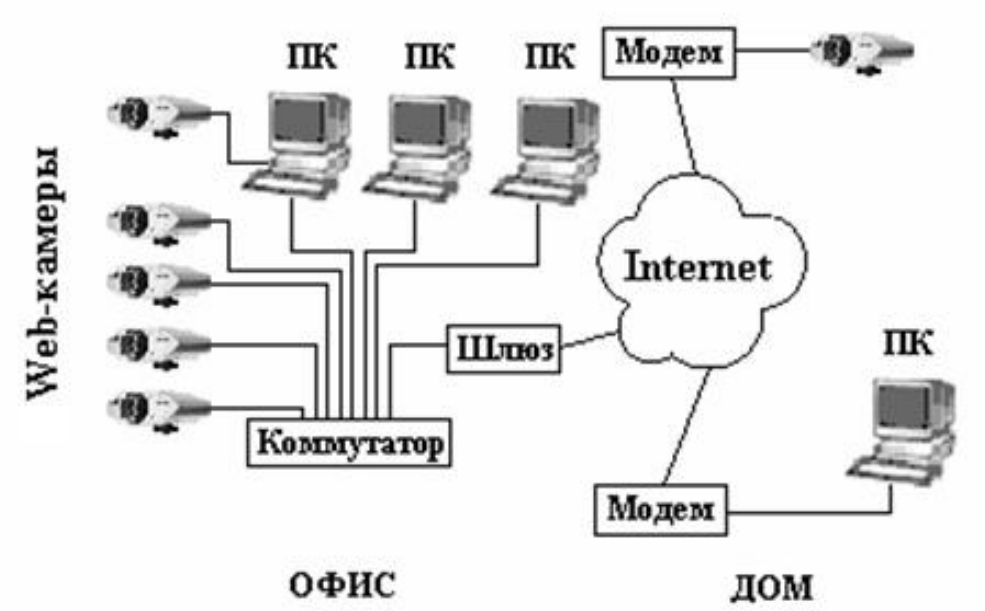


Рисунок 2.29. Варианты подключения Web-камер

2.7. Устройства вывода информации

Предназначены для представления различных видов информации, с которыми работает компьютер, в привычном для человека виде (изображения, текст, числа, звук). К устройствам вывода относятся:

- Видеокарты и мониторы
- Принтеры
- Плоттеры
- Звуковые платы и колонки

2.7.1. Видеоадаптер

Видеоадаптер – устройство для сопряжения ПК с монитором.

Может быть интегрирован с материнской платой или выполнен в виде отдельной платы (видеокарта), подключаемой к слоту PCI Express x16 материнской платы (в старых компьютерах для подключения использовался слот AGP). Возможно использование обоих вариантов при подключении более одного монитора к ПК.

Современные чипсеты и даже процессоры, как указывалось выше, могут иметь высококачественные видеоподсистемы, например ***Intel® Graphics Media Accelerator X4500*** для чипсета Intel® G41 Express и ***Intel® HD Graphics*** для Core i3/i5. Данные видеоподсистемы используют основную оперативную память для своей работы.

Однако, наибольшие возможности для работы с графикой имеют дискретные графические адаптеры интерфейса PCI Express фирм NVIDIA (GeForce) и ATI (Radeon, см. рисунок 2.30), которые имеют графический процессор, использующий специальные технологии обработки графической информации, и собственную видеопамять большого объема.

Рисунок 2.30. Двухпроцессорная видеокарта Radeon HD 5970

Современный видеопроцессор имеет более сложную архитектуру, чем центральный процессор. В нем реализованы потоковые процессоры с раздельными блоками для текстурных и растровых операций. Программы генерации видеоизображения (шейдеры) подразделяются на вершинные, пиксельные и геометрические шейдеры (последние только для DirectX 10 и 11). Вершинные программы-шейдеры отвечают за построение и изменение 3D-объектов, построенных из трехмерных точек (вершин). Пиксельные программы-шейдеры позволяют менять цвета и интенсивность пикселей в результате каких-либо событий в программах. Геометрические шейдеры изменяют параметры трехмерных изображений, позволяют ломать, модифицировать и уничтожать их. Новые стандарты графического интерфейса DirectX 10 и 11 опираются на архитектуру унифицированных (единых) шейдеров, то есть структура кода вершинных, геометрических и пиксельных программ единая, хотя шейдеры выполняют разную работу.

Основные характеристики некоторых видеокарт (с интерфейсом PCI Express) приведены в таблице 2.16.

Таблица 2.16. Технические характеристики некоторых видеокарт

Видеоадаптер	Процессоры, частота, МГц	Память, частота МГц	Тип памяти	Число потоковых процессоров	Объем памяти
GeForce GTS 250	738	1100	GDDR3 256bit	128	512 МБ
GeForce GTX 260	576	1998	GDDR3 448bit	216	896 МБ
GeForce GTX 295	576	1998	GDDR3 448 bit	480	1792 МБ
Radeon HD 4870	750	900	GDDR5 256 bit	800	512 МБ
Radeon HD 5770	850	1200	GDDR5 256 bit	800	1024 МБ
Radeon HD 5970 (двухпроцессорный)	2 по 725	1000	GDDR5 256 bit	2×1600	2×1024 МБ

Производительность видеокарты зависит как от частоты видеопроцессора, так и от количества потоковых процессоров, а также от объема видеопамяти, используемой для хранения текстур и от ширина шины памяти. Современные шины памяти имеют ширину от 128 до 448 бит.

Следует заметить, что стоимость видеокарт с памятью 512 МБ 1500-3000 руб. и выше, цена самых новых видеокарт доходит до 14000 руб.

Современные видеокарты могут обеспечивать работу с разрешением до 2560 на 1600 пикселей, выходы DVI, VGA, HDTV (разрешение HDTV 1920×1080, обычное DVD качество – 720×576).

Видеокарты могут потреблять до 300 Вт электроэнергии и выделять соответствующее количество тепла при этом. Подобное количество тепла, выделяемое с поверхности одного кремниевого чипа, может легко его сжечь кристалл. Охлаждение процессора и памяти видеокарт обычновыполняется с помощью кулера (вентилятор + радиатор).

2.7.2. Монитор

Монитор – устройство отображения графической информации, формируемой видеоадаптером.

В настоящее время основным типом мониторов для ПК является жидкокристаллический (ЖК, LCD – Liquid Crysta Display). До недавнего времени выпускались и использовались ЭЛТ (CRT) мониторы на основе электронно-лучевой трубки (см. рисунок 2.31).

Рисунок 2.31. ЖК- и ЭЛТ-монитор

Кроме того, существуют плазменные панели большого размера (42" – 60"), которые в основном используются в домашних кинотеатрах.

Во всех типах мониторов для формирования цветного изображения используется цветовая схема **RGB** (см. раздел 1.2.3).

Сенсорные мониторы (см. ранее п. 2.6.4) могут быть одновременно устройствами ввода и вывода информации.

Основные характеристики современного монитора следующие.

Размер экрана – обычно измеряется по диагонали в дюймах и составляет у современных ЖК-мониторов 17″, 19″, 20″, 21″, 22″, 24″, 27″, 30″, 32″, 40″, 46″ (стоимость соответственно от 6000 до 88000 руб.).

Разрешение (разрешающая способность) – характеризует способность отображать на экране определенное количество точек по горизонтали и вертикали. Для каждого монитора существует определенное максимально возможное разрешение (которое должно согласовываться с возможным разрешением видеокарты). Обычное разрешение для 17″ – 1024×768, для мониторов больших размеров может использоваться разрешение 1280×1024, 1600×1200, 1920×1200, 2560×1600, 3840×2400. Для ЖК

мониторов существует штатное разрешение, которое соответствует количеству жидкокристаллических пикселей. У мониторов большого размера соотношение сторон может быть 4:3 или 16:10 (см. таблицу 2.17).

Таблица 2.17. Характеристика ЖК-матриц большого размера

Диагональ, дюймы	Разрешение	Соотношение сторон	Шаг точки, мм	Разрешение, dpi
30	2560×1600	16:10	0,258	98,4
24	1920×1200	16:10	0,27	94
23,1	1920×1200	16:10	0,258	98,4
22,2	1920×1200	16:10	0,249	102
22,2	3840×2400	16:10	0,1245	204
21,3	1600×1200	4:3	0,27	94
20,8	2048×1536	4:3	0,207	122,7
20,1	1600×1200	4:3	0,255	99,6
20	1680×1050	16:10	0,258	98,4

Частота кадровой развертки или частота смены кадров, выраженная в герцах (Гц) – сколько в секунду изображение обновляется. Показатель имел большое значение для ЭЛТ-мониторов, где изображение постоянно регенерировалось бегущим по строчкам лучом: чем выше у них была частота кадровой развертки, тем меньше был уровень нежелательного мерцания изображения, и, следовательно, меньше нагрузка на зрение. Поэтому желаемым значением было 80 - 85 Гц. Для ЖК-мониторов используется частота 60 Гц, т. к. изображение у них не мерцает.

Шаг точки – это расстояние по диагонали между двумя пикселями жидких кристаллов или точками люминофора одного цвета. Этот размер обычно выражается в миллиметрах (например, 0,297 мм).

Для ЖК-мониторов в паспортных данных указываются также следующие характеристики:

Яркость – обычно имеет значение 250 – 300 кд/кв. м.

Контрастность 400:1, 550:1, 600:1.

Время отклика пикселя 8 – 16 мс.

Угол обзора: по горизонтали (например, 170°) и по вертикали (170°).

Ориентация экрана у ЖК-монитора в отличие от ЭЛТ-монитора может быть как ландшафтная (горизонтальная), так и портретная (вертикальная). В то время как традиционные экраны ЭЛТ-мониторов и ЖК-экраны ноутбуков имеют только ландшафтную ориентацию, обусловлен-

ную тем, что поле зрения человека в горизонтальном направлении шире, чем в вертикальном, в ряде случаев (работа с текстовыми документами, Web-страницами) удобнее работать с экраном портретной ориентации. ЖК-мониторы некоторых моделей можно развернуть на 90° с сохранением правильной ориентации изображения.

Мониторы, подобно всем электрическим приборам, должны соответствовать требованиям по безопасной эксплуатации, закрепленным в соответствующих стандартах. В паспортных данных мониторов можно найти, например, соответствие их следующим стандартам:

- по энергосбережению: EPA Energy Star;
- электрической безопасности: CE (EU), CUL, SEMKO (Швеция), TUV-GS (Германия), UL (USA), ГОСТ;
- электромагнитной совместимости: C-Tick, FCC Класс B;
- по эргономике: ISO 13406-2, FCC-B, C-TICK, VCCI-2;
- защиты окружающей среды: TCO-99;
- Plug&Play: DDC-2B

Тип подключения ЖК-монитора к видеокарте компьютера может быть цифровой – DVI-интерфейс или аналоговый – VGA-интерфейс (у старых моделей).

ЖК-мониторы, в отличие от ЭЛТ, работают с цифровым сигналом, и использование DVI-интерфейса позволяет избежать двойного преобразования сигнала от компьютера, как это было бы при использовании VGA-интерфейса («цифра» – «аналог» и обратно). Современные графические карты оснащены, как минимум, одним выходом DVI в комбинации с аналоговым разъёмом VGA. Однако, все еще существуют и старые графические карты без DVI, поэтому пока ЖК-мониторы оснащаются и входом VGA.

Первая монохромная ЖК-панель увидела свет в 1970 г. С тех пор технология непрерывно совершенствовалась. В современных профессиональных мониторах применяются активные матрицы на тонкоплёночных транзисторах (AM TFT – active matrix thin film transistor) в различных модификациях. Таким образом, термин LCD (ЖК) более общий, а TFT – одна из разновидностей ЖК мониторов.

Принцип работы ЖК монитора следующий (см. рисунок 2.32).

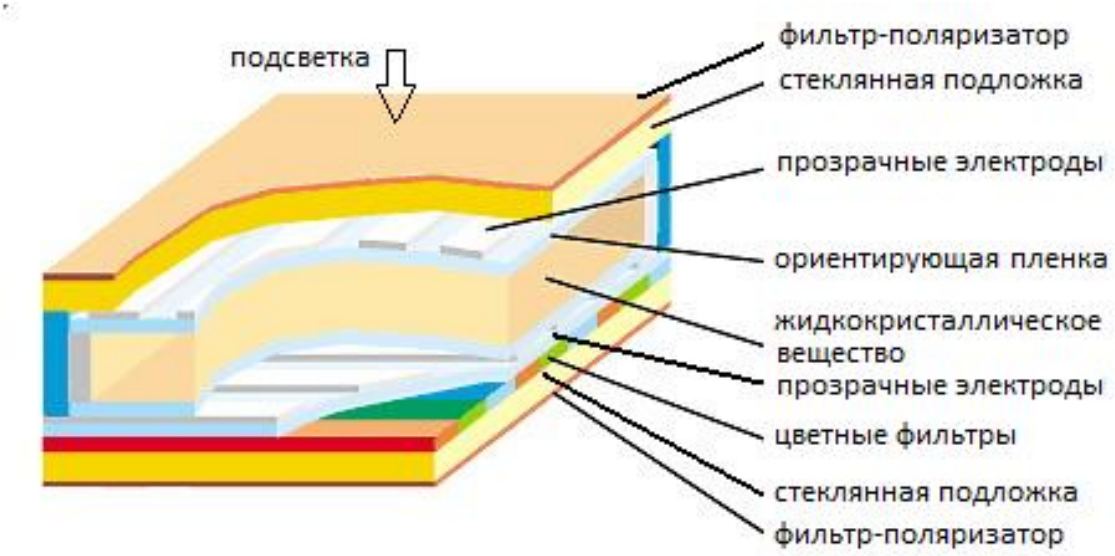


Рисунок 2.32. Устройство жидкокристаллической матрицы

Слой жидкокристаллического вещества (ЖКВ) располагается между двумя полированными прозрачными пластинами, сделанными из свободного от натрия и очень чистого стекла, на которые нанесены прозрачные электроды, управляющие каждой ячейкой структуры TFT. На стеклянных панелях имеются бороздки, которые ориентируют молекулы ЖКВ в определенном направлении. С двух сторон от стекол находятся поляризаторы, оси которых перпендикулярны. Это требуется потому, что плоскость поляризации светового луча поворачивается на 90° при прохождении ЖК-слоя. С внешней стороны экрана располагаются светофильтры, с помощью которых формируются пиксели из трех ячеек основных цветов (RGB). ЖК-панель освещается источником света сзади, то есть работает на прохождение света. Свет от лампы подсветки, пройдя поляризатор и стеклянную пластину, попадает в слой ЖКВ. Если на элементарную ячейку не подано напряжение, то при прохождении света через слой жидкого кристалла его поляризация поворачивается в соответствии с поворотом оптической оси кристалла. В результате на выходе свет пройдет через второй поляризатор, и этот пиксель дисплея выглядит для наблюдателя окрашенным в один из трёх цветов RGB (в зависимости от фильтра), сумма максимальных яркостей трёх сабпикселей дает белый цвет. Если же на электроды пиксельной ячейки подано напряжение, то свет пройдет слой жидкого кристалла, не изменив своей поляризации, и поляризатор на выходе не пропустит его. Этот пиксель будет выглядеть черным на светлом фоне.

Управление каждым сабпикселем выполняется активно (активная матрица), с помощью двух транзисторов и прозрачного электрода. Транзи-

сторы сформированы на прозрачной пленке и выполнены либо из аморфного (a-Si), либо из поликристаллического кремния (p-Si). Поданный уровень напряжения удерживается и сохраняется до следующего сигнала. В принципе, поскольку каждым сабпикселем управляют индивидуально через адресные линии строк и столбцов, необходимости в регулярном обновлении всех элементов изображения нет – достаточно скорректировать управляющие сигналы в тех областях, где изображение изменилось.

2.7.3. Принтер

Принтер – устройство вывода на бумагу текстовой и графической информации.

В настоящее время широкое распространение получили 2 типа принтеров: лазерные – в организациях, для печати большого количества документов ежедневно, и струйные – в домашних условиях с возможностью печати цветных изображений. Ранее использовались также матричные принтеры. Лазерные принтеры обеспечивают более высокое качество печати, чем струйные.

✦ ***Лазерный принтер*** использует фотоэлектронный способ печати, основанный на формировании лазером изображения на заряженной светочувствительной поверхности промежуточного носителя (фотобарабана) в виде электростатического рельефа, притягивающего частицы красителя, которые далее переносятся на бумагу.

Функциональная схема лазерного принтера приведена на рисунке 2.33.

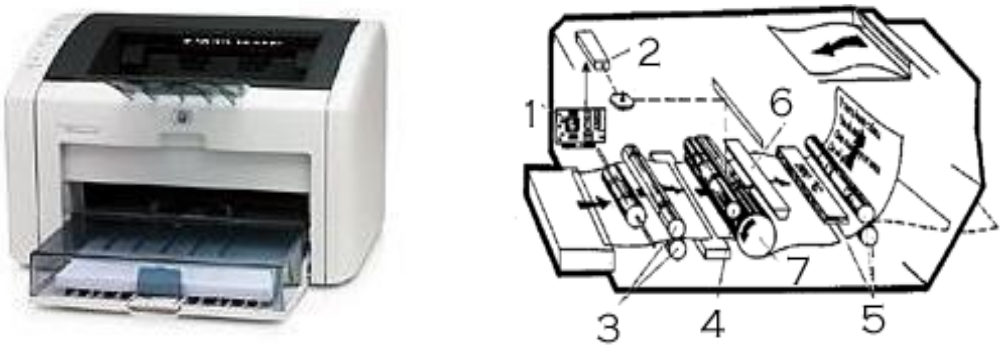


Рисунок 2.33. Лазерный принтер и его схема (1 – процессор, 2 – лазер, 3 – механизм подачи бумаги, 4 – барабан-девелопер, 5 – механизм нагрева бумаги, 6 – лезвие, 7 – фотобарабан)

Наиболее известными фирмами-разработчиками лазерных принтеров являются Hewlett-Packard, Epson, Lexmark, Canon, Toshiba.

Основным элементом конструкции лазерного принтера является вращающийся фотобарабан, служащий промежуточным носителем, с помощью которого производится перенос изображения на бумагу. Барабан представляет собой цилиндр, покрытый тонкой пленкой светочувствительного полупроводника. Обычно в качестве такого полупроводника используется оксид цинка или селен. По поверхности барабана равномерно распределяется статический заряд. Это обеспечивается с помощью тонкой проволоки или сетки, называемой коронирующим проводом. На этот провод подается высокое напряжение, вызывающее возникновение вокруг него светящейся ионизированной области, называемой короной.

Лазер, управляемый процессором принтера, генерирует тонкий световой луч, направляемый с помощью зеркала на фотобарабан. Развертка изображения происходит так же, как и в телевизионном кинескопе: есть движение луча по строке и кадру. С помощью поворачивающегося зеркала луч направляется на фотобарабан, его яркость меняется в соответствии с печатаемым изображением, и в зависимости от интенсивности луча изменяет электрический заряд поверхности фотобарабана. Таким образом, на фотобарабане, промежуточном носителе, возникает скрытая копия изображения в виде электростатического рельефа.

На следующем этапе на фотонаборный барабан наносится тонер – мелкодисперсная краска. Под действием статического заряда эти частицы легко притягиваются к поверхности барабана в точках, подвергшихся экспозиции, и формируют изображение уже в виде красителя.

Бумага втягивается из подающего лотка и с помощью системы валиков перемещается к барабану. Перед самым барабаном бумаге сообщается статический заряд. Затем бумага соприкасается с барабаном и притягивает, благодаря своему заряду, частички тонера нанесенные ранее на барабан. Для фиксации тонера бумага вновь заряжается и пропускается между двумя роликами с температурой около 180^{o}C. После окончания процесса печати барабан полностью разряжается, очищается от прилипших лишних частиц, тем самым готовясь для нового процесса печати.

Цветные лазерные принтеры делятся на однопроходные и четырёхпроходные. В однопроходных все 4 картриджа (цветовая схема **CMYK** – Cyan, Magenta, Yellow, blacK) расположены последовательно по ходу движения бумаги, у каждого картриджа имеется свой фотобарабан и свой источник лазер.

Основные характеристики лазерного принтера следующие.

Качество печати зависит от многих факторов. Основной параметр, по которому можно судить о качестве печати, – это разрешение принтера

(измеряется в точках на дюйм – dpi). Современные лазерные принтеры печатают с разрешением от 600 dpi до 2400 dpi. Помимо физического разрешения существуют технологии, улучшающие качество печати, например, ImageRET у HP, Color RIT у Epson, FinePoint у XEROX и т.д. Рядом с названием технологии пишется число, показывающее, к какому разрешению пытаются приблизиться производители – например, Epson описывает свой принтер, имеющий физическое разрешение 600×600 т/д, как печатающий с разрешением 2400 dpi (с использованием технологии RITech 2400). Конечно, качество изображения от применения таких технологий несколько улучшается, но качество печати все же хуже, чем у принтеров с разрешением 1200 dpi. **Скорость печати** лазерного принтера измеряется в страницах в минуту и для обычных принтеров находится в диапазоне от 10 – 14 страниц в минуту. При печати сложных графических изображений скорость печати лазерного принтера может снижаться. Высокопроизводительные сетевые принтеры обеспечивают скорость печати более 20 страниц в минуту. Скорость печати лазерного принтера зависит от скорости обработки данных, поступающих от ЭВМ и формирования растровой страницы для печати. Как правило, лазерный принтер оснащен собственным процессором. Скорость печати определяется не только работой процессора, но и существенно зависит от объема памяти, которую имеет принтер.

Память лазерного принтера, который обрабатывает информацию постранично, должна обеспечивать большое количество вычислений. Например, при разрешении 300×300 dpi на странице формата А4 насчитывается почти 9 млн. точек, а при разрешении 1200x1200 - более 140 млн. Минимальной величиной памяти лазерного принтера считается 1 Мбайт, а в основном используют память от 2 до 4 Мбайт, причем цветные лазерные принтеры обладают еще большей памятью. Сетевой лазерный принтер может иметь и внешнюю память (винчестер).

Интерфейс подключения самый распространенный – USB. И здесь есть несколько нюансов: существует две версии – USB 1.1 и USB 2.0 (HiSpeed USB). Интерфейс USB 2.0 намного быстрее первой версии, и желательно, чтобы компьютер поддерживал именно последнюю версию. Второй нюанс: для подключения по USB нужен качественный кабель, так как некоторые принтеры выдают при работе со слабеньким кабелем труднолокализуемые ошибки. Старые принтеры подключаются по параллельному интерфейсу (LPT). Многие принтеры (в основном для рабочих групп) поддерживают подключение по сетевому кабелю через Ethernet-карту – в названии таких принтеров присутствует буква «N». Принтеры,

подключаемые через сеть, поставляются с развитыми средствами администрирования. В дополнение к этим распространённым интерфейсам подключения для многих принтеров опционально предлагаются средства беспроводной связи.

Формат бумаги – в основном лазерные принтеры используют для печати бумагу формата А4 и только некоторые модели обеспечивают печать на листах формата А3. Некоторые модели лазерных принтеров используют для работы бумагу в рулоне, выполняют двухстороннюю печать или имеют возможность выборки листов из нескольких лотков и раскладки напечатанных листов по нескольким приемным карманам.

Языки управления принтером и драйверы. Принтер – устройство растровое, и чтобы превратить отправленный на печать документ в образ, понятный принтеру, нужна программа, которая выполнит все преобразования (масштабирование, растрирование и т.д.). Этим занимаются драйверы с помощью языка описания страниц. На вершине иерархии находится язык Postscript – аппаратно независимый язык, гарантирующий, что распечатки, полученные на разных принтерах от разных производителей, будут выглядеть одинаково. Посередине находится многочисленная группа принтеров, печатающих под управлением языка PCL, разработанного компанией HP. В отличие от Postscript'а, драйвер PCL загружает работой как компьютер пользователя, так и процессор принтера, разделяя нагрузку между ними. Ещё одно отличие от Postscript'а заключается в сильной оптимизации страниц, отправленных на печать, в результате точность вывода снижается, но увеличивается скорость печати. Поэтому для точной печати (дизайн, верстка) нужно пользоваться Postscript'ом, а когда важна скорость, можно воспользоваться PCL-драйвером. Самый низ иерархии – это GDI-принтеры, в этом случае всю задачу по формированию образа страницы для принтера берёт на себя компьютер пользователя. Отличительные особенности GDI-принтеров – это небольшой объём памяти, небыстрый процессор и, часто, печать только из Windows. Обычно к принтеру идёт несколько драйверов: Postscript, PCL-драйвер для цветных принтеров, цветные и монохромные версии Postscript и PCL-драйверов.

✦ ***Струйные принтеры*** имеют печатающую головку, перемещающуюся поперек листа бумаги, имеющую набор тонких сопел, через которые выбрасываются чернила. Диаметр сопел составляют десятые доли миллиметра. В основном число сопел в моделях различных изготовителей составляет от 16 до 64. Хранения чернил обеспечивается двумя конструктивными решениями. В одном из них головка принтера объединена с резервуаром (картриджем) для чернил, причем замена резервуара с чернила-

ми одновременно связана с заменой головки. Другое предусматривает использование отдельного картриджа, который через систему капилляров обеспечивает чернилами головку принтера.

В струйных принтерах в основном используют следующие методы нанесения чернил: термическая струйная (пузырьковая) технология и пьезоэлектрический метод.

Термическая струйная печать разработана фирмами Hewlett-Packard и Canon. Успех принтеров этой технологии был обусловлен тем, что они обеспечивали качество печати близкое к лазерным принтерам при значительно меньшей цене. Качество печати таких устройств зависит от размера точки от капельки чернил, а он очень маленький. Конструкция печатающей головки позволяет достичь разрешения до 9600 × 2400 точек на дюйм (Canon PIXMA iP4500).

Hewlett-Packard называет свою струйную технологию *drop-on-demand* (см. рисунок 2.34), а Canon – пузырьковая технология (bubble-jet – рисунок 2.35).

Рисунок 2.34. Принцип технологии *drop-on-demand* фирмы Hewlett-Packard

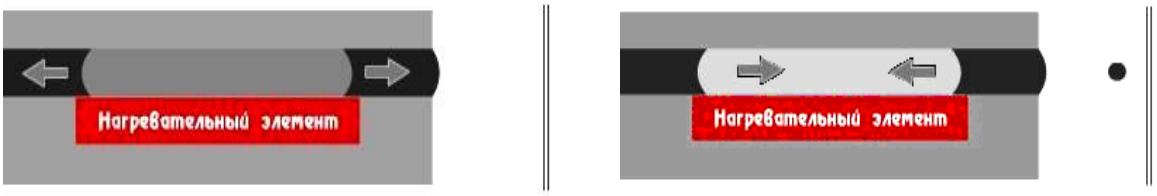


Рисунок 2.35. Принцип пузырьковой технологии (bubble-jet) фирмы Canon

В печатающих системах, использующих струйную пузырьковую технологию, текст и графика получаются при попадании на бумагу капелек чернил, вылетевших из очень тонких сопел.

Принцип работы термического струйного печатающего устройства заключается в следующем.

В стенку сопла печатающей головки встроен нагревательный элемент в виде тонкопленочного резистора. который при пропускании через него тока за 7-10 микросекунд нагревается до высокой температуры. Тем-

пература, необходимая для испарения чернил, например, фирмы Hewlett-Packard, достигает 650°С. Возникающий при резком нагревании чернильный паровой пузырь, стремится вытолкнуть через выходное отверстие сопла каплю жидких чернил диаметром менее 0,16 мм, которая переносится на бумагу. При отключении тока тонкопленочный резистор быстро остывает, паровой пузырь уменьшается в размерах, что приводит к разрежению в сопле, куда и поступает новая порция чернил.

Важной конструктивной особенностью такого печатающего устройства является простая конструкция сопел. Причем кроме низкой стоимости изготовления, такая конструкция устройства имеет ряд других преимуществ:

- высокая надежность каждого сопла;
- сопла можно располагать близко друг к другу, что позволяет получить высокое разрешение при печати;
- отсутствует шум от работы печатающей головки.

Пьезоэлектрический метод, разработанный фирмой Epson, основан на управлении выбрызгиванием чернил с использованием обратного пьезоэффекта, который заключается в деформации пьезокристалла под действием электрического поля. Для реализации этого метода каждое сопло имеет свой плоский пьезокристалл, который является одной из стенок камеры печатающей головки с чернилами.

Под действием электрического напряжения пьезокристалл выгибается и уменьшает объем камеры, под действием избыточного давления жидкие чернила вылетают из сопла в виде капли и образуют на бумаге точку (см. рисунок 2.36).

Рисунок 2.36. Принцип пьезоэлектрической технологии фирмы *Epson*

Пионер пьезоэлектрической технологии – фирма Epson не смогла успешно соревноваться в объеме продаж со своими конкурентами Canon и Hewlett-Packard из-за сравнительно высокой технологической стоимости пьезоэлектрических печатающих головок – они дороже и сложнее, чем пузырьковые печатающие головки.

2.7.4. Плоттер

Плоттер (графопостроитель) – устройство вывода графической информации (чертежей, схем, карт, плакатов) на бумажные и другие листовые носители размером A1 (594×841 мм), A0 (841×1189 мм) и бóльших форматов за счет рулонной подачи бумаги.

До недавнего времени использовались различные типы перьевых плоттеров, однако в настоящее время в основном используются широкоформатные ***струйные плоттеры*** (см. рисунок 2.37) с термической и пьезоэлектрической технологиями печати***, лазерные*** и ***LED-плоттеры***.

Принцип действия лазерных плоттеров аналогичен описанному для принтеров. LED-плоттеры вместо лазера и системы призм используют линейку точечных светоизлучающих полупроводниковых светодиодов (light emitted diod – LED). Общий принцип создания изображения тот же, что и в лазерных принтерах – изображение формируется на заряженном барабане с селеновом или органическим фоточувствительным покрытием, к которому притягивается сухой тонер, который затем переносится на проходящую под барабаном бумагу. LED-плоттеры относятся к классу растровых, каждой точке строки изображения соответствует свой светодиод (например, при разрешении 400 точек на дюйм, линейка для формата A1 (24") состоит из 9600 диодов).

Рисунок 2.37. Плоттер

К плоттерам для различных видов использования предъявляются следующие требования:

- САПР (системы автоматизированного проектирования) – вывод черно-белых и цветных чертежей; воспроизведение тонких линий и мелких объектов; достаточно высокая скорость вывода; возможность печати на бумаге низкого качества;
- ГИС (геоинформационные системы) – высокая точность –максимально допустимая ошибка вычерчивания карт равна 0,2 мм;
- полноцветная широкоформатная печать изображений – возможность печати как на бумаге, так и на синтетических материалах с использованием водных чернил или чернил на органических растворителях (сольвентная печать).

В свою очередь, третий тип плоттеров можно по сферам использования:

- ✦ печать художественных изображений,
- ✦ профессиональная полиграфия,
- ✦ фотопечать,
- ✦ рекламная и оперативная печать.

Основные технические характеристики плоттеров:

- ♦ размер бумаги;
- ♦ разрешение при печати, точек на дюйм;
- ♦ скорость печати, м/ч или $м^2$/ч;
- ♦ память плоттера;
- ♦ наличие винчестера у плоттера и его объем;
- ♦ интерфейс подключения (USB 2.0, IEEE-1284, IEEE-1394a, Ethernet 10/100/1000).

В таблице 2.18 приведены некоторые модели плоттеров различного назначения и их краткая характеристика.

Таблица 2.18. Характеристика различных типов плоттеров

Назначение	Модель (ширина бумаги)	Скорость печати, $м^2$/ч	Максимальное разрешение, dpi	Точность	Технология печати
САПР	HP DesignJet 800 (610 мм)	7,9	2400	±0,2% или 0,25 мм	Термоструйная
САПР	OCE 9555 (904 мм)	65	400		LED
ГИС	Mutoh Falcon RJ-800/801 (914 мм)	3	720	±0,1% или 0,25 мм	Пьезоэлектрическая
ГИС	Mutoh Falcon RJ-4100 (935 мм)	3	1440	±0,1% или 0,25 мм	Пьезоэлектрическая
Плакаты	HP DesignJet Z6100ps (107 см)	105	2400	± 0,1%	Термоструйная
Наружная реклама	Mimaki JV3-160SP (160 см)	30	1440		Пьезоэлектрическая

2.7.5. Мультимедиапроектор

Мультимедиапроектор предназначен для вывода видеоинформации с компьютера на большой экран.

Размер изображения начинается со ста дюймов и практически не имеет предела. Разрешение 1024×768 (или его широкоформатные вариации) является уже стандартным и идеально подходит как для компьютерного сигнала, так и для кино высокой четкости; сами проекторы постоянно бьют рекорды по компактности и яркости (2000 ANSI-лм позволяют работать даже при дневном свете). Благодаря системам коррекции изображения проектор можно установить под любым углом к экрану при сохранении прямоугольного изображения.

Наиболее популярные технологии мультимедийных проекторов на сегодня – **LCD** и **DLP**.

Принципиальное различие технологий заключается в элементах, с помощью которых формируется изображение. В случае **LCD** – это жидкокристаллическая матрица, через которую пропускается свет от источника, в случае **DLP** – матрица микроскопических подвижных зеркал.

LCD-проекторы

В LCD технологии используется свойство жидкокристаллического вещества менять пространственную ориентацию молекул под воздействием электрического поля и оказывать поляризующий эффект на световые лучи. Упрощенная схема LCD-проектора показан на рисунке 2.38. Свет от лампы проходит последовательно через три поляризатора, отсекающие последовательно из всего цветового спектра красный, зеленый и синий цвета. Таким образом, вос-

Рисунок 2.38. Принципиальная схема LCD-проектора

создается цветовая палитра RGB, необходимая при передаче изображения с компьютера или любого другого источника с RGB-выходом.

В современных проекторах используется полисиликоновая технология модуляции изображения. После разделения светового потока на три составляющие, пучки цвета проходят каждый через свою полисиликоновую матрицу, каждая из которых формируют только одну цветовую составляющую изображения. Затем три цветовых компонента изображения – красный, зеленый и синий – смешиваются системой призм (цветосмесительным призматическим блоком). Тот же блок передает уже полноцветное изображение на оптическую систему, которая формирует и фокусирует готовую картинку.

Цена LCD-проектора зависит от яркости светового потока, для 1500 лм составляет около 20 000 руб., для 6000 лм – до 200 000 руб. При этом показатели контрастности от 500:1 до 2500:1.

Пример проектора наименьшей цены: Epson EMP-S42, LCD, SVGA (800x600), 1500 lm, 500:1, 2.6 кг, 20 0 50 руб.; более мощный проектор: Sony VPL-FX52, LCD, XGA, 6000 lm, 1000:1, 10,5 кг, 210 5 36 руб.

DLP-проекторы

DLP технология расшифровывается как Digital Light Processing, то есть цифровая обработка света. Ключевым элементом служит DMD-матрица, уникальная разработка компании Texas Instruments. DMD – Digital Micromirror Device – цифровое микрозеркальное устройство. DMD-матрица состоит из тысяч микроскопических зеркал квадратной формы, с ребром 16 микрометров, которые, в зависимости от наличия или отсутствия сигнала, принимают одно из двух стационарных положений, тем самым либо направляя свет в оптическую систему, либо отклоняя его (см. рисунок 2.39).

По количеству DMD-матриц DLP-проекторы бывают одноматричные и трехматричные.

В первом случае между матрицей и источником света располагается трехцветный светофильтр, своим вращением последо-

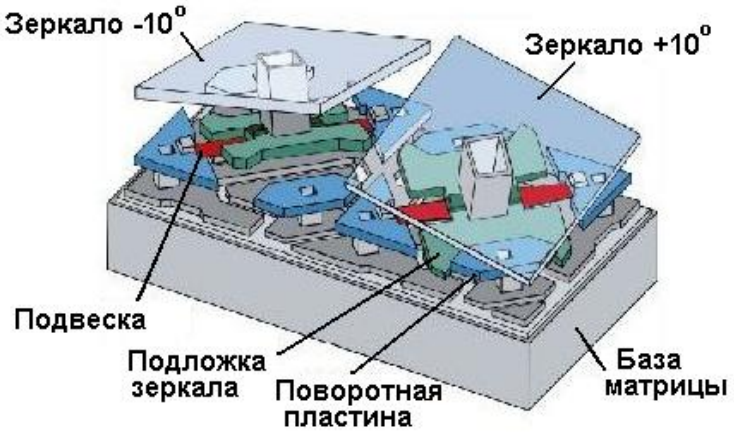


Рисунок 2.39. Зеркала DMD-матрицы

вательно выделяющий цвета палитры RGB. На выходе оптической системы мы получаем последовательно сменяющиеся три составляющие изображения, которые чередуются настолько быстро, что человеческий глаз аппроксимирует их и получает полноцветное яркое изображение.

В случае с трехкристальной системой свет от источника с помощью дихроичных призм расщепляется на три составляющих RGB, после чего каждый пучок попадает на свой кристалл, откуда отражается на оптическую систему синхронизированно с двумя другими. Таким образом, каждая составляющая изображения проецируется отдельной матрицей, что положительно сказывается на качестве получаемой картинки.

DLP-проекторы, пожалуй, самые миниатюрные представители своего класса аппаратуры – есть модели, которые без труда уместятся в дамской сумочке. К недостаткам можно отнести порой заметный «эффект радуги» – наличие радужного ореола вокруг объектов изображения. Также, при длительном просмотре фильмов на однокристальном проекторе некоторые отмечают утомление глаз. По сравнению с аналогичными моделями LCD, этот тип проекторов обладает меньшей четкостью тонких деталей картинки. Однако, при всем этом надо заметить, что более 70% всего рынка профессиональной проекционной техники принадлежит трехкристальным DLP-проекторам.

Цена DLP-проекторов от 18 0 00 руб. при яркости 1800 лм и контрастности 2200:1 (Acer XD1150) до 43 0 00 руб. при ярости 5500 лм, контрастности 7500:1 и разрешении 1920x1080 (Projectiondesign F3).

2.7.6. Устройства вывода звука

Для обеспечения воспроизведения звука на ПК (аудио- и видеофайлов, радио и телевещания Интернета и пр.) необходимо иметь звуковой адаптер и колонки, подключенные к нему (или наушники).

В связи с продвижением ведущими разработчиками аппаратных компонентов ПК (и, прежде всего, фирмой Intel) концепции ПК – домашнего медиацентра и технологии Intel Viiv (программно-аппаратная платформа, позволяющая создавать ПК с расширенными мультимедиавозможностями как компонента цифрового дома**)** современные чипсеты и материнские платы имеют интегрированные аудиоподсистемы с достаточно хорошим качеством воспроизведения многоканального (7.1) звука (High Definition Audio, см. разделы 2.2 и 2.3).

Однако существуют и отдельные платы – аудиоадаптеры, обеспечивающие очень высокое качество воспроизведения звука, подключаемые к

слоту PCI Express, например Auzen X-Fi Bravura 7.1, совместимая с операционными системами Windows 7/Vista/XP (см. рисунок 2.40).

Рисунок 2.40. Звуковая карта Auzen X-Fi Bravura 7.1

Производители этой платы обещают обеспечить «захватывающее» качество музыки и звукового сопровождения в фильмах и играх. В основные возможности звуковой карты входит воспроизведение звука на 7.1-канальной системе с качеством до 24 бит/96 КГц и проигрывание стерео на наушниках с качеством до 24 бит/192 КГц с использованием ЦАП 120 дБ. Специальный усилитель позволяет использовать профессиональные Hi-Fi наушники с разъемом 1/4 дюйма. Разработанное заземление исключает появление помех от блока питания в аудио. Стоимость звуковой карты Auzen X-Fi Bravura 7.1 составляет около 4000 руб.

Стоимость аудиокарт со схемами звука 7.1 может доходить до 7000 руб., самые дорогие карты имеют пульт дистанционного управления.

Количество а***кустических колонок***, подключаемых к аудиовыходам материнской платы или аудиокарты, может соответствовать количеству каналов аудиоподсистемы ПК, или быть меньше. Например, при схеме звука аудиоподсистемы 7.1 можно подключить 7 колонок плюс 1 сабвуфер, но можно подключать и меньшее их количество (5.1, 4.1, 2.1). ***Сабвуфер*** – звуковая колонка для воспроизведения низкочастотных звуков, диапазон воспроизводимых частот от 25 до 150 – 200 Гц.

Первые звуковые карты ПК могли обеспечить вывод двухканального звука (стереозвук). Сейчас это соответствует звуковой схеме 2.0 (первая цифра – количество колонок, вторая – количество сабвуферов). Первые многоканальные акустические системы имели схему 4.0, в состав их входили 4 колонки – две фронтальные и две тыловые. В акустике 4.1 был добавлен сабвуфер. Правда, эти системы все равно остались четырехканальными – низкочастотные сигнала в них выделяются с помощью специального кроссовера.

Следующий тип акустических систем уже обладает полноценным 6-канальным звуком – это акустика 5.1 (рисунок 2.41). В состав этих ком-

плектов входят две фронтальные лонки, две тыловые, одна центральная и сабвуфер. То есть по сравнению с акустикой 4.1 появился центральный излучатель. Он нужен для соответствия формату Dolby Digital, часто используемом в фильмах, особенно на DVD-дисках. По центральной колонке передаются диалоги действующих лиц. Кроме того, акустика 5.1 может оснащаться декодерами DTS и Dolby Pro Logic. Таким образом, системы 5.1 – минимально необходимые для домашнего кинотеатра. На рисунке 2.42 показана схема расположения колонок для звуковой схемы 5.1.

Рисунок 2.41. Колонки звуковой схемы 5.1

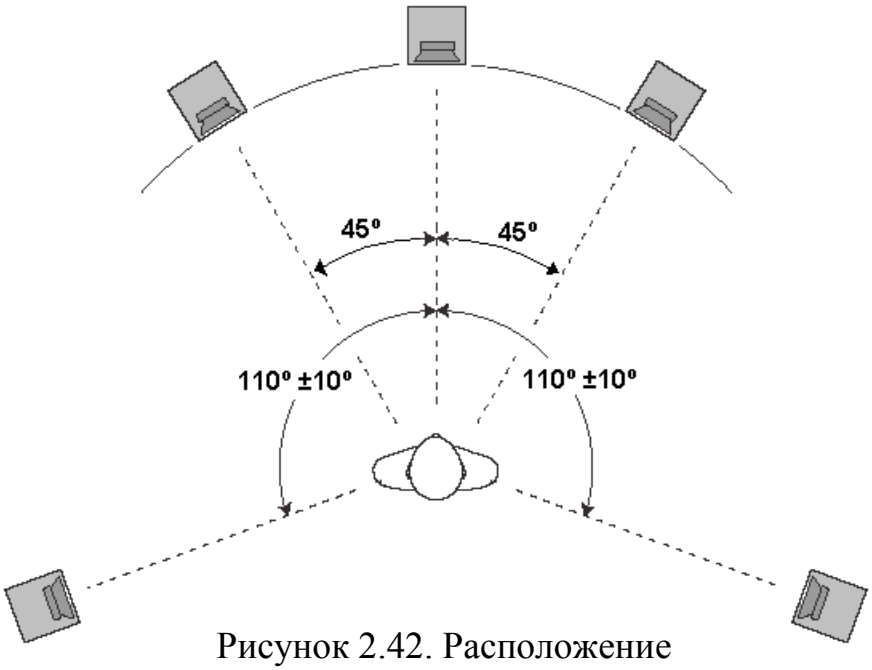


Рисунок 2.42. Расположение колонок звуковой схемы 5.1

Последняя разработка в области звуковых схем – 8-канальные 7.1 и 7.2. В этой акустике добавились еще два динамика – тыловые центральные. Кроме того, в системе 7.2 появился дополнительный сабвуфер, вот только «басовый» канал остался единым, так что особого эффекта эта прибавка не дает. Если вы решитесь приобрести подобную акустику, то сможете дома насладиться звуком формата Doulby digital Surround EX или

DTS Surround EX, который можно найти только в самых современных кинотеатрах.

На рисунке 2.43 показана схема расположения колонок для звуковой схемы 7.1.

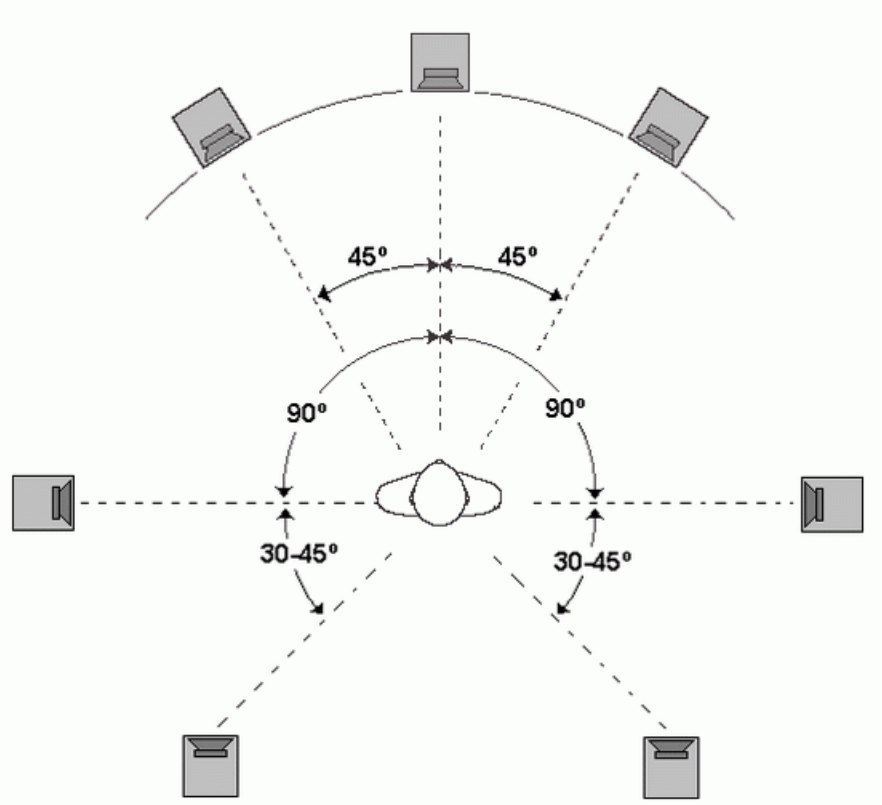


Рисунок 2.43. Расположение колонок звуковой схемы 7.1

Сокращения, используемые для обозначения различных колонок акустической системы:

- LF, RF – левая и правая фронтальные колонки;
- LS, RS – левая и правая тыловые колонки;
- C – центральная колонка (системы 5.1, 7.1 и 7.2);
- SW – сабвуфер;
- LRS, RRS – левая и правая тыловые центральные колонки (системы 7.1 и 7.2).

2.8 Оборудование компьютерных сетей

Локальная сеть (ЛС, LAN) обеспечивает передачу информации между ПК и другими сетевыми устройствами, обычно находящимися в одном или нескольких близко расположенных зданий.

Территориально-распределенная сеть (WAN) соединяет несколько локальных сетей, географически удаленных друг от друга.

Основные задачи, которые позволяет решать ЛС:

- использование автоматизированных систем управления предприятием и создание единой информационной среды предприятия;
- надежное хранение больших объемов информации на сервере с регулярным ее резервным копированием;
- совместное использования в организациях общих ресурсов, таких как жесткие диски, принтеры, накопители CD-ROM, серверные приложения (серверы баз данных, почтовые серверы, Интернет-серверы и пр.).

Для организации территориально-распределенных сетей используется коммутируемая телефонная сеть общего пользования (PSTN, Public Switched Telephone Network) с соединением через модем, линии высокоскоростной цифровой сети с предоставлением комплексных услуг (ISDN, Integrated Services Digital Network), оптоволоконная связь, спутниковые каналы связи.

При постоянном использовании организацией линии территориально-распределенной сети (прежде всего, для связи с Интернет-провайдером), используют выделенную телефонную линию с аналоговой или цифровой передачей данных. За выделенную линию вносится фиксированная плата. Коммутируемое соединение или ISDN предусматривают поминутную оплату.

Стандарты в области локальных сетей объединяет семейство IEEE 802.××, а также ISO 8802-1...5. Эти стандарты были созданы на основе распространенных фирменных стандартов сетей Ethernet, ArcNet и Token Ring. В частности, стандарт IEEE 802.3 описывает модификации сетей Ethernet – 10Base-5, 10Base-2, 10Base-T, 10Base-F, раздел стандарта IEEE 802.3u описывает технологию Fast Ethernet (100Base-T), IEEE 802.3ak стандартизует одну их технологий гигабитных сетей – 10GBase-CX4.

Для организации работы локальных сетей может использоваться следующее основное оборудование:

- интегрированные на материнской плате сетевые адаптеры или сетевые интерфейсные платы;
- концентраторы;
- коммутаторы;
- кабели;
- маршрутизаторы (территориально-распределенные сети);
- модемы (территориально-распределенные сети).

Сетевые адаптеры

Назначение сетевого адаптера – обеспечение работы ПК в локальной сети. Для подключения к сетевому кабелю сетевая карта имеет соответствующий разъем.

Материнские платы современных ПК обычно имеют интегрированную гигабитную сетевую подсистему (10/100/1000 Мбит/с) технологии Ethernet (см. п. 2.3).

Однако выпускаются и отдельные сетевые интерфейсные платы (NIC, Network Interface Card), подключаемые к слоту PCI или USB.

Концентраторы

В локальной сети технологии Ethernet все входящие в сеть ПК взаимодействуют с концентраторами (HUB) или коммутаторами. Соединенные с одним концентратором ПК образуют сегмент локальной сети. Концентраторы бывают разных видов и обычно обеспечивают соединение 12 – 24-х пользователей в пределах одного помещения или группы соседних помещений.

При применении концентратора все пользователи делят между собой полосу пропускания сети. Пакет, принимаемый по одному из портов концентратора, рассылается во все другие порты, которые анализируют этот пакет (предназначен он для них или нет).

В настоящее время вместо концентраторов используются неуправляемые коммутаторы (hub-switch, см. рисунок 2.44).

Коммутаторы

Рисунок 2.44. 48-портовый неуправляемый коммутатор

Коммутатор предоставляет каждому устройству (серверу, ПК или другому коммутатору/концентратору), подключенному к одному из его портов, всю полосу пропускания сети. Это повышает производительность и уменьшает время отклика сети за счет сокращения числа пользователей на сегмент. Новые коммутаторы обычно поддерживают скорости передачи 10/100/1000 Мбит/с в зависимости от максимальной скорости подключаемого устройства и могут автоматически настраиваться на оптимальную скорость.

В отличие от концентраторов, осуществляющих широковещательную рассылку всех пакетов, принимаемых по любому из портов, коммутаторы передают пакеты только целевому устройству (адресату), так как знают MAC-адрес (Media Access Control) каждого подключенного устройства. В результате уменьшается трафик и повышается общая пропускная способность, а эти два фактора являются критическими с учетом растущих требований к полосе пропускания сети современных сложных бизнес-приложений.

Пример описания неуправляемого коммутатора 8 D-Link DGS-1008D (стоимость около 1500 руб.):

коммутатор с 8 медными портами Gigabit Ethernet. Стандарты: IEEE 802.3 10BASE-T Ethernet, IEEE 802.3u 100BASE-TX Fast Ethernet, IEEE 802.3ab 1000BASE-T Gigabit Ethernet, ANSI/IEEE 802.3 NWay определение скорости и режима работы, IEEE 802.3x управление потоком. Количество портов: 8 портов 10BASE-T/100BASE-TX/1000BASE-T. Полный/полудуплекс для скоростей 10/100Мбит/с, полный дуплекс для скорости Gigabit Ethernet. Метод коммутации: Store-and-forward. Размер таблицы MAC-адресов: 8K записей на устройство. Изучение MAC -адресов: автоматическое обновление. Буфер RAM: 256K на устройство. Размеры: 235×161.9×35.6 мм.

Кабели

В наиболее распространенной в настоящее время сетевой технологии ***Ethernet*** сетевой кабель соединяет между собой сетевой адаптер ПК и концентратор или неуправляемый коммутатор (switch). Для подключения на концах кабеля обжимаются коннекторы (наиболее часто – восьми контактные **RJ-45**). Существует несколько типов кабелей, применяемых в различных сетевых технологиях.

Витая пара (TP, Twisted Pair) – основной тип кабеля в настоящее время. Бывает двух видов: неэкранированная витая пара (UTP, Unshielded Twisted Pair, см. рисунок 2.45) и экранированная витая пара (STP, Shielded Twisted Pair). Оба типа кабеля состоят из нескольких пар скрученных изолированных медных проводов. Кабель типа неэкранированная витая пара стал наиболее популярным благодаря своей низкой стоимости, гибкости и простоте установки. Недостатком

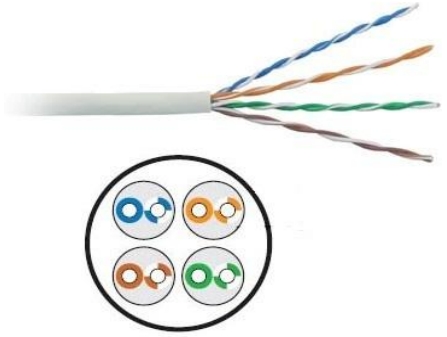


Рисунок 2.45. Кабель UTP, категория 5, 4 пары

такого кабеля является уязвимость к электрическим помехам и шумам в линии. Кабели витая пара бывают разной категории (3, 4, 5 или 6). Чем выше номер категории, тем большую скорость передачи поддерживает кабель.

Оптоволоконный кабель применяется в центральных магистральных сетях и связи между коммутаторами локальных сетей, находящимися на значительном расстоянии один от другого. Обеспечивает полную защиту от электрических помех, позволяет передавать информацию на большие расстояния. Поддерживает скорость передачи данных 10, 100, 1000 Мбит/с, 10 Гбит/с. Данные передаются с помощью световых импульсов, проходящих по оптическому волокну. Недостатки – сложность монтажа и высокая стоимость.

Тонкий и толстый коаксиальный кабель – в настоящее время практически не используется. Конструкция его аналогична стандартному телевизионному кабелю.

Каждая технология ЛС (10-Мбит/с Ethernet, 100-Мбит/с Fast Ethernet или Gigabit Ethernet, 10-Gigabit Ethernet) предполагает использование одного или нескольких типов сетевого кабеля. Наиболее часто применяется кабель UTP категории 5 и 5e (см. таблицу 2.19).

Таблица 2.19. Использование различных типов сетевого кабеля

Сетевой стандарт	Кабель	Число пар проводов
10BASE-T (Ethernet)	UTP, категории 3/4/5	2
100BASE-TX (Fast Ethernet)	UTP, категория 5	2
100BASE-T4 (Fast Ethernet)	UTP, категории 3/4/5	4
100BASE-FX (Fast Ethernet)	Оптоволокно	-
1000BASE-T (Gigabit Ethernet)	UTP, категория 5	4
1000BASE-FX (Gigabit Ethernet)	Оптоволокно	-
10GBase-CX4 (10 Gigabit Ethernet)	CX4	8
10GBase-LX4 (10 Gigabit Ethernet)	Оптоволокно OM3	-

Маршрутизаторы

Маршрутизаторы могут выполнять следующие функции:

- Подключение локальных сетей (LAN) к территориально-распределенным сетям (WAN).

• Соединение нескольких локальных сетей.

Производительность маршрутизатора (объем передаваемых данных в секунду) обычно пропорциональна его стоимости. Поскольку маршрутизатор работает на основе протокола, он может принимать решение о наилучшем маршруте доставки данных, руководствуясь такими факторами, как стоимость, скорость доставки и т.д. Кроме того, маршрутизаторы позволяют эффективно управлять трафиком широковещательной рассылки, обеспечивая передачу данных только в нужные порты.

Модемы

Модемы позволяют пользователям ПК обмениваться информацией и подключаться к Internet по обычным телефонным линиям. Название «модем» означает «модулятор/демодулятор». Модем кодирует цифровые сигналы, поступающие от ПК, в аналоговые или цифровые сигналы, передаваемые по телефонной сети общего пользования, а модем на приемном конце линии демодулирует эти сигналы, снова преобразуя их в первоначальную цифровую форму.

Модем поддерживает в каждый момент только одно соединение. Для ПК применяются встроенные и внешние модемы, а для портативных компьютеров обычно используются модемы формата PC Card.

Модемы используют специальные протоколы модуляции при передаче информации, коррекции ошибок и сжатия данных.

Протоколы передачи данных для модемов установлены Международным институтом телекоммуникаций ITU (International Telecommunication Union).

Основными характеристиками протоколов передачи данных являются скорость передачи и режим: дуплекс или полудуплекс. В полнодуплексном режиме устройство может одновременно передавать и принимать данные.

Аналоговые модемы голосового диапазона используют протоколы V.42, V.42bis, V.92, позволяющие эффективно выполнять коррекцию ошибок и сжатие данных. Скорость передачи информации для протокола V.92 – 56 Кбит/с.

Модемы DSL-технологии (Digital Subscriber Line – цифровая абонентская лини) позволяют организовать цифровую передачу данных по телефонной линии, одновременно та же линия может использоваться и для обычных телефонных разговоров. В последнее время широкое распространение получило безлимитное подключение к Интернету по технологии ADSL (Asymmetric Digital Subscriber Line – асимметричная цифровая

абонентская линия) с использованием ADSL-модемов. Соединение является некоммутируемым, т. е. для подключения к провайдеру достаточно включить настроенный модем и компьютер. Технология ADSL обеспечивает скорость нисходящего потока данных (от абонента к провайдеру) в пределах от 1,5 Мбит/с до 8 Мбит/с и скорость восходящего потока от 640 Кбит/с до 1,5 Мбит/с. Скорость обычно ограничивается провайдером в зависимости от суммы ежемесячной абонентской оплаты за соединение.

2.9 Оборудование беспроводных сетей

Беспроводное соединение компьютеров и периферийного оборудования получает все большее распространение.

Беспроводные локальные сети (Wi-Fi) позволяют организовать связь между ПК там, где затруднено кабельное соединение или необходима полная мобильность. При этом они совместимы с проводными сетями. Необходимо учитывать возможность создания беспроводных соединений ПК при проектировании любых сетей – от малого офиса до предприятия.

В настоящее время на рынке сетевого оборудования Wi-Fi представляет три стандарта: IEEE 802.11a, IEEE 802.11b и IEEE 802.11g.

Стандарт IEEE 802.11a позволяет передавать данные на скорости до 108 Мбит/с. Эти беспроводные сети работают на частоте 5 Ггц и обеспечивают возможность шифрования с использованием WEP.

Стандарт IEEE 802.11b позволяет передавать данные на скорости до 11 Мбит/с и работает на частоте 2,4 Ггц по протоколу широкополосной передачи данных - DSSS.

Стандарт IEEE 802.11g использует скорость передачи до 54 Мбит/с и работает на частоте 2.4 Ггц – той же, что и устройства стандарта IEEE 802.11b, что обеспечивает совместимость этих стандартов.

Оборудование беспроводных сетей включает сетевые адаптеры, точки доступа (Access Point), беспроводные маршрутизаторы (см. рисунок 2.46).

В настоящее время Wi-Fi адаптерами оснащаются все ноутбуки и другие мобильные устройства (планшетные компьютеры, КПК, коммуникаторы).

Wi-Fi адаптеры для настольных ПК выпускаются для подключения к PCI-слоту или USB-порту. Существуют также материнские платы настольных ПК с интегрированными Wi-Fi адаптерами.

Рисунок 2.46. Локальная сеть с использованием Wi-Fi

Стоимость Wi-Fi адаптеров (1000 – 2000 руб.) несколько выше, чем проводных адаптеров (200 – 300 руб.). Точка беспроводного доступа стоит 1500 – 3000 руб.

2.10. Дополнительное оборудование

Прежде всего это фильтры питания, которыми, как минимум, должны быть оснащены все компьютеры – рабочие станции.

На серверах и на тех компьютерах, где предъявляются высокие требования к сохранности информации, необходимо использовать источники бесперебойного питания (ИБП, UPS).

Назначение фильтра питания (например, Power Cube, стоимостью около 200 руб., см. рисунок 2.47) – защита от короткого замыкания, защита от импульсных помех; энергия скачка до 90 Дж; сила входного тока: до 10 А.

Рисунок 2.47. Фильтр питания

Классификация ИБП производится обычно по двум базовым показателям – по его мощности и типу.

ИБП малой мощности подключаются непосредственно к защищаемому оборудованию и питаются от электрической сети через штепсельные розетки. Данные устройства изготавливаются в настольном, реже напольном исполнении, а также

для монтажа в стойку и выпускаются в диапазоне мощностей от 250 до 3000 ВА.

ИБП средней мощности имеют встроенный блок розеток для подключения защищаемого оборудования, либо подключаются к групповой розеточной сети, выделенной для питания защищаемых электроприемников. К питающей сети эти ИБП подсоединяются кабелем от распределительного щита через защитно-коммутационный аппарат. Они рассчитаны на установку как в специально приспособленных электромашинных помещениях, так и в технологических комнатах, где размещается инфокоммуникационное оборудование и допускается постоянное присутствие персонала. Типичный диапазон их мощностей – от 3 до 40 кВА.

ИБП большой мощности подключаются к питающей сети с помощью кабеля от распределительного щита через защитно-коммутационный аппарат, а к защищаемому оборудованию – через выделенную групповую розеточную сеть. Они имеют напольное исполнение для размещения в специально приспособленных электромашинных помещениях. Типичный диапазон мощностей охватывает значения от 10 кВА до нескольких сотен (на рынке имеются модели мощностью до 800 кВА). Параллельные системы ИБП и энергетические массивы могут иметь мощности до нескольких тысяч киловольт-ампер.

По принципу устройства ИБП можно отнести к двум основным типам:

- источники бесперебойного питания пассивного типа с режимом работы «вне линии» (off-line). Принцип действия заключается в питании нагрузки (ПК) от энергосети и быстром переключении на внутреннюю резервную схему при отключении питания или выходе напряжения за пределы допустимого диапазона. Время переключения обычно составляет около 4–12 мс, что вполне достаточно для большинства технических средств инфокоммуникаций с импульсными блоками питания;
- источники бесперебойного питания активного типа с режимом работы «на линии» (on-line). Питание нагрузки осуществляется источником, поэтому время переключения равно нулю. Наряду с резервированием электроснабжения, устройства способны обеспечить необходимое качество электроэнергии при его нарушениях в питающей сети и фильтрацию привносимых помех.

Контрольные вопросы к главе 2

1) Каково назначение персонального компьютера?
2) Какое устройство ПК выполняет обработку информации?
3) От чего зависит производительность процессора?
4) Как выглядит эволюционный ряд процессоров Intel?
5) Назовите марки наиболее производительных процессоров
6) Что такое чипсет?
7) Чем отличаются чипсеты для процессоров Core i5, Core i7 и Core 2 Duo?
8) Какие интегрированные подсистемы может иметь чипсет?
9) Что такое материнская (или системная) плата?
10) Какие большие микросхемы размещены на материнской плате?
11) Какие задачи решает микросхема BIOS?
12) Перечислите слоты, внутренние и внешние порты современной материнской платы.
13) Каково назначение основной оперативной памяти?
14) В чем особенность работы динамической памяти?
15) Какие типы устройств хранения информации Вы знаете?
16) Какой принцип записи информации используется в винчестерах?
17) В чем особенность записи информации на оптические диски?
18) Какой объем информации можно записать на CD, DVD и BD диски?
19) Какой принцип записи информации использует флэш-память?
20) Что такое стример?
21) Какой тип сканеров наиболее распространен?
22) Что такое дигитайзер?
23) Какие основные характеристики имеют современные видеокарты?
24) Опишите принцип работы жидкокристаллического монитора.
25) Какие основные характеристики имеют современные мониторы?
26) Какой принцип формирования изображения использует лазерный принтер?
27) Какой принцип формирования изображения использует струйный принтер?
28) Какие устройства для работы со звуком используются на ПК? Какие звуковые схемы используются?
29) Какое оборудование используется для работы локальных сетей по технологии Ethernet?
30) Что такие ИБП (UPS)?

Глава 3. Программное обеспечение

Все программное обеспечение (ПО, software) по назначению и использованию можно разделить на 2 большие группы – системное и прикладное.

Системное ПО – совокупность программ для обеспечения работы компьютера.

Прикладное ПО – комплекс программ для решения задач определённого класса конкретной предметной области.

В то же время, в состав основных системных средств – операционных систем (ОС) – входят и компоненты прикладного ПО, например, простейшие текстовые редакторы, программы для работы с мультимедиа, с Интернетом и другие. В составе прикладного ПО можно выделить особую группу – средства разработки ПО, назначение которых – разработка новых программ, в том числе и операционных систем. Так что эти 2 большие группы имеют некоторое пересечение по своим компонентам. Многие другие группы ПО также имеют общие компоненты, например система автоматизации управления предприятиями работает с системой управления базой данных (СУБД), имеет средства разработки, может обмениваться информацией с САПР, и т. п. На рисунке 3.1 показана классификация ПО, которая учитывает наличие общих компонентов в различных видах ПО и взаимодействие различных видов ПО.

Рисунок. 3.1. Классификация программного обеспечения

Без системного ПО не может работать ни один компьютер. Наиболее распространенной операционной системой в настоящее время является **Microsoft Windows**. Драйверы (программы, обеспечивающие работу ОС со всеми компонентам ПК) могут присутствовать в составе ОС, но для самых новых материнских плат и внешних устройств поставляются отдельно. В сервисное ПО входят программы проверки дисков и файловой системы, антивирусные программы, архиваторы, программы записи оптических дисков, программы обслуживания компьютерных сетей и пр.

Среди прикладного ПО особое место по сложности разработки, внедрения и эксплуатации, по количеству компонентов системы, по объему обрабатываемой информации занимают системы комплексной автоматизации управления предприятиями, на верхнем уровне которых – ERP-системы (ERP – Enterprise Resource Planning). В составе ERP-системы – несколько десятков подсистем, работающих с информацией корпорации (***Планирование, Производство, Снабжение, Склады и сбыт, Финансы*** и пр.), вся информация хранится на сервере баз данных **Oracle** или **Microsoft SQL Server**, в систему входят также средства разработки модулей для работы с базой данных, средства для взаимодействия с системами автоматизации проектирования (САПР) и другими системами.

Среди основной массы пользователей ПК обязательным компонентом ПО, установленного на компьютере, являются офисные системы, в состав которых входят текстовый редактор, табличный процессор, СУБД, органайзер и средства для работы с электронной почтой, средства разработки презентаций и пр. В нашей стране это обычно система **Microsoft Office**.

Прочее ПО в схеме рисунке 3.1 – специальные математические программы, программы-переводчики, программы для распознавания текста и голоса, обучающие программы, экспертные, геоинформационные системы и многое другое.

Помимо классификации ПО по назначению существует ещё одна классификация – рыночная.

Лицензионные или ***коммерческие программы*** (*commercial ware*) распространяется на коммерческой основе. Для получения права на использование таких программ необходимо купить экземпляр дистрибутива такой программы. Порядок использование программы оговаривается в лицензионном соглашении, подлинность ПО подтверждается сертификатом (**Certificate of Authenticity**), имеющим уникальный код каждой копии дистрибутива (product key). Стоимость лицензионной операционной системы Windows Vista составляет около 815 руб. для версии Starter, около 3000

руб. для Home Premium и 4000-12000 руб. для версии Business. Стоимость системы Microsoft Office в различной комплектации от 190 до 11000 руб. Пакет разработчиков *Visual Studio Professional Edition 2008* стоит более 80000 руб.

Бесплатные программы (*freeware*) – распространяются авторами бескорыстно. Чаще всего это небольшие программы, написанные одним или несколькими программистами для себя, и предложенные ими для общего пользования. Такие программы можно найти в Интернете.

Условно-бесплатные программы (*shareware*) занимают промежуточное положение между коммерческими и бесплатными программами. Самая массовая группа программ, в которую входят практически все утилиты, а часто и весьма серьёзные, нужные многим программные пакеты. Эти программы предоставляются бесплатно, однако по истечении определенного срока необходимо заплатить их автору или распространителю небольшую сумму. В противном случае программа либо не загружается, либо начинает надоедать просьбами о её регистрации.

Пробные версии программ (*trialware*) – это полноценные версии коммерческих систем, которые можно использовать какое-то время бесплатно. По истечении этого времени программы прекращают работать.

Демо-версии (*demoware*) – демонстрационные версии коммерческих программ и игр с ограниченными возможностями. Например, с выключенной функцией сохранения документов в текстовом редакторе, с ограниченным количеством записей в базе данных информационной системы или с несколькими турами в играх.

Крупнейшим производителем коммерческого ПО является корпорация **Microsoft** (см. *www.Microsoft.ru* или *www.Microsoft.com*), которая производит следующие группы ПО:

1) операционные системы (**Windows**);
2) офисное ПО (**Microsoft Office System**);
3) средства для разработчиков-программистов (**Microsoft Visual Studio** и пр.);
4) сервер баз данных (**Microsoft SQL Server**);
5) системы автоматизации управления предприятиями (**Microsoft Dinamics)**;
6) ПО для КПК, коммуникаторов;
7) игры.

3.1 Операционные системы

Операционная система – комплекс программ, обеспечивающих работу компьютера.

Функции современной ОС для ПК:

1. Управление конфигурацией компьютера.
2. Управление процессами, потоками и заданиями.
3. Управление памятью.
4. Обеспечение информационной безопасности.
5. Управление подсистемой ввода-вывода.
6. Управление внешней памятью.
7. Управление файловой системой.
8. Поддержка сетей.

В управлении конфигурацией компьютера (в ОС **Windows**) ключевую роль играют данные, сохраняемые в реестре этой системы (база данных общесистемных и пользовательских параметров) и инструментарий управления **Windows** (WMI).

Процесс в ОС – исполняемая программа и ее данные в закрытом адресном пространстве, данные защиты и прочие объекты, принадлежащие процессу.

Поток – некоторый компонент внутри процесса, который получает процессорное время для выполнения.

Задание (job) – совокупность процессов, управляемых как единая группа.

ОС управляет виртуальной памятью, которую проецирует на физическую память компьютера.

Информационная безопасность предусматривает защиту любых разделяемых системных объектов (файлов, каталогов, процессов, потоков и т. д.), аутентификацию паролей при регистрации, предотвращение несанкционированного доступа, аудит событий в системе (ведение журналов событий).

Диспетчер ввода-вывода ОС подключает приложения и системные компоненты к виртуальным, логическим и физическим устройствам, а также определяет инфраструктуру, поддерживающую драйверы устройств.

Управление внешней памятью включает организацию работы с жесткими и гибкими дисками, с оптическими дисками, магнитными лентами, флэш-памятью и другими устройствами.

Управление файловой системой, определяющей методы хранения информации на дисках, важнейшая функция любой ОС. Многие ОС поддерживают работу с несколькими файловыми системами, например, **Windows 2000, XP** и его более новые версии могут управлять файловыми системами **NTFS** (основная для магнитных дисков), **FAT12, FAT16, FAT32** (были основными для **DOS** и **Windows 3.1 - Me**), **CDFS** и **UDF** (для лазерных дисков). Для управления используется драйвер файловой системы, который через драйверы устройств внешней памяти работает с информацией логических томов. Основные функции работы с информацией файловой системы – **CreateFile** (создать файл), **ReadFile** (читать файл) и **WriteFile** (записать в файл).

Поддержка сетей присутствует во всех современных ОС. В ОС **Windows** появилась с версии 3.11. К четырем базовым типам сетевого программного обеспечения относятся сетевые службы, программные интерфейсы сетевого программного обеспечения (сетевые API), протоколы и драйверы устройств сетевых адаптеров. Многоуровневая архитектура поддержки сетей обеспечивает ее легкую расширяемость при появлении новых сетевых протоколов и нового оборудования.

3.1.1 История развития операционных систем для ПК

Первый персональный компьютер PC появился в конце 1981 года (IBM 5150 PC: 4.77-MHz Intel 8088 CPU, 64KB RAM, 40KB ROM, 5.25-диймовый флоппи-дисковод) с операционной системой PC-DOS 1.0. Далее на рынке появились так называемые IBM-совместимые персональные компьютеры. Операционная система этих компьютеров называлась **MS DOS** 1.0 – корпорация **Microsoft** предоставила в распоряжение фирм, производящих эти машины, точную копию операционной системы PC-DOS.

Всего было выпущено 12 версий MS DOS: 1.0, 1.1, 2.0, 3.0, 3.3, 4.0, 4.01, 5.0, 6.0, 6.2, 6.21, 6.22. Версия 7 входила в состав **Windows** 95 и отдельно не выпускалась.

Все версии **MS DOS** были однозадачными ОС с текстовым интерфейсом и управлением из командной строки. Правда, основной программой того времени, облегчавшей работу с файловой системой, был **Norton Commander** – файловый менеджер, имевший два окна для показа папок (directory – каталогов, как их тогда называли) и файлов.

Основой **MS DOS** был 1 файл – командный процессор **command.com.** Существовали еще 2 скрытых файла – ibmbio.com и ibmdos.com (или их аналоги) – модуль расширения BIOS и модуль обработки прерываний, а также достаточной большое количество утилит (слу-

жебных программ) и драйверов внешних устройств (программ для работы с клавиатурой, мышкой, принтерами и пр. периферией).

Некоторые команды **MS DOS** могут использоваться и в настоящее время в командном интерпретаторе **Windows** (окно программы **cmd.exe**, похожее на сеанс **MS DOS**, но им не являющееся, см. рисунок 3.2), обеспечивающем текстовый интерфейс между пользователем и этой ОС, такие, как:

dir – просмотреть список файлов текущего каталога (папки);
ver – узнать версию операционной системы;
cd <имя> – сменить каталог;
md <имя> – создать подкаталог;
C: \> D: – сменить текущий;
copy... – копирование файлов;
time, date – настроить время и дату на системных часах ПК и др.

Полный список выводится по команде help, содержит 86 команд.

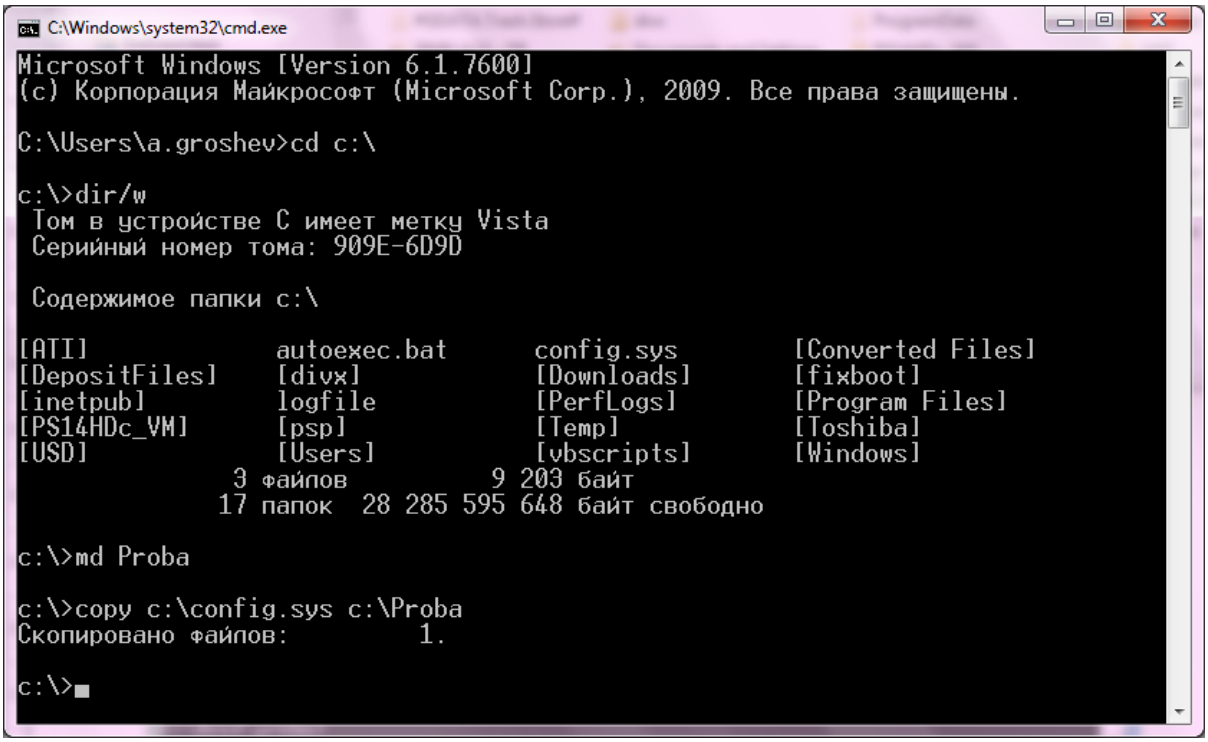


Рисунок. 3.2. Командный интерпретатор **Windows** (cmd.exe)

Первые версии **Windows** использовали **MS DOS** для своей инсталляции (установки на компьютер) или запускались из MS DOS.

Современные версии **Windows** не используют для этих целей MS DOS, но могут запускать некоторые из программ, написанных для **MS DOS** в окне или в полноэкранном режиме. Однако, сложные программы

для **MS DOS** (например, игры), не запускаются в последних версиях **Windows.** Для этой цели можно использовать программу **DosBox** (Free Software Foundation, Inc), версии которой существуют для многих современных ОС.

История развития ОС **Microsoft Windows** началась 20 ноября 1985 г. с выходом первой версии. Выпуском этой версии фирма **Microsoft** заявила о своих намерениях разработки перспективной многозадачной операционной системы с графическим интерфейсом для быстро развивающихся IBM-совместимых персональных компьютеров.

Windows 1.0 могла использовать 256 цветов, имела окна изменяемого размера, область свернутых программ (первоначальную концепцию панели задач), способность настраивать вид окон. **Microsoft** включила в нее то, что мы теперь называем **Панель управления**. Она имела элементы интерфейса, которые присутствуют в сегодняшних версиях **Windows** как поля редактирования, кнопки, полосы прокрутки и пункты меню.

В интерфейсе **Windows 1.0** окна могли быть развернуты, свернуты или расположены рядом (tiled). Окна не могли пересекаться, что было неудобно при работе с большим их количеством.

В систему были включены приложения: менеджер файлов, календарь, картотека, записная книжка, калькулятор, часы и телекоммуникационные программы. Пользователь мог переключаться между несколькими запущенными программами, т.е. система была многозадачной.

С появление новых процессоров для ПК совершенствовалась и ОС **Windows,** расширялся объем памяти, с которым она могла работать, появились первые программы, написанные для работы в **Windows** (первая версия текстового редактора **Word for Windows** вышла в 1989 г.).

Windows for Workgroups 3.11 с поддержкой одноранговых сетей вышел 15 февраля 1994 г.

Чуть раньше, в августе 1993 г. появилась новая ОС – **Windows NT 3.1** («Новая технология» – New Technology, так официально трактовали аббревиатуру **NT** в **Microsoft**). **Microsoft** создал новую операционную систему с нуля. **Windows NT** был первой операционной системой **Windows**, которая была предназначена для организации выделенного файлового сервера в составе локальной сети и включала поддержку высокопроизводительных клиент-серверных бизнес-приложений. Операционная система имела ряд новых ключевых особенностей: приоритетный многозадачный планировщик для **Windows**-приложений, интегрированная работа с сетями, разграничения прав пользователей для сервера домена,

поддержка **OS/2** и **POSIX** подсистем, поддержка многопроцессорной архитектуры, новая более надежная файловая система **NTFS**.

Windows NT 3.1 был создан для деловых пользователей и был выпущен в двух редакциях: **Server edition** – для серверов (**Windows NT Advanced Server**) и **Workstation edition** – для рабочих станций (без включения в состав поставки серверных служб).

Реально ОС **Windows** начала широко использоваться в 1995 г. с выходом версии **Windows 95**, имевшей новый графический интерфейс, очень удобный в работе. Основная масса прикладного программного обеспечения стала выпускаться для этой ОС.

Windows 95 уже не графическая оболочка пользователя для **MS DOS** (как это было в предыдущих версиях), а полноценная многозадачная 32-разрядная операционная система. Хотя пользователи могли видеть окно **MS DOS** в процессе загрузки, система заменяла собой **MS DOS 7.0** после того, как загружалась полностью.

ОС на ядре **NT** получила широкое распространение среди обычных пользователей с выпуском 25 октября 2001 г. ОС **Microsoft Windows XP**, которая имела новый графический интерфейс, использовала, как основную, файловую систему **NTFS** и имела возможность разграничения прав пользователей на локальном компьютере. Система **Windows XP** создавалась на основе усовершенствованного кода **Windows 2000**, причем были разработаны различные версии для пользователей домашних компьютеров и бизнес-пользователей: **Windows XP Home Edition** и **Windows XP Professional**.

30 января 2007 года была выпущена в продажу еще одна написанная полностью с нуля ОС – **Microsoft Windows Vista**.

22 октября 2009 года в продаже официально появилась Windows 7.

В таблице 3.1 приведены сведения по времени выпуска различных версий ОС **Microsoft Windows** и показана динамика изменения требований к объему оперативной памяти ПК

Таблица 3.1. Основные этапы развития ОС **Microsoft Windows**

Год выпуска	**Windows с файловой системой FAT**	**Windows с файловой системой NTFS**	
	Версия	Версия	ОП, МБ, минимум
1985	1.0		
1987	2.0		
1990	3.0		
1992	3.1		

Продолжение таблицы 3.1

Год выпуска	**Windows с файловой системой FAT**	**Windows с файловой системой NTFS**	
	Версия	Версия	ОП, МБ, минимум
1993		NT 3.1	
1994	3.11	NT 3.5	
1995	95 (ОП не менее 16 МБ)	NT 3.51	
1996		NT Server 4.0	16
		NT Workstation 4.0	12
1997		NT Server 4.0 Enterprise Edition	64
1998	98 (ОП не менее 24МБ)	NT Server 4.0 Terminal Server Edition	64
2000	Millennium Edition (ОП не менее 32 МБ)	2000 Server	128
		2000 Professional	64
2001		XP	128
2003		Server 2003 Standard Edition, Enterprise Edition, Web Edition	128 (256)*
		Server 2003 Datacenter Edition	512 (1024)
2007		**Vista** (x86 и x64)	1024
2008		Server 2008 (x86 и x64)	2048
2009		**7** (x86 и x64)	1024

Требования ОС **Microsoft Windows Vista** и **Windows 7** к объему ОП составляет 1 ГБ для версий **Home Premium** / **Business** / **Ultimate** при этом требуется не менее 16 ГБ свободного места на жестком диске для установки системы.

Из таблицы 3.1 видно, что требование по объему ОП, которое было и остается главным лимитирующим фактором для установки новой ОС на ПК, в 1996 г. было ниже для **Windows NT Workstation 4.0** (12 МБ), чем для **Windows 95** (при одинаковом графическом интерфейсе систем). Фирма **Microsoft** ожидала перехода на ОС **NT** для ПК, работающих на предприятиях, однако более распространенными версиями ОС оставались **Windows 95** и **98**. Повсеместный переход к **Windows** на ядре **NT**, как сказано выше, произошел только после 2001 г. **Windows XP** оказалась самой долгоживущей версией, она была лидирующей на протяжении не менее 5 лет в 2001 – 2007 г.

Новые ОС **Windows Vista** и **Windows** 7 предъявляют достаточно высокие требования к конфигурации ПК и не могут полноценно функционировать на старых ПК. Высокие требования предъявляются не только к объему ОП, но и к графической видеоподсистеме ПК (32 битный видеоадаптер с аппаратной поддержкой DirectX 9c, минимум 128 МБ графической памяти, аппаратная поддержка пиксельных шейдеров 2.0).

3.1.2 Файловые системы

Файловая система – методы организации хранения и использования информации во внешней памяти компьютера, которой являются магнитные и лазерные диски, флэш-память и прочие ее виды.

Каждая операционная система использует одну основную файловую систему, а также дополнительно может работать с некоторыми другими файловыми системами.

Файл – упорядоченная совокупность информации на диске, имеющая свое имя.

Для файловой системы **NTFS** теоретически файл – единица информации, имеющая номер в Master File Table (**MFT**, общей таблице файлов), однако на практике все же основной атрибут файла – его имя.

Сохранение информации о файлах обеспечивает файловая система. Правила именования файлов, способы доступа к данным файла, структура его данных зависят как от типа используемой файловой системы, так и от типа файла. Файловая система берет на себя размещение информации файла на диске, сохранение информации об этом в таблице файлов диска и обеспечение доступа к этим данным.

Кроме имени каждый файл может иметь и другие свойства:

- расширение имени файла, характеризующее его тип (символы после последней точки, справочник типов см. *http://open-file.ru/*);
- атрибуты (архивный, только чтение, скрытый, шифрованный и пр.);
- время создания, изменения, последнего открытия;
- права доступа (безопасность);
- метаданные (название, тема, ключевые слова, авторы и пр.).

Каждый логический раздел жесткого диска может быть отформатирован в той или иной файловой системе. Например, основной файловой системой ОС **Windows** на ядре **NT** является **NTFS**. Однако ОС **Windows**

XP может также работать и с информацией, хранящейся на разделах диска с файловой системой **FAT** и **FAT32.**

При организации разделов в первый сектор диска (цилиндр 0, головка 0, сектор 1) заносится главная загрузочная запись – **Master Boot Record** (**MBR**). В ней содержатся:

1) исполняемый код – программа начальной загрузки (внесистемный загрузчик – non-system bootstrap);
2) таблица разделов – таблица разбиения диска (Disk Partition Table).

В таблице разбиения диска содержатся сведения о том, с каких цилиндров, головок и секторов начинаются и какими заканчиваются имеющиеся на диске разделы. В этой таблице также содержатся указания для системной **BIOS**, какой из разделов является загрузочным, т.е. где следует искать основные файлы операционной системы. Сектора и дорожки формируются во время низкоуровневого форматирования при изготовлении жесткого диска (размер сектора равен 512 байт).

Поскольку файловые системы являются общим хранилищем файлов, принадлежащих разным пользователям, системы управления файлами должны обеспечивать авторизацию доступа к файлам. В общем виде подход состоит в том, что по отношению к каждому зарегистрированному пользователю данной операционной системы для каждого существующего файла указываются действия, которые разрешены или запрещены данному пользователю или группе пользователей. В современных ОС для ПК предопределены 3 основные группы с разными правами: **администраторы, опытные пользователи, пользователи.**

Первой файловой системой для ПК была **FAT** – файловая система операционной системы **MS DOS**. Эта система и в настоящее время используется для форматирования дискет (на смену которым уже сейчас пришла USB-флэш-память). Позднее появилась файловая система **NTFS**, разработанная для операционной системы **Windows** NT 3.1, более эффективная и надежная для работы с дисками большого объема. Однако **NTFS** требуется для нормальной работы больший объем оперативной памяти ПК, чем **FAT** (в 90-х годах 20 века ПК обычно имели ОП не более 8 Мбайт).

С выходом **Windows** 95 появилась возможность использовать длинные имена файлов в файловой системе **FAT32** и жесткие диски больших объемов.

В файловых системах **FAT** и **NTFS** может существовать несколько логических дисков (каждый винчестер может быть одним разделом или может быть разбит на несколько логических дисков, общие сетевые ресур-

сы при необходимости также могут быть подключены как логические диски). Каждый логический диск имеет имя – латинская буква с двоеточием. A: и B: зарезервированы для флоппи-дисководов; жесткие, лазерные, сетевые диски – C:, D: и т. д.. С этого имени начинается дерево каталогов диска (см. рисунок 3.3). Полное имя файла начинается с имени диска, далее может идти перечисление папок (путь к файлу), последним идет имя файла. Например, E:\Program Files\Microsoft Office\Office11\ Winword.exe. Таким образом, в этих системах может существовать несколько изолированных файловых деревьев. Каждый файл имеет расширение (символы после последней точки в имени файла – Winword.exe, Заявка12.12.2006.doc).

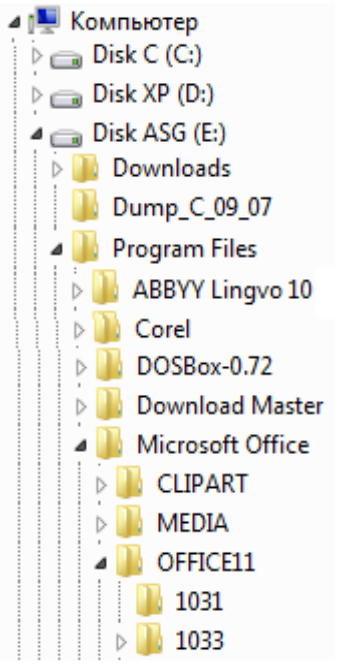


Рисунок 3.3. Дерево каталогов (папок)

На ПК может использоваться также ОС **Linux**, которая имеет собственную файловую систему. Принципиальное отличие этой файловой системы – отсутствие имен дисков и расширений файлов. Это централизованная файловая система, вся совокупность каталогов и файлов представлена как единое дерево. Для подключения дисков используется операция монтирования дисков к общей файловой системе.

CD и DVD диски используют файловые системы, предназначенные специально для работы с оптическими дисками (ISO 9660, **CDFS, UDF** и пр.).

В файловой системе **FAT** каждый логический раздел диска (том) имеет собственную таблицу размещения файлов (File Allocation Table – **FAT**), которая выполняет две функции: содержит информацию распределения данных каждого файла по кластерам раздела диска в форме списка связей модулей распределения и указывает, какие кластеры свободны.

На рисунке 3.4 показана структура тома в системе **FAT.**

Загрузочный сектор раздела	**FAT**	Копия **FAT**	Корневой каталог	Файлы и каталоги

Рисунок 3.4. Структура тома в **FAT**

Том, отформатированный для использования файловой системы **FAT**, размечается по кластерам (минимальная адресуемая единица в фай-

ловой системе). Размер кластера по умолчанию определяется размером тома.

В название версии **FAT** входит число, которое указывает разрядность, применяемую для идентификации кластеров на диске. 12-разрядный идентификатор кластеров в **FAT12** ограничивает размер дискового раздела 2^{12} (4096) кластерами. **FAT16** – за счет 16-разрядных идентификаторов кластеров – может адресовать до 2^{16} (65 536) кластеров. В **Windows** 2000 размер кластера **FAT16** варьируется от 512 байт до 64 Кб, поэтому размер FAT16-тома ограничен 4 Гб. Размер кластеров, используемых **Windows 2000**, зависит от размера тома (таблица 3.2).

Таблица 3.2. Размеры кластеров в **FAT16** по умолчанию

Размер раздела, МБ	Количество секторов в кластере	Размер кластера, КБ
0 – 32	1	0,512
33 – 64	2	1
65 – 128	4	2
129 – 256	8	4
257 – 511	16	8
512 – 1023	33	16
1024 – 2047	64	32
2048 – 4096	128	64

Пользователь имеет возможность указать другой размер кластера, однако устанавливаемый размер не может быть меньше размера по умолчанию. Ограничение системы **FAT** на размер логического диска составляет 4 Gb. Файловую систему **FAT**, не рекомендовалось использовать для томов, размер которых превышал 511 Mb.

Файловая система **FAT32** впервые появилась в **Windows 95 OSR2**, затем использовалась в **Windows 98** и **Windows ME**. **FAT32** использует 32-разрядные идентификаторы кластеров, но при этом резервирует старшие 4 бита, так что эффективный размер идентификатора кластера составляет 28 бит. Поскольку максимальный размер кластеров **FAT32** равен 32 Кб, теоретически **FAT32** может работать с 8-терабайтными томами (таблица 3.3).

Таблица 3.3. Размер кластеров на томах **FAT32** по умолчанию

Размер раздела	Размер кластера, КБ
от 32Мб до 8Гб	4
8 – 16 Гб	8
16 – 32 Гб	16
более 32 Гб	32

Помимо большего предельного числа кластеров преимуществом **FAT32** перед **FAT12** и **FAT16** является тот факт, что место хранения корневого каталога **FAT32** не ограничено предопределенной областью тома, поэтому его размер не ограничен. Кроме того, для большей надежности **FAT32** хранит вторую копию загрузочной за-

писи. В **FAT32**, как и в **FAT16**, максимальный размер файла равен 4 Гб, поскольку длина файла в каталоге описывается 32-битным числом.

Файловая система **NTFS** обеспечивает производительность, надежность и совместимость, которые не в состоянии предоставить файловая система **FAT**. **NTFS** разрабатывалась с целью обеспечения высокой скорости и надежности выполнения операций с файлами (включая чтение, запись, поиск) при использовании дисков больших объемов. **NTFS** использует 64-разрядные индексы кластеров. Это позволяет **NTFS** адресовать тома размером до 16 экзабайт (16 миллиардов Гб).

NTFS поддерживает контроль доступа к папкам и файлам как для локальных, так и сетевых ресурсов.

Как и другие файловые системы, **NTFS** делит все полезное место диска на кластеры – адресуемые блоки данных, размер которых может быть от 512 байт до 64 Кбайт (см. таблицу 3.4), стандартом считается кластер размером 4 Кбайт.

NTFS поддерживает ряд дополнительных возможностей – защиту файлов и каталогов, дисковые квоты, сжатие файлов, символьные ссылки на основе каталогов и шифрование.

Таблица 3.4. Размеры кластеров в **NTFS** по умолчанию

Размер раздела, МБ	Размер кластера, КБ
512 и менее	0,512
513 – 1024	1
1025 – 2048	2
более 2048	4

Одно из важнейших свойств **NTFS** – ***восстанавливаемость.*** При неожиданной остановке системы целостность метаданных тома **FAT** может быть утрачена, что вызовет повреждение структуры каталогов и значительного объема данных. **NTFS** ведет журнал изменений метаданных путем протоколирования транзакций, поэтому целостность структур файловой системы может быть восстановлена без потери информации о структуре файлов или каталогов (однако данные файлов могут быть потеряны). Транзакция – операция, выполняющаяся по принципу «все или ничего». При неудачном выполнении операции в файловой системе происходит ***откат*** (roll back), после которого система возвращается в исходное состояние, в котором она была до начала транзакции.

NTFS – это не только восстанавливаемая, защищенная, надежная и эффективная файловая система, способная работать в ответственных информационных системах. Она поддерживает ряд дополнительных возможностей:

- файлы и каталоги могут содержать несколько потоков данных;
- имена файлов, каталогов и томов имеют кодировку Unicode;

- универсальный механизм индексации файлов для ускорения их поиска;
- динамическое переназначение плохих кластеров;
- работа со сжатыми файлами;
- протоколирование изменений;
- квоты томов, индивидуальные для каждого пользователя;
- отслеживание ссылок (правильность ссылок ярлыков и OLE-связей при перемещении источников);
- шифрование (механизм Encrypting File System – EPS);
- поддержка POSIX (чувствительность к регистру в именах, цепочечные разрешения доступа, метки времени изменения файлов).

В **NTFS** все данные, хранящиеся на томе, содержатся в файлах. Это относится и к структурам данных, используемым для поиска и выборки файлов, к начальному загрузочному коду и к битовой карте, в которой регистрируется состояние пространства всего тома (метаданные **NTFS**). Хранение всех видов данных в файлах позволяет файловой системе легко находить и поддерживать данные, а каждый файл может быть защищен дескриптором защиты. Кроме того, при появлении плохих секторов на диске **NTFS** может переместить файлы метаданных.

Диск **NTFS** делится на две части. Первые 12% диска отводятся под **MFT** зону – пространство, в котором растет файл **MFT** (Master File Table, главная таблица файлов, см. рисунок 3.5). Это сделано для того, чтобы файл **MFT** не был фрагментирован. Но когда все остальное место на диске заполняется – зона **MFT** может несколько раз уменьшаться в два раза, пока это возможно (на больших дисках эта зона – десятки ГБ и полностью под файл **MFT** не используется).

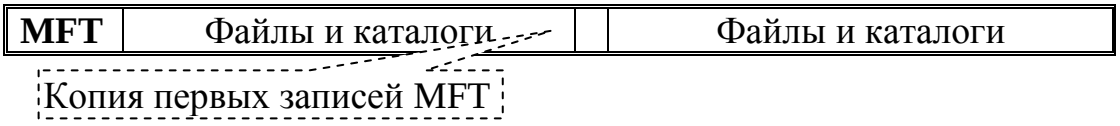


Рисунок 3.5. Структура тома в NTFS

MFT реализована как массив записей о файлах и папках (номер в **MFT**, имя, размер, положение на диске отдельных фрагментов и т. д). Размер каждой записи о файле фиксирован и равен 1 Кб. Кроме **MFT** в корневом каталоге каждого тома **NTFS** имеется набор файлов метаданных с информацией, необходимой для реализации структуры файловой системы. Имена всех файлов метаданных **NTFS** начинаются со знака доллара ($), хотя эти знаки скрыты. Так, имя файла **MFT** – $**MFT** (см. таблицу 3.5. Остальные файлы **NTFS**-тома являются обычными файлами и каталогами.

Таблица 3.5. Метафайлы **NTFS**

Имя файла	Назначение
$MFT	файл **MFT**
$MFTmirr	копия первых 16 записей **MFT**, размещенная посередине диска
$LogFile	файл поддержки журналирования (лог-файл)
$Volume	служебная информация о томе – метка тома, версия файловой системы, т.д.
$AttrDef	список стандартных атрибутов файлов на томе
$.	корневой каталог
$Bitmap	карта свободного места тома
$Boot	загрузочный сектор (если раздел загрузочный)
$Quota	файл, в котором записаны права пользователей на использование дискового пространства (с NT5)
$Upcase	таблица соответствия заглавных и прописных букв в именах файлов на текущем томе

В таблице 3.6 приведены сравнительные данные по файловым системам **FAT** и **NTFS**.

Таблица 3.6. Сравнение файловых систем **NTFS** и **FAT**

Наименование	**NTFS**	**FAT**	**FAT32**
Доступ к файлам локальных разделов диска из ОС	**Windows 2000**, **XP**, семейство **Windows Server 2003**, **Windows NT 4.0 SP 5** или более поздние	**MS-DOS**, все версии **Windows, OS/2**.	**Windows 95 OSR2, 98, ME, 2000, XP, Windows Server 2003**
Минимальный размер тома	10 МБ	Флоппи-диск	33 МБ; 32 ГБ в **Windows 2003**
Максимальный размер тома	16 ТБ минус 4 КБ для динамических дисков с размером сектора 4 КБ	4 ГБ. Не поддерживает домены	2 ТБ
Максимальный размер файла	16 ТБ минус 64 КБ	2 ГБ	4 ГБ

Если Вы используете операционную систему на ядре NT (**Windows XP, Vista, 7**), то использовать какую-либо файловую систему, отличную от **NTFS** – значит существенно ограничивать свое удобство и гибкость ра-

боты самой операционной системы. Множество полезных возможностей NT напрямую завязано на физическую и логическую структуру этой файловой системы.

Файловые системы CD и DVD дисков – **CDFS** и **UDF.**

CDFS или файловая система CD-ROM, имеет относительно простой формат, определенный в 1988 году применительно к носителям CD-ROM, предназначенных только для чтения. В ОС **Windows** реализована **CDFS** совместимая со стандартом ISO 966O, которая поддерживает длинные имена файлов в соответствии с ISO 966O Level 2. Формат **CDFS** очень прост и поэтому ему присущ ряд ограничений:

- длина имен файлов и каталогов не должна превышать 32 символов;
- глубина вложения каталогов не может превышать 8 уровней.

CDFS считается устаревшим форматом, поскольку принят новый стандарт для компакт-дисков – Universal Disk Format (**UDF**). **Windows** поддерживает файловую систему **UDF**, совместимую со стандартом ISO 13346. Ассоциация **OSTA** (Optical Storage Technology Association) определила **UDF** в 1995 году как формат магнитооптических носителей, главным образом DVD-ROM, предназначенный для замены **CDFS.**

Файловая система **UDF** обладают следующими преимуществами:

- длина имени файла – до 255 символов;
- максимальная длина пути – 1023 символа;
- имена файлов могут включать буквы как верхнего, так и нижнего регистра.

Windows Vista и **Windows 7** поддерживают работу с файловыми системами **CDFS** и **UDF.**

3.1.3 Операционные системы Windows Vista и Windows 7

ОС Windows Vista и созданная на ее ядре Windows 7 призваны помочь пользователю быстрее работать с большим количеством информации разных типов, хранящейся на компьютере. Этому служат усовершенствованные меню Пуск, Панель задач, Проводники и множество других нововведений. Существуют 32- и 64-разрядные версии этих ОС (в одном дистрибутиве для Windows 7), причем возможность использовантя 64-разрядные версии может быть определена самой программой установки.

Семейства Windows

Существует несколько основных вариантов (выпусков) Windows.
Для домашних пользователей:

1. ***Home Basic*** (Домашняя базовая)
2. ***Home Premium*** (Домашняя расширенная)

Для предприятий:

3. ***Windows Vista Business*** или ***Windows 7 Professional***
4. ***Windows Ultimate*** (Максимальная)/***Windows Enterprise***.

Кроме того, существует дополнительный выпуск ***Window Starter,*** предназначенный для домашних пользователей начального уровня развивающихся рынков.

Выпуски различаются возможностями и комплектом устанавливаемых на компьютере компонентов, входящих в состав ОС, полный список различий см. в **Приложении 3**.

Некоторые самые существенные различия выпусков:

- интерфейс пользователя **Windows Aero**, архивация файлов пользователя по расписанию – недоступны в выпуске ***Home Basic***;
- поддержка двух процессорных сокетов, теневое копирование для восстановления предыдущих версий документов – недоступны в ***Home Basic*** и ***Home Premium.***
- **Windows** Media Center и DVD-студия присутствует только в выпусках ***Home Premium*** и ***Ultimate***;
- родительский контроль – отсутствует в выпусках ***Business*** и ***Enterprise;***
- распознавание речи – отсутствует в выпуске ***Business***;
- шифрование диска **Windows BitLocker** и возможность выбора языков интерфейса пользователя для всех стран мира (36 языков) – только в ***Enterprise*** и ***Ultimate***.

Требования к компьютерам выпуска Windows Premium и системные требования для использования интерфейса Windows Aero совпадают. Минимальные требования для 32-х разрядной ОС:

- 32-разрядный (x86) или 64-разрядный (x64) процессор с тактовой частотой 1 гигагерц или выше
- 1 гигабайт (ГБ) (для 32-разрядной системы) или 2 ГБ (для 64-разрядной системы) оперативной памяти
- 16 ГБ (для 32-разрядной системы) или 20 ГБ (для 64-разрядной системы) пространства на жестком диске
- графическое устройство DirectX 9 с драйвером WDDM версии 1.0 или выше.

Интерфейс пользователя

В **Windows Vista** и **Windows** 7 возможны четыре вида интерфейса: упрощенный, классический, стандартный и **Windows Aero**.

Упрощенный интерфейс пользователя предназначен для настольных ПК начального уровня. Он значительно модернизирован по сравнению с предыдущими версиями **Windows**. Этот интерфейс содержит переработанное меню кнопки ***Пуск*** **,** позволяющее быстро найти и запустить любое приложение на ПК. Усовершенствованные Проводники и такие вспомогательные функции, как интегрированное средство быстрого поиска (поле поиска справа от адресной строки) и динамические значки, позволяют повысить эффективность работы пользователя. Упрощенный интерфейс можно использовать в любом выпуске, в том числе в выпуске ***Starter***.

Классический интерфейс сочетает в себе все возможности упрощенного интерфейса с оформлением в стиле **Windows 2000**. Этот вид интерфейса включен во все выпуски операционной системы, в том числе в выпуск ***Starter***.

Стандартный интерфейс предназначен для настольных ПК среднего уровня. На компьютерах, графическое оборудование которых поддерживает новую технологию модели видеодрайвера **Windows** (**WDDM**), передовые графические технологии гарантируют удобную работу с окнами, повышенную надежность и отсутствие проблем с отображением графики. Стандартный интерфейс недоступен в выпуске ***Starter***, он предназначен для покупателей выпуска ***Home Basic***. Стандартный интерфейс имеет такую же производительность и надежность, как интерфейс **Windows Aero**, однако в нем отсутствуют такие визуальные возможности интерфейса Aero, как новый вид списка окон и трехмерный вид списка окон. Этот интерфейс недоступен в выпусках ***Home Premium, Business, Enterprise и Ultimate***.

Интерфейс Windows Aero является основным, имеет профессионально выполненный дизайн рамок и панелей окон в стиле прозрачного стекла с эффектами отражения. Цвет, насыщенность и уровень прозрачности интерфейса можно изменять. Две новые интересные возможности интерфейса **Aero** – пролистывание с отображением содержания окон (***Alt+Tab***) и трехмерный вид списка окон (***Start+Tab***, при этом окна располагаются друг за другом в трехмерной проекции) – позволяют пользователю уверенно управлять большим количеством окон на рабочем столе, упорядочивая их необычным, но удобным способом. Интерфейс **Aero** доступен только в выпусках ***Home Premium, Business, Enterprise и Ultimate***.

В **Windows 7** добавлены новые возможности работы с окнами: **Aero Peek** – показ полного окна для миниатуры кнопки панели задач с просвечиванием других окон и кнопка в правой части панели задач для просвечивания, сворачивания и разворачивания всех окон, **Aero Shake** – встряхивание окна для сворачивания и разворачивания всех остальных окон и **Aero Snap** – изменение размера окна при перетаскивании его к верхней или боковой границе экрана.

Усовершенствованные меню Пуск, Панель задач, Проводники, Диалоговые окна, Панель управления и Проводник программ (в русской версии называется Программы и компоненты, в более ранних версиях - Установка и удаление программ) помогают легко и быстро управлять настройками ПК, запускать программы и открывать файлы, устанавливать и удалять компоненты **Windows** и приложения пользователей. **Windows 7** имеет новую концепцию работы с информацией путем создания библиотек документов, музыки, видеофайлов, изображений и пр. – тематического объединения файлов и папок независимо от того, где они фактически хранятся.

Рабочий стол

После регистрации в ОС пользователь видит свой ***Рабочий стол*** (см. рисунок 3.6), который является одним из компонентов его профиля. Другой пользователь может иметь свое оформление ***Рабочего стола***, другой набор ярлыков и другой список программ в меню ***Пуск***.

Настройка ***Рабочего стола*** выполняется с помощью пункта контекстного меню **Персонализация** (рисунок 3.7).

На Рабочем столе можно разместить ярлыки ***Компьютер, Файлы пользователя***, ***Сеть***, ***Корзина***, ***Панель управления***, если Вы привыкли использовать их в своей работе (ярлыки отсутствуют при первом запуске).

В нижней части экрана находится ***Панель задач***. На ней слева находится круглая кнопка с флагом корпорации **Microsoft** – **Пуск**, которая открывает ***Главное меню*** **Windows**.

Новый элемент ***Рабочего стола*** – мини-приложения (гаджеты). В состав ОС **Vista** включены мини приложения ***Часы***, ***Календарь***, ***Индикатор ЦП***, ***Показ слайдов***, ***Заголовки новостей Web-каналов*** и пр.

Рисунок 3.6. Рабочий стол Windows 7 и список окон (ALT+Tab)

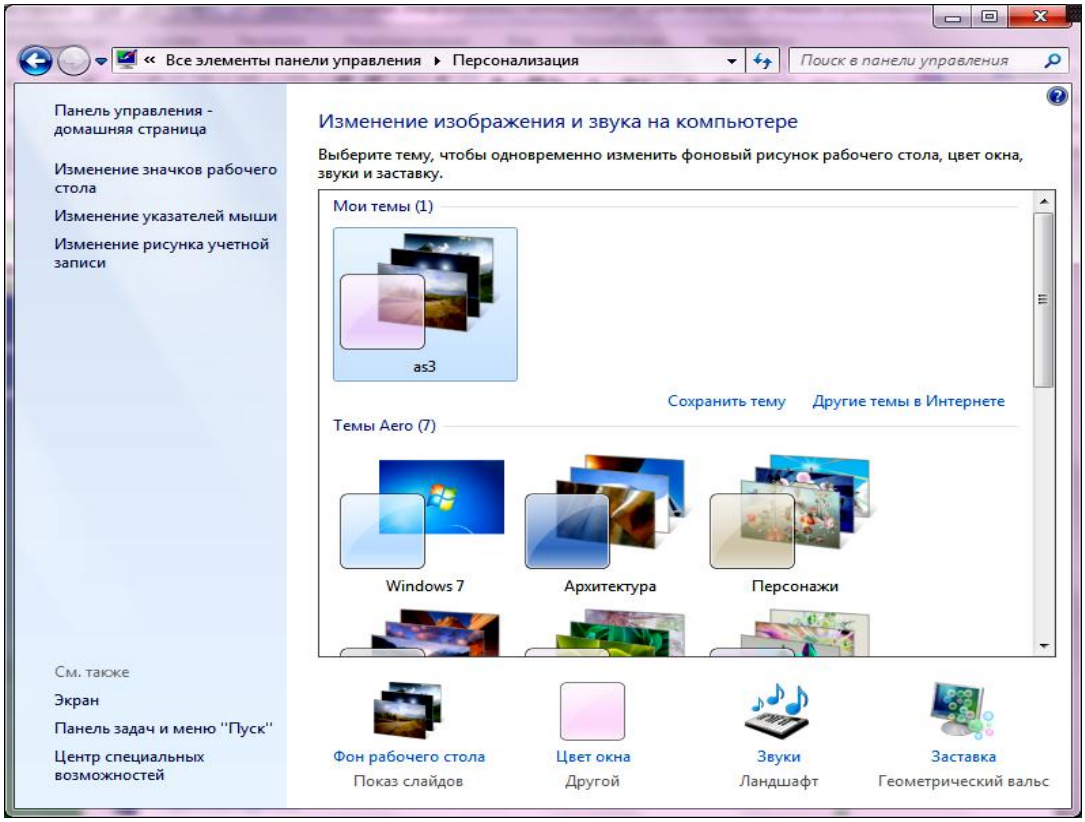


Рисунок 3.7. Окно настройки рабочего стола – Персонализация

Меню Пуск

Меню Пуск было усовершенствовано с целью упростить и ускорить поиск приложений, файлов, сообщений электронной почты, компонентов панели управления, записей в избранном, а также программ в пункте меню **Все программы** (см. рисунок 3.8).

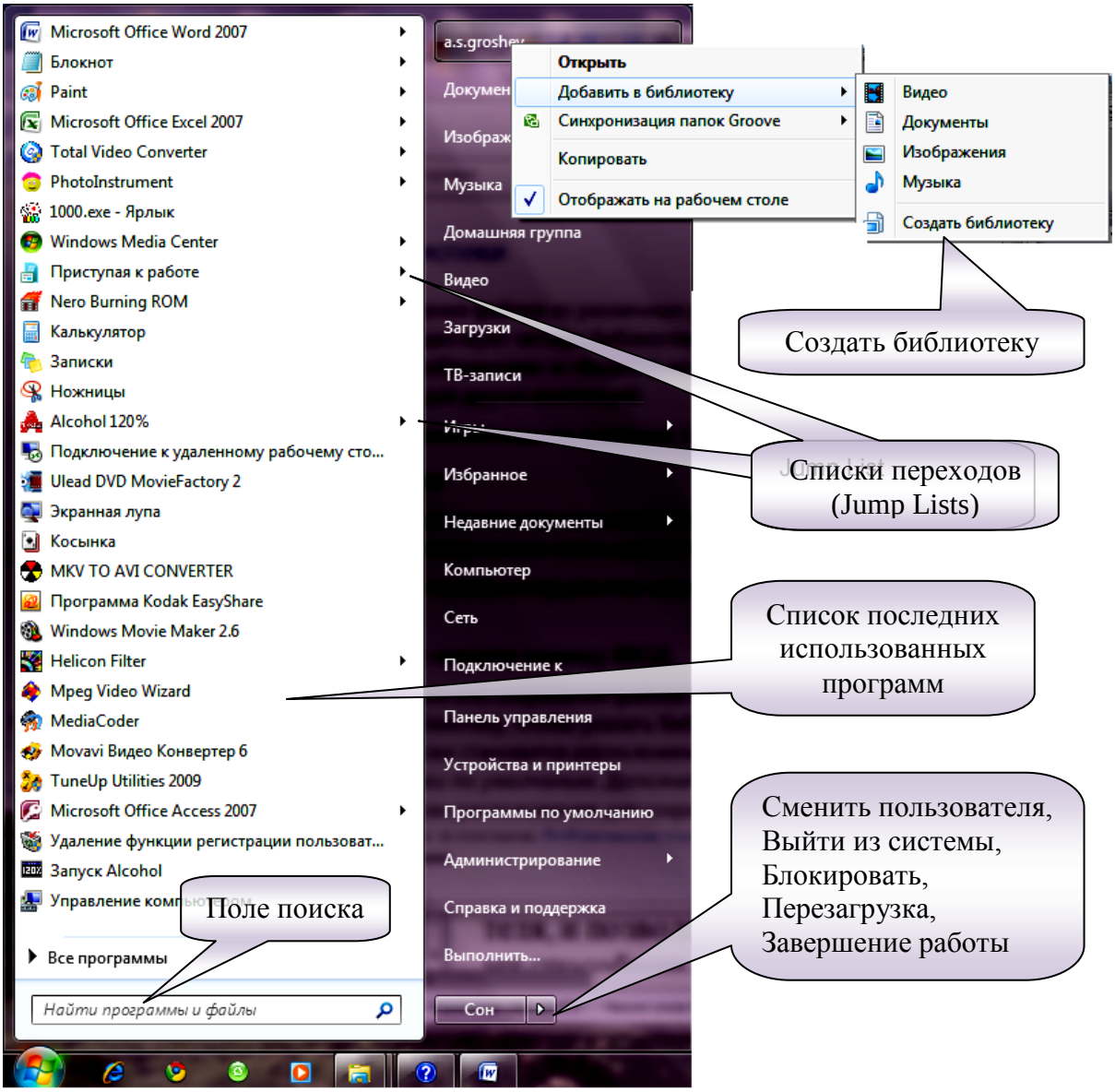


Рисунок 3.8. Главное меню

Чтобы найти определенное приложение или файл, можно ввести ключевое слово в поле поиска меню ***Пуск***. По мере ввода **Windows** осуществляет поиск по именам файлов, метаданным и полному тексту файлов. Варианты выбора группируются по типу, например: приложения, файлы Интернета, файлы приложений и сообщения электронной почты. Так, что-

бы найти приложение **Microsoft Office Excel**, достаточно ввести часть его названия, например Exc, в поле быстрого поиска. **Windows Vista** немедленно отобразит в меню ярлык **Excel** (при условии, что это приложение установлено) и другие подходящие ссылки.

Windows Vista и **7** позволяют быстрее отображать список всех приложений, установленных на компьютере. Подменю по-новому располагаются на экране, занимают меньше места за счет того, что открываются в той же позиции по вертикали, что и пункт, их вызвавший.

Интерфейс Windows Aero

Прозрачность. Рамки и некоторые панели окон являются полупрозрачными. Цвет, насыщенность и уровень прозрачности интерфейса можно изменять, что позволяет настроить внешний вид в соответствии с предпочтениями пользователя (Панель управления, раздел Персонализация).

Динамическое изменение положения окон. При открытии и закрытии окна ведут себя по-новому. Открытые окна плавно скользят по экрану в процессе перемещения или изменения размера.

Поддержка экранов с высоким разрешением. В предыдущих версиях **Windows** и многих приложениях сторонних разработчиков не обеспечивалась адекватная поддержка экранов с разрешением более 96 точек на дюйм. В результате элементы диалоговых окон были слишком мелкими, пользователям приходилось отказываться от работы с высоким разрешением. Особые проблемы возникали в переносных компьютерах с небольшими экранами. Теперь в **Windows** появилась возможность изменять масштаб интерфейса и приложений на экранах с разрешением до 144 точек на дюйм.

Динамические эскизы кнопок панели задач. Динамические эскизы на панели задач **Windows Aero** отображают содержимое открытых окон. Задержав указатель мыши на эскизе над кнопкой на панели задач, можно в реальном времени увидеть содержимое соответствующего окна на ***Рабочем столе***, не переключаясь на него (см. рисунок 3.9).

Список окон (Alt+Tab). Это обновленный вариант возможности, доступ к которой и в предыдущих версиях **Windows** осуществлялся с помощью сочетания клавиш Alt+Tab (см. рисунок 3.6). В новом интерфейсе вместо стандартных значков отображаются эскизы открытых окон, отображающие их содержимое. Это позволяет быстрее находить нужные окна.

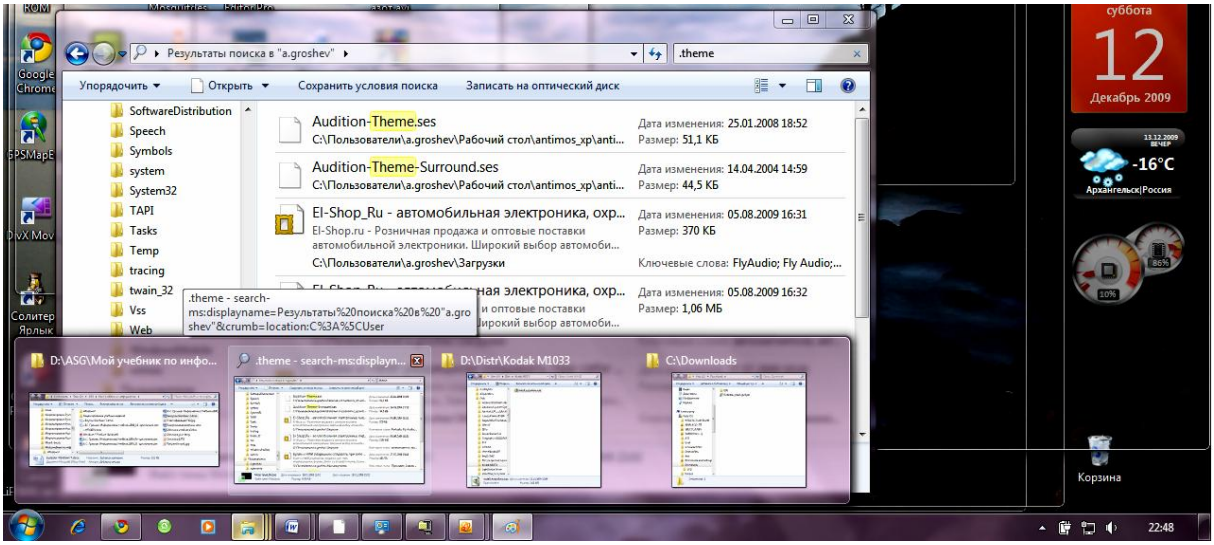


Рисунок 3.9. Динамические эскизы кнопок панели задач

Трехмерный вид списка окон. При нажатии клавиш **Start**+**Tab** динамически отображаются все открытые окна в трехмерной проекции (см. рисунок 3.10).

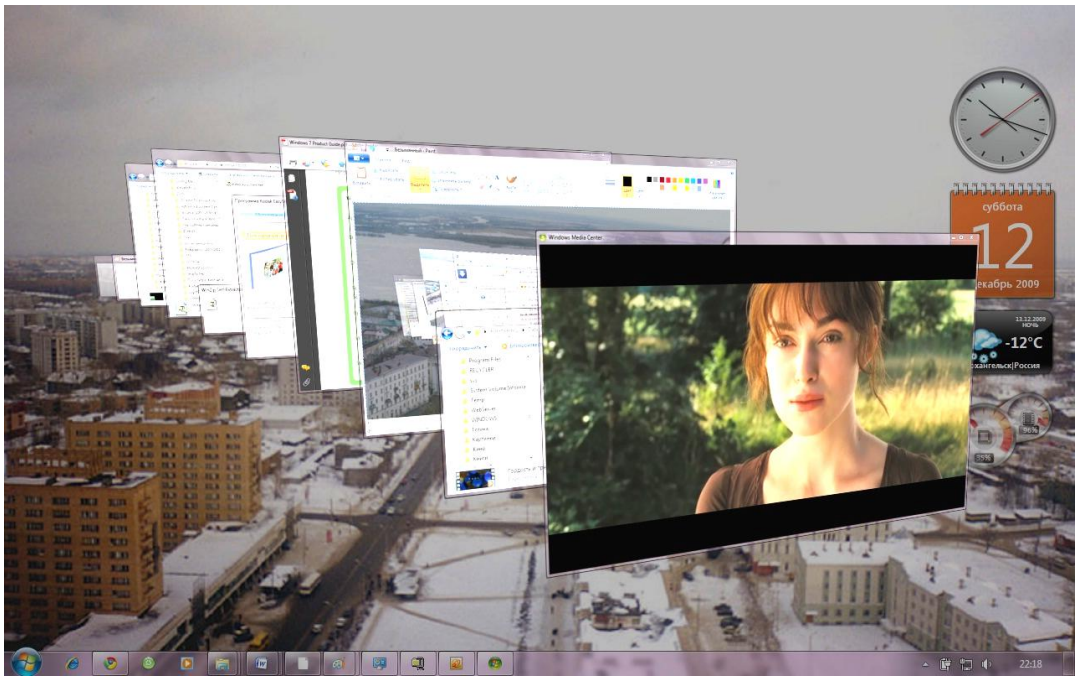


Рисунок 3.10. Трехмерный вид списка открытых окон

С помощью этой функции можно переключаться между открытыми окнами для нахождения нужного окна. В данном режиме могут отображаться даже такие динамичные процессы, как воспроизведение видео (см. окно на первом плане рис. 3.10). Для плавного переключения между от-

крытыми окнами и выбора нужного окна можно использовать клавиши управления курсором или колесико прокрутки мыши.

Проводник

Проводник – основное средство поиска информации в папках на локальных дисках компьютера, а также общих ресурсов локальной сети и Интернета.

К основным элементам Проводников **Windows** относятся (см. рисунок 3.11):

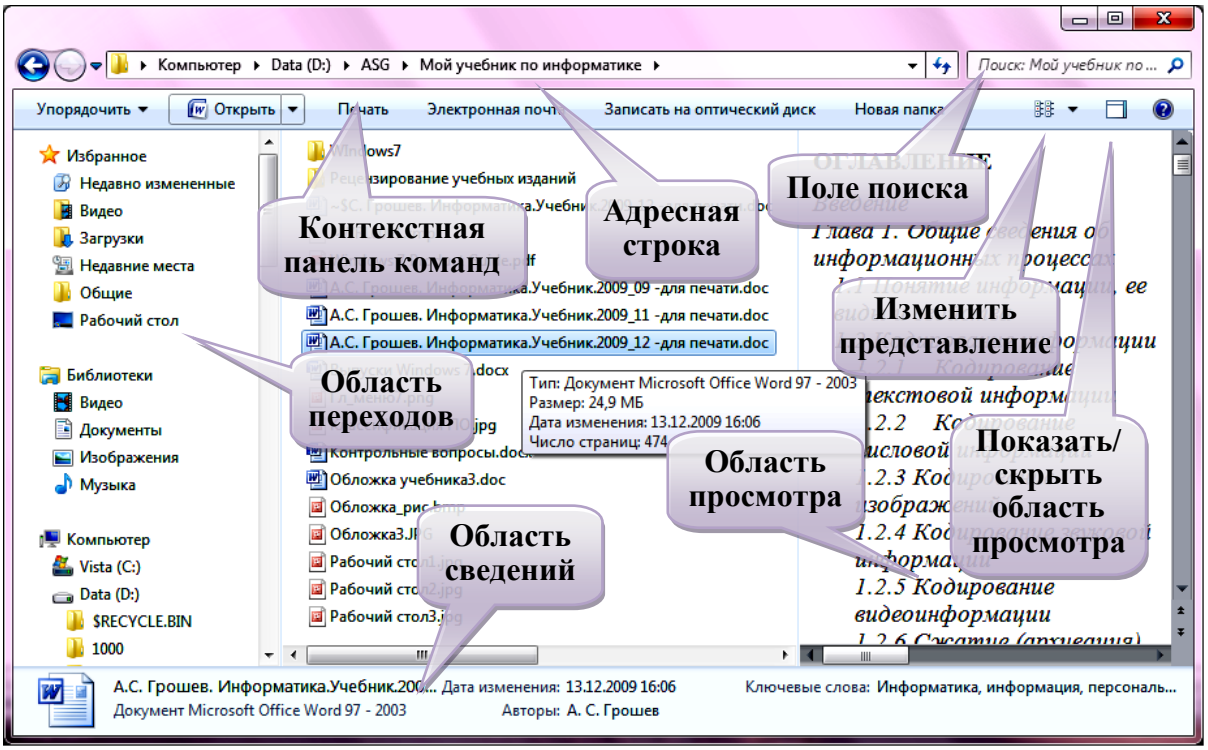


Рисунок 3.11. Проводник Windows 7

- **контекстная панель команд** (или панель инструментов) – набор кнопок изменяется в зависимости от позиции в окне и выбранного в нем объекта;
- **адресная строка**, состоящая из отдельных пунктов – раскрывающихся списков для каждой папки в адресе, что позволяет осуществлять простой переход назад или вперед в любое место из текущего положения. Старая кнопка **Вверх** исчезла за ненадобностью. При щелчке кнопкой мыши на пустом месте адресной строки она превращается в обычную строку, которую можно скопировать в буфер обмена (такая необходимость иногда появляется);

- **поле поиска** – средство быстрого поиска, которое позволяет отбирать файлы в окне Проводника по заданному слову или фразе; поиск выполняется по именам и контексту файлов;
- **динамические значки**, отображающие содержимое каждого файла;
- **область переходов**(можно показывать или нет), содержащая ссылки на те места, где хранятся документы, рисунки, музыка, фотографии, папки поиска и отображающая дерево папок на компьютере;
- **область сведений** (можно показывать или нет), предоставляющая подробные сведения о файлах (метаданные) и позволяющая легко добавлять и изменять метаданные файлов;
- **область просмотра** (можно показывать или нет), позволяющая пользователям просматривать содержимое файлов в тех приложениях, которые поддерживают данную функцию.

Если Вы привыкли пользоваться пунктами меню окна **Проводника**, можно настроить показ и этого средства работы с данными, однако, более прогрессивным является использование только контекстной панели и контекстного меню или вызвать меню, нажав кнопку ALT. Настройка окна выполняется с помощью кнопки **Свойства**, расположенной в правой части панели команд, когда в окне не выбран никакой файл.

Контекстная панель команд. Содержимое панели команд зависит от содержимого папки и выбранной позиции в окне **Проводника**. Например, при выборе в списке текстового файла или файла документа **Word** в панели задач, кроме постоянных кнопок ***Упорядочить, Виды, Записать на оптический диск*** и ***Новая папка,*** появятся кнопки ***Открыть, Печать, Электронная почта***. Название постоянной кнопки ***Упорядочить*** не соответствует пунктам ее подменю, которое содержит команды ***Вырезать, Копировать, Вставить, Отменить, Вернуть, Выделить все, Представление (***с подменю ***Строка меню, Область сведений, Область предпросмотра, Область переходов), Удалить, Переименовать, Удалить свойства, Свойства, Закрыть,*** т. е. почти все пункты старых пунктов меню окна ***Файл, Правка*** и ***Вид.***

Поле поиска Проводника немедленно осуществляет отбор данных по именам файлов, свойствам файлов (метаданным) и тексту внутри каждого файла и быстро отображает результаты. Например, при вводе в поле быстрого поиска букв «РФ» будут отобраны и показаны папки и файлы, содержащие эти буквы в своих названиях и в своем содержании (см. рисунок 3.12,).

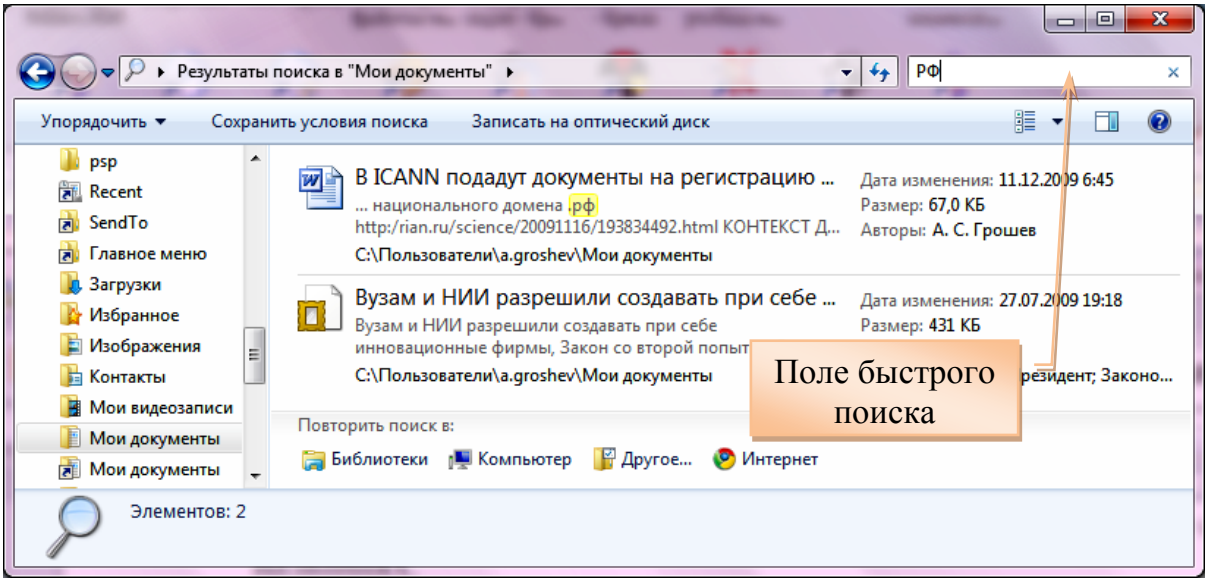


Рисунок 3.12. Отбор данных с использованием поля поиска

Значки папок и файлов в **Проводнике Windows** являются динамическими и могут отображать содержимое в режимах **Огромные значки – Крупные значки – Обычные значки – Мелкие значки.** Размер их может плавно изменяться с использованием скроллинга кнопки ***Изменить представление*** или прокруткой колесика маши при нажатой кнопке CTRL. Пользователь может просматривать, не открывая файлы, фотографии и первый кадр видеофайлов. Кроме того, можно выбрать фиксированные представления **Список, Таблица** и **Содержимое**.

Область переходов содержит следующие группы ссылок: ***Избранное, Библиотеки, <Пользователь>, Компьютер*** и ***Сеть***(см. рисунок 3.13). Традиционные папки и ее содержимое находятся в определенном месте на жестком диске. В противоположность этому ***Библиотеки*** содержат список ссылок на папки и файлы, которые могут храниться в разных местах.

Область просмотра. Проводники документов, музыки и изображений также имеют дополнительную область просмотра, предназначенную для более подробного ознакомления с содержимым документов и файлов мультимедиа без открытия каждого файла в отдельности (см. рисунки 3.11). Приложения, поддерживающие данную функцию, позволяют просматривать различные документы в удобном для чтения виде или воспроизводить несколько секунд содержимого файлов мультимедиа. Данная функция подобна предварительному просмотру сообщений в области просмотра **Microsoft Office Outlook**.

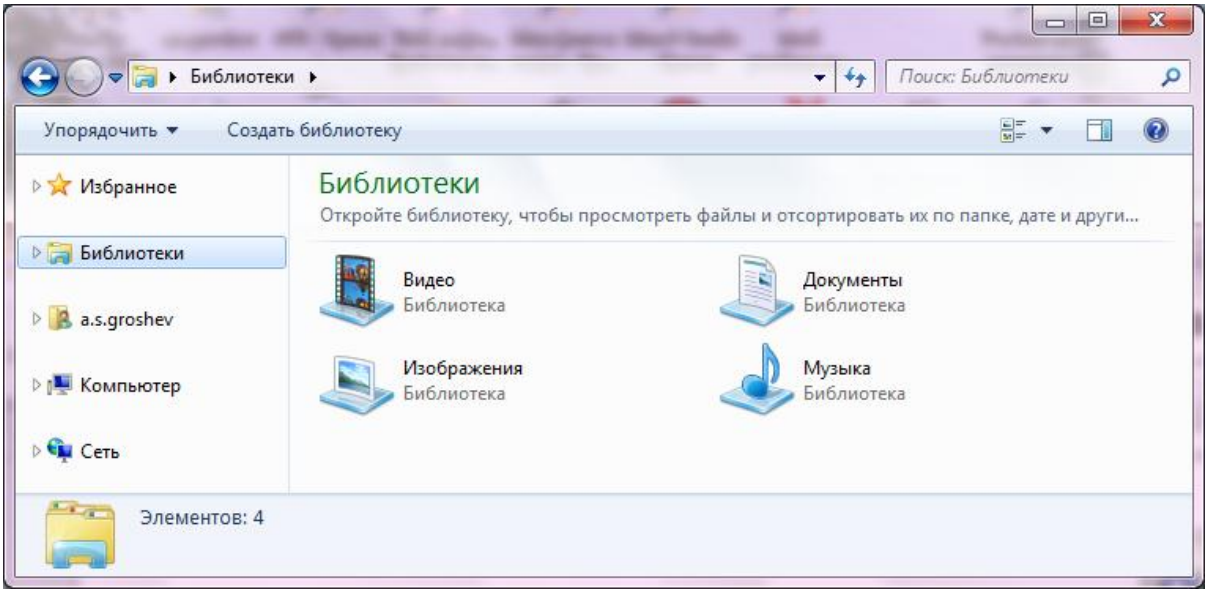


Рисунок 3.13. Окно Проводника и область переходов

Адресная строка, заголовки границы окна. Улучшенная адресная строка снабжена раскрывающимися меню для каждого из ее пунктов (см. рисунок 3.14). Они позволяют осуществлять простой переход назад или вперед в любое место из текущего положения. Такая адресная строка особенно удобна на планшетных ПК. С ее помощью можно переходить точно в нужное место несколькими касаниями пера.

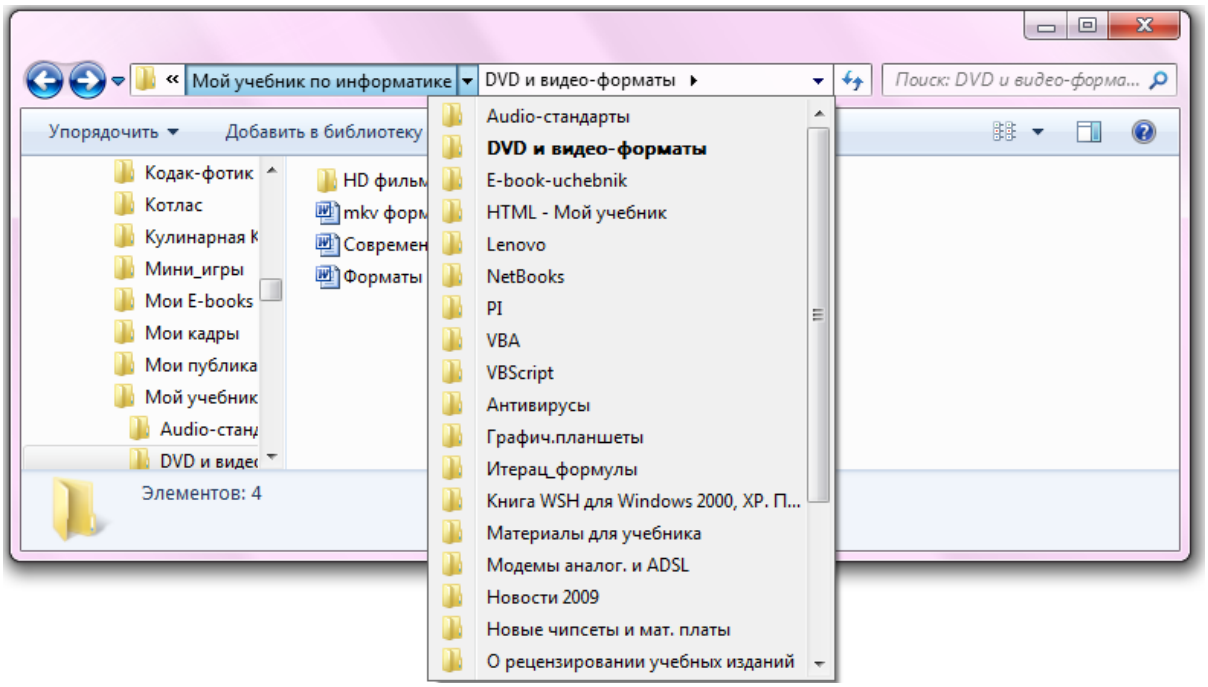


Рисунок 3.14. Адресная строка с раскрывающимся списком папок

Элементы управления заголовками столбцов в представлении ***Таблица*** позволяют лучше управлять большим количеством файлов в **Проводнике**. Раскрывающиеся списки показывают все значения любого из столбцов для всех файлов, отображаемых в представлении. Например, если выбрать заголовок столбца **Тип**, можно увидеть все типы файлов в данном окне (см. рисунок 3.15). При помощи простых флажков пользователь может выбрать один или несколько типов файлов. Например, при выборе типов **PPT** и XLS скрываются документы всех других типов, и в окне остаются только файлы **Microsoft Office PowerPoint** и **Excel**.

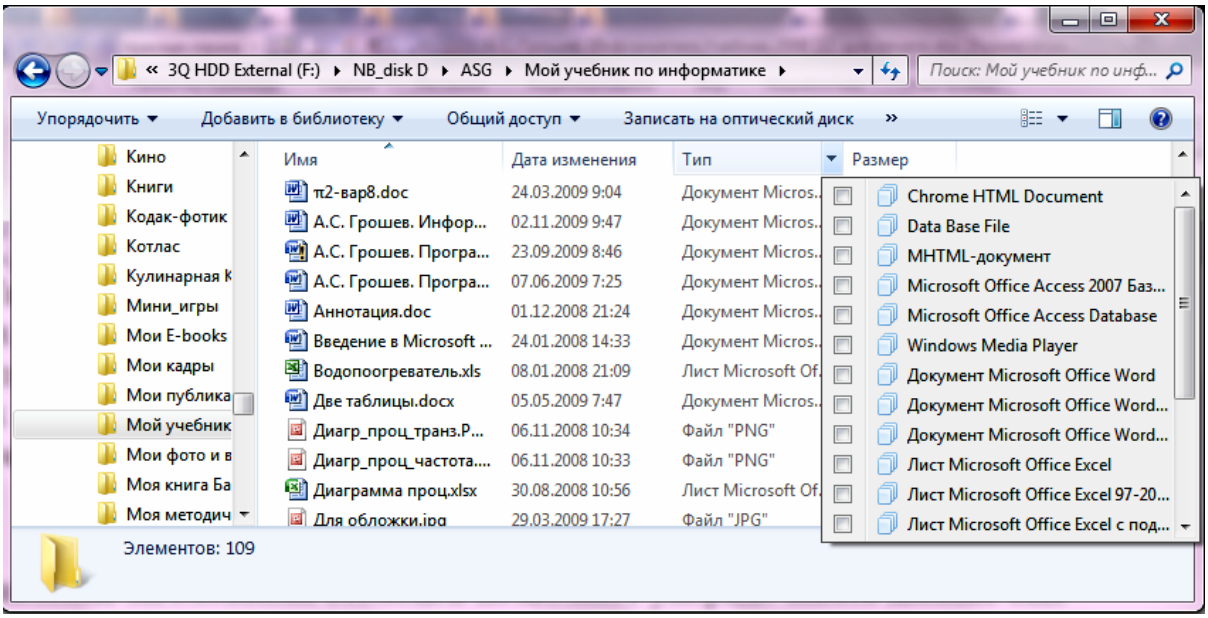


Рисунок 3.15. Раскрывающийся список типов файлов в папке

Библиотеки Windows 7

В операционной системе Windows 7 появился новый компонент - библиотеки.

Библиотека – упорядоченная информация о файлах определенного типа.

В операционной системе Windows 7 имеются четыре библиотеки по умолчанию: «Документы», «Музыка», «Изображения» и «Видео». Каждая библиотека оптимизирована для отдельного типа файлов (например, для музыки или изображений). Оптимизация библиотеки для определенного типа файлов изменяет набор параметров, доступных для упорядочивания файлов, например, для музыки: папка, альбом, исполнитель, композиция, жанр, оценка (см. рисунок 3.16). Просмотр файлов в библиотеке осущест-

вляется так же, как в папке. Можно создавать новые библиотеки (ранее см. рисунок 3.8).

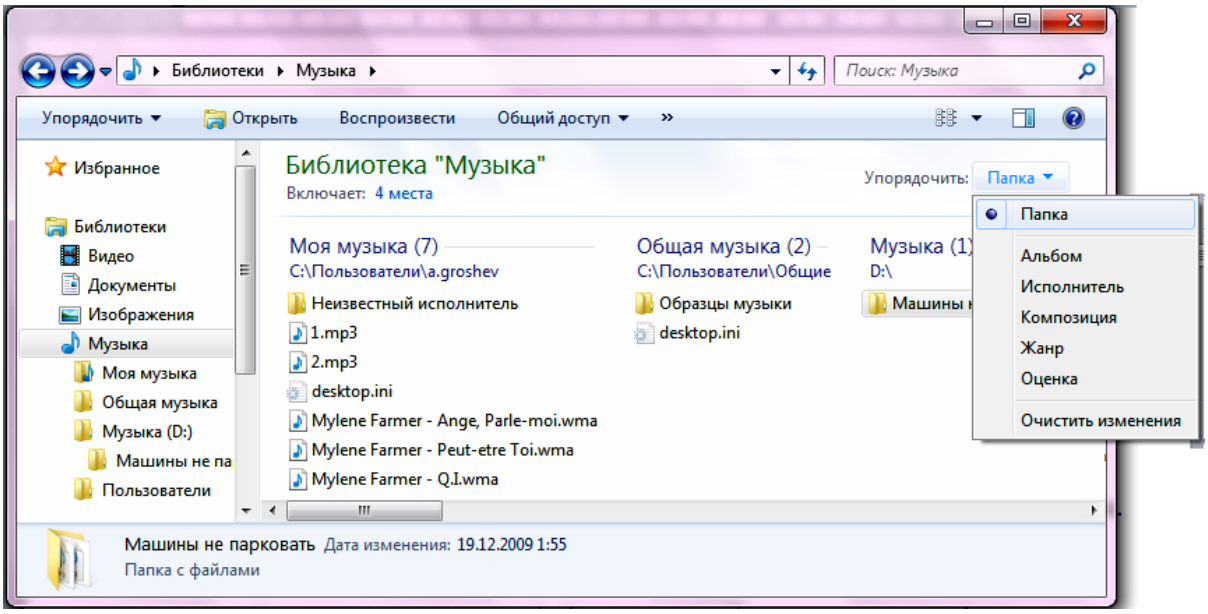


Рисунок 3.16. Библиотека «Музыка» и виды упорядочения

В отличие от папки, библиотека отображает в списке файлы, которые хранятся в нескольких расположениях. В библиотеках не хранятся объекты. Библиотеки содержит ссылки и позволяют просматривать и упорядочивать объекты библиотеки различными способами. Например, если у пользователя имеются музыкальные файлы в папках на жестком диске и на внешнем диске, можно организовать доступ ко всем музыкальным файлам с помощью библиотеки «Музыка».

Технология ClearType

Большую часть времени работы с компьютером люди тратят на чтение документов, электронной почты и просмотр веб-страниц. Благодаря патентованной технологии **Microsoft ClearType**, используемой в **Windows Vista**, читать текст на экране стало удобнее.

Развитию **ClearType** способствовал переход от мониторов на электронно-лучевых трубках (**ЭЛТ**) к жидкокристаллическим экранам (**ЖК**).

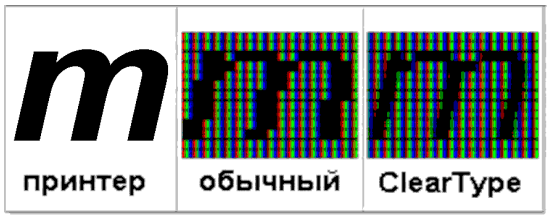


Рисунок 3.17. Технология **ClearType**

ClearType была впервые использована в операционной системе **Windows XP** (см. рисунок 3.17). Несмотря на то, что эта функция позво-

ляла несколько улучшить читаемость текста для пользователей **ЭЛТ**-экранов, по умолчанию она была выключена. Причиной тому было малое распространение **ЖК**-экранов, на которых преимущества **ClearType** особенно заметны. В **Windows Vista** функция **ClearType** включена по умолчанию для всех пользователей вне зависимости от типа экрана. Для полного использования возможностей **ClearType** в **Windows Vista** добавлены новые шрифты, специально созданные для работы с данной функцией: **Constantia, Cambria, Corbel, Calibri, Candara, Consolas** и др. (см. таблицу 3.7).

Таблица 3.7. Новые шрифты **ClearType** в **Windows Vista**

Шрифт	Области применения	Образец текста
Constantia	Для набора текста, электронной почты, журналов и веб-страниц	В **Vista** новые шрифты
Calibri	- " -	В **Vista** новые шрифты
Cambria	Для деловых документов, электронной почты и веб-страниц	В **Vista** новые шрифты
Corbel	- " -	В **Vista** новые шрифты
Candara	Доя электронной почты, журналов, веб-страниц и пр.	В **Vista** новые шрифты
Consolas	Для программных сред	В Vista новые шрифты
Segoe Script	Новый рукописный шрифт	*В Vista новые шрифты*

Диалоговые окна и Мастера

Диалоговые окна и **Мастера** в были переработаны и стали более понятными и функциональными. Варианты выбора теперь снабжены полезными пояснениями, шрифты стали крупнее, а внешний вид и цвет графических элементов изменились. Новые диалоговые окна и **Мастера** позволяют находить нужную информацию быстрее и более точно. Теперь они имеют больше общего с интерфейсами других компонентов. Это, например, относится к использованию таких возможностей перехода в стиле веб-обозревателя, как кнопки **Вперед** и **Назад**. Стандартные функции диалоговых окон приобрели большую гибкость по сравнению с диалоговыми окнами **Windows XP**, что облегчает их интеграцию в приложения (см. пример на рисунке 3.18).

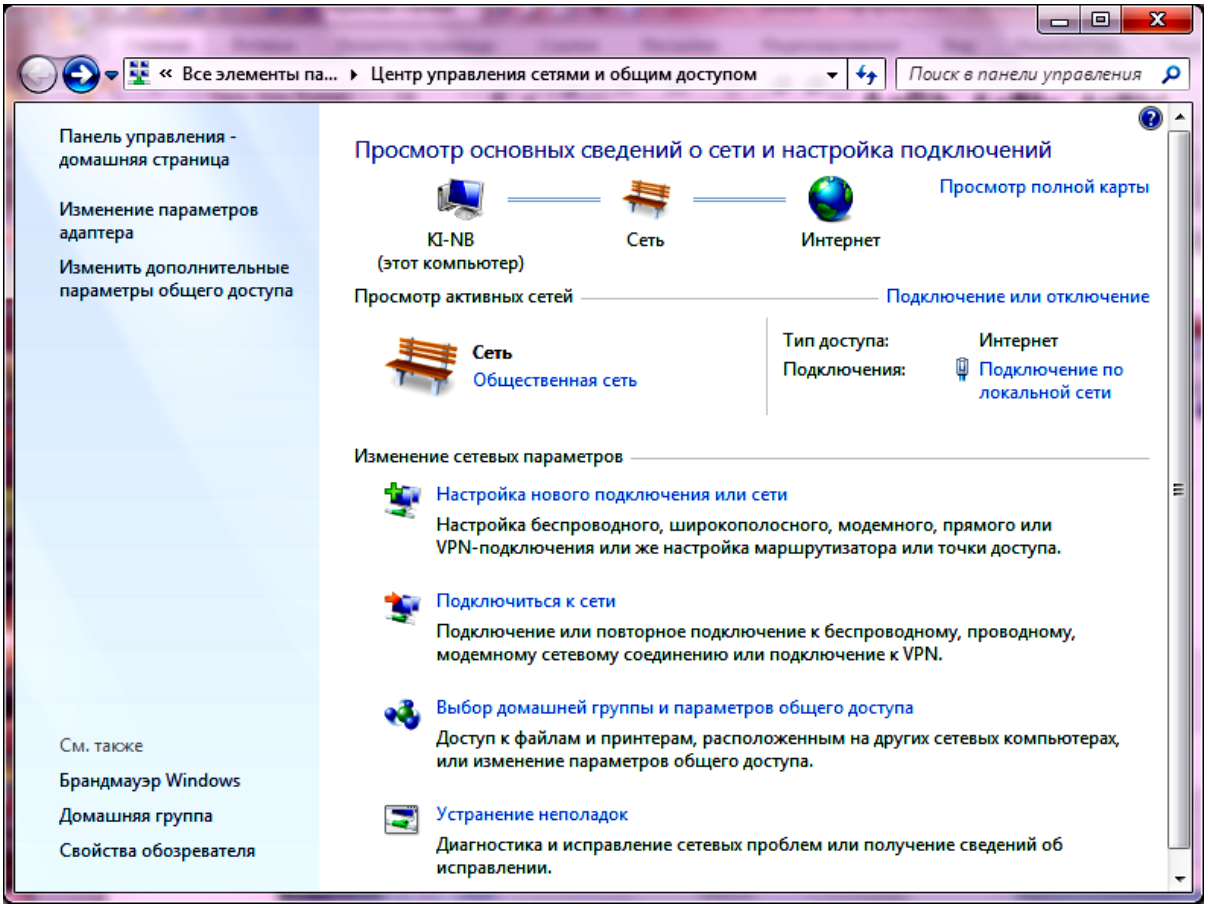


Рисунок 3.18. Центр управления сетями и общим доступом

Мастера в **Windows** стали удобнее в использовании и проще в разработке. Улучшен доступ ко всем элементам, в том числе графическим объектам окон. Благодаря этому разработчики могут создавать единообразные, понятные и эффективные **Мастера**, способные лучше помогать пользователям. Например, разработчики могут изменять набор используемых кнопок в зависимости от последовательности действий пользователя (набор кнопок больше не ограничен вариантами **Назад, Далее, Готово** и **Отмена**). Размер окон **Мастера** можно изменять, чтобы предоставить пользователю более подробную информацию при использовании различных сценариев. В оформлении окон **Мастера** может применяться фирменная символика с целью обеспечения унификации с интерфейсом приложений сторонних производителей.

Панель управления

П**анель управления** также построена на основе **Проводника** и имеет три вида просмотра (см. рисунок 3.19), кроме того строка поиска позволяет быстро отобрать задачи, необходимые для настройками ПК (напри-

мер, чтобы найти действие *Настройка разрешения экрана*, достаточно ввести слово монитор). Удобнее всего использовать режим с поиском.

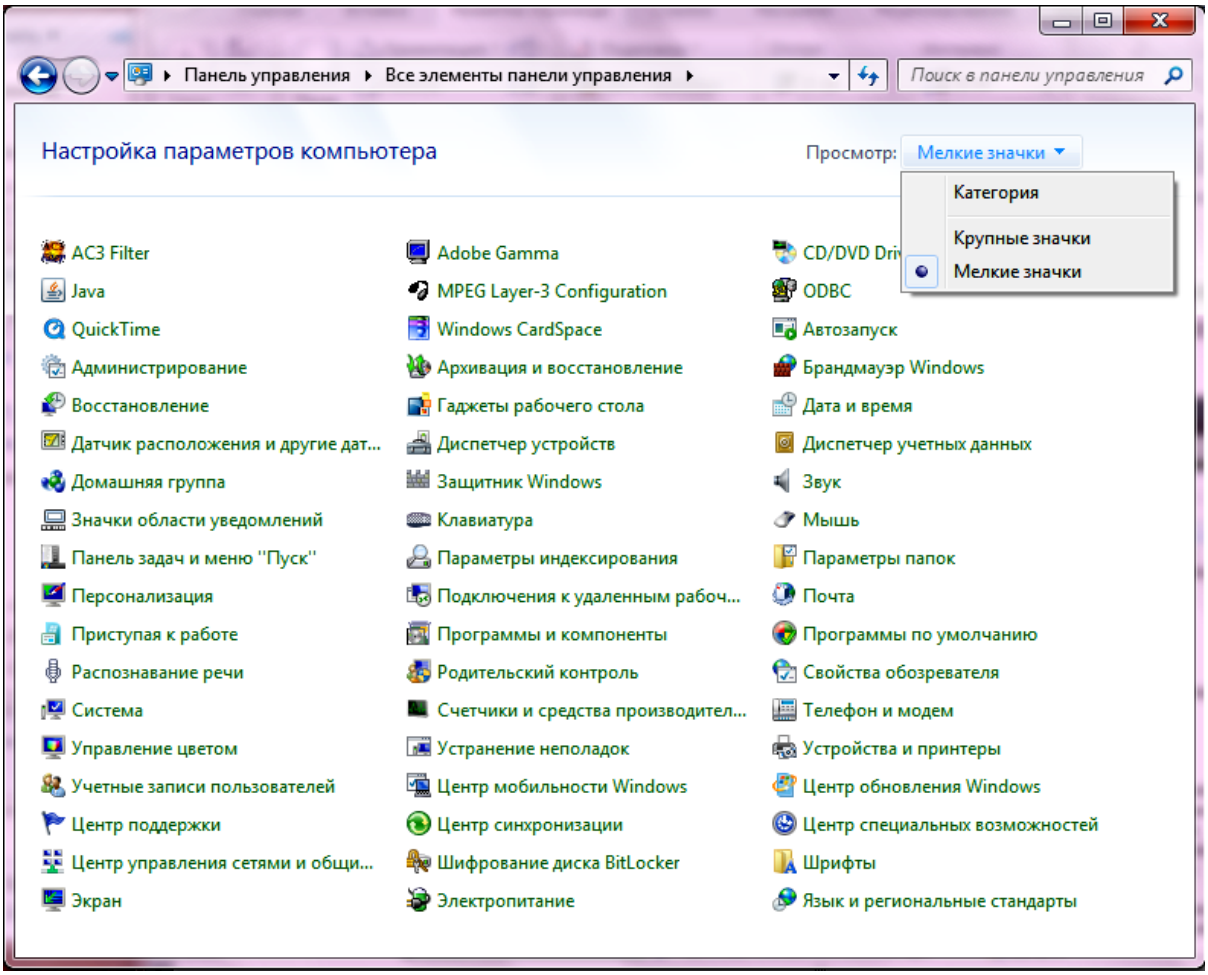


Рисунок 3.19. Панель управления

Первое представление – ***Категории*** – является усовершенствованным вариантом представления по категориям, впервые появившегося в **Windows XP**.

Второй и третий вид просмотра – классические представления крупных и мелких значков программ настройки системы. Средство быстрого поиска также работает и в этом случае.

Метаданные файлов

Новые средства **Windows** для поиска и упорядочения файлов широко используют свойства файлов (метаданные). Многие файлы уже содержат метаданные, готовые к использованию в **Windows**. Система **Microsoft Office** автоматически сохраняет определенные свойства создаваемых документов, например имя автора и дату создания. Музыкальные файлы час-

то имеют такие свойства, как название песни, альбома и имя исполнителя. Изменить метаданные файла или группы файлов можно с использованием ***Области сведений***, окна свойств или окна сохранения файла.

• ***Область сведений***. Самый простой способ добавить свойства файла – выбрать этот файл и изменить его свойства в области сведений, расположенной в нижней части **Проводника**. Многие поля ввода поддерживают автозаполнение. Данный способ можно применять как для одного, так и для нескольких файлов. Если выбрать несколько файлов и добавить какое-либо свойство в области сведений, оно будет применяться для всех выбранных файлов.

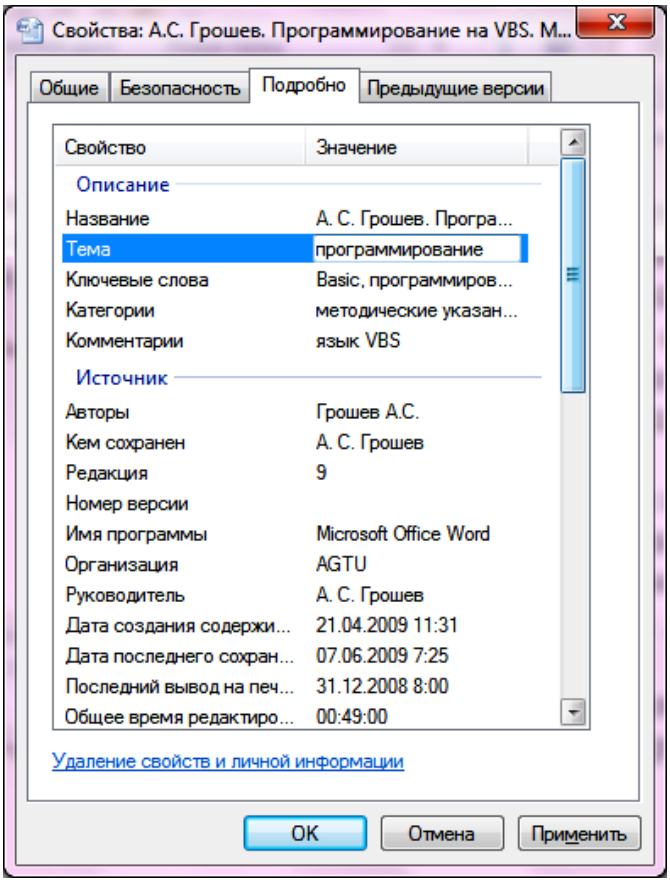


Рисунок 3.20. Свойства файла

• **Окно свойств**, как обычно, открывается с использованием контекстного меню файла. Вкладка ***Подробно*** позволяет просматривать и редактировать большой список метаданных файла. На этой же вкладке можно удалить все свойства файла командой *Удаление свойств и личной информации* (см. рисунок 3.20).

• **Окно сохранения файла** подверглось значительной модификации при сохранении можно редактировать значительное количество метаданных файла (см. рисунок 3.21).

Боковая панель Windows использовалась в **Windows Vista** (располагалась справа рабочего стола и служила для размещения на ней (или на Рабочем столе) мини-приложений. В **Windows 7** мини-приложения (**Гаджеты**) размещаются на **Рабочем столе** (см. ранее рисунок 3.6, 3.9).

Мини-приложения используются для самых разных целей. Они могут соединяться с веб-службами и доставлять деловую информацию, прогноз погоды, свежие новости, схемы дорожного движения, воспроизводить интернет-радио и показывать слайды из фотоальбомов на компьютере или в Интернете. Мини-приложения могут быть встроены в приложе-

ния для упрощения взаимодействия с ними. Например, при помощи мини-приложений пользователь может одновременно увидеть все контакты служб мгновенных сообщений, ознакомиться с планами на день в календаре или с управлять проигрывателем мультимедиа. Кроме того, они могут использоваться отдельно в различных качествах – например, как калькуляторы, игры или записные книжки.

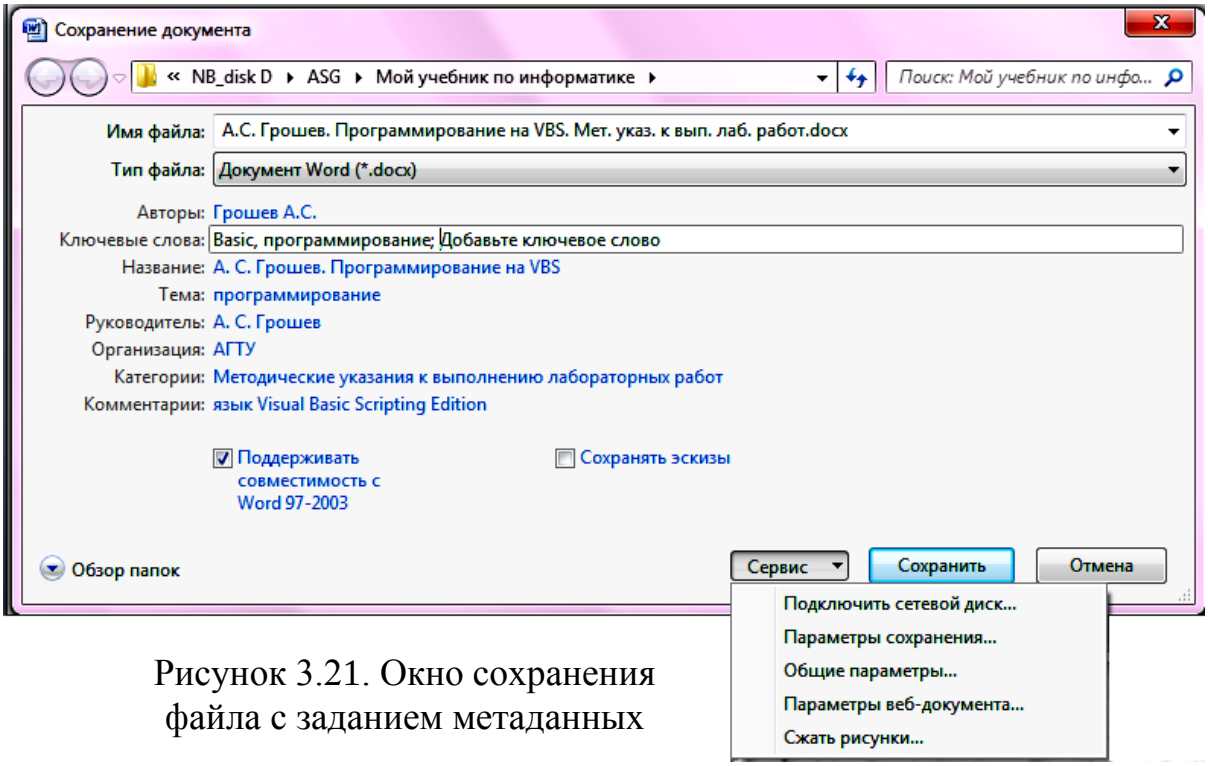


Рисунок 3.21. Окно сохранения файла с заданием метаданных

В состав **Windows** входят основные мини-приложения, необходимые для начала работы. Дополнительные мини-приложения можно загрузить из веб-коллекции мини-приложений. Эта коллекция предлагает широкий выбор мини-приложений для выполнения самых разных задач. Она содержит мини-приложения, предоставленные различными разработчиками и издателями (см. рисунок 3.22).

Internet Explorer 7 и *8*

Internet Explorer 7 – веб-обозреватель, который входит в состав **Microsoft Windows Vista**. **Internet Explorer 8** может поставляться с **Windows 7** или отдельно.

Усовершенствования браузера относятся к трем областям.

1. Упрощение выполнения повседневных задач. Усовершенствованные средства переходов, вкладки и возможность поиска в Интернете непосредственно с панели команд облегчают выполнение повседневных за-

дач. Расширенные возможности печати, упрощенные средства поиска и чтения RSS-каналов и подписки на них позволяют получать оперативный доступ к нужной пользователю информации.

Рисунок 3.22. Мини-приложения (гаджеты)

2. Динамическое обеспечение безопасности. Новая надежная архитектура **Internet Explorer'a** включает средства безопасности, которые позволяют защититься от вредоносных программ. Она также помогает защитить пользователей от непреднамеренного предоставления личных данных на мошеннические веб-узлы, прибегающие к различным формам обмана, например фишингу.
3. Улучшенная платформа для разработки веб-узлов и управления ими. **Internet Explorer 7** и **8** обеспечивают улучшенную поддержку таблиц стилей (CSS), а также предоставляют надежные инструменты для развертывания обозревателя и управления им в условиях крупных организаций.

Элементы интерфейса пользователя стали значительно проще и удобнее, в нем гораздо меньше ненужных элементов. В правом верхнем углу окна находится поле быстрого поиска (см рисунок 3.23).

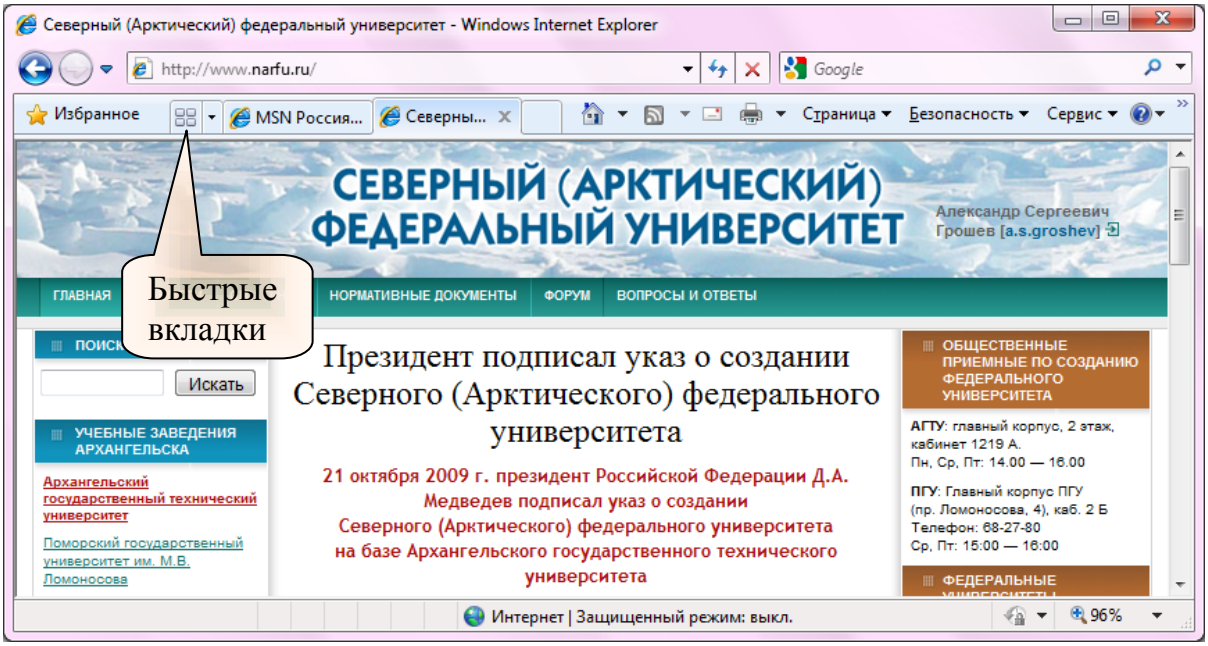


Рисунок 3.23. Веб-обозреватель **Internet Explorer 8**

Просмотр веб-страниц с использованием вкладок позволяет открыть несколько веб-узлов в одном окне просмотра. Для создания или открытия вкладок достаточно щелкнуть свободную вкладку на панели инструментов. Можно также щелкнуть правой кнопкой мыши гиперссылку на веб-странице и выбрать пункт ***Создать вкладку***. Щелкнув вкладку правой кнопкой мыши, можно обновить каждую страницу по отдельности или все страницы в группе, а также закрыть отдельные вкладки или всю группу. Кроме того, перетаскивая вкладки, можно изменять их порядок на панели вкладок.

Internet Explorer 7 и **8** позволяет управлять множеством вкладок при помощи функции под названием **Быстрые вкладки**. С помощью этой функции можно одновременно увидеть эскизы всех открытых вкладок (см. рисунок 3.24). Чтобы просмотреть все открытые вкладки, достаточно просто щелкнуть значок быстрых вкладок, расположенный справа от значка **Избранное**. В представлении **Быстрые вкладки** можно открыть любую вкладку, щелкнув ее изображение. Чтобы закрыть вкладку, следует щелкнуть значок × в правом верхнем углу изображения. Масштаб страницы быстрых вкладок зависит от числа открытых вкладок. Если число открытых вкладок больше 20 (что не рекомендуется), эскизы будут показаны в меньшем масштабе, но пользователь по-прежнему сможет увидеть все эскизы в одном окне.

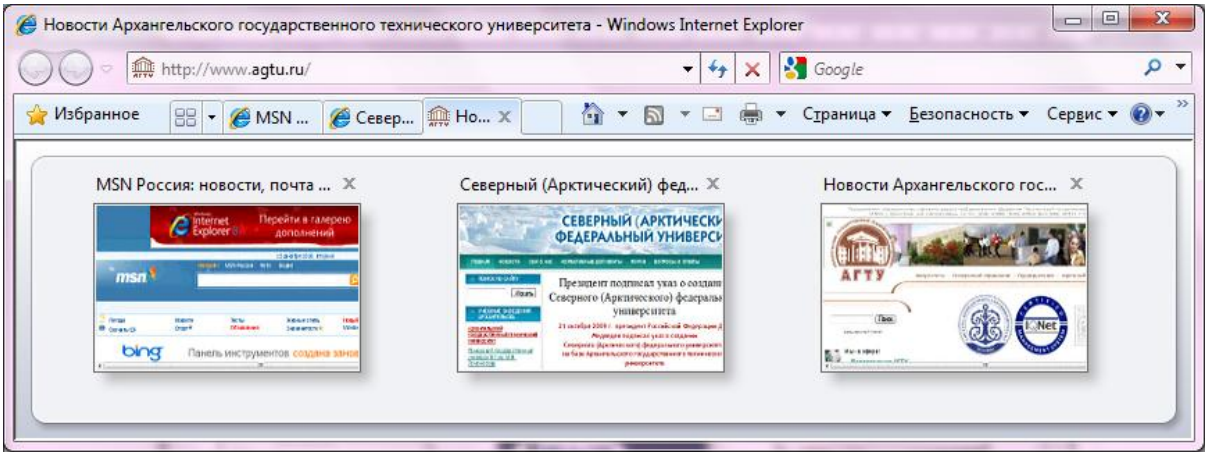


Рисунок 3.24. Режим **Быстрые вкладки** веб-обозревателя

Несколько вкладок можно объединить в одну группу, а затем сохранить ее в избранном. Вкладки можно группировать по темам или категориям, например покупки, финансы или погода.

Средство быстрого поиска – поле *Поиск в Интернете* – позволяет быстро выполнять поиск в **Интернете** непосредственно из рамки обозревателя. При этом используется служба поиска, выбранная пользователем. Можно выбирать службы поиска из раскрывающегося списка, а также добавлять в список новые службы.

Раскрывающийся список средства быстрого поиска включает специализированные и общие службы поиска. В списке поля *Поиск в Интернете* команда ***Найти другие службы*** позволяет добавить в список множество общих и специализированных служб поиска.

Усовершенствования в области печати. По умолчанию **Internet Explorer** сжимает текст веб-страницы ровно настолько, чтобы обеспечить печать всей страницы. Можно задать масштаб печати, изменять поля страницы, удалять колонтитулы и увеличивать или уменьшать область печати. При желании можно напечатать веб-страницу без графических изображений или рекламы. Достаточно просто выделить (при помощи мыши) текст и изображения, которые нужно напечатать, и выбрать команду **Предварительный просмотр**. Эта функция показывает только выделенную область и печатает ее соответствующим образом.

Функция изменения масштаба позволяет увеличивать или уменьшать размер страницы для улучшения просмотра. Для изменения масштаба достаточно покрутить колесико мышки при нажатой клавише CTRL. Изменяется не только размер шрифта, но и размер изображений и текста в изображениях.

Internet Explorer имеет встроенную функцию поддержки RSS-каналов (новостные, торговые, спортивные и пр. каналы), благодаря чему пользователи могут легко осуществлять поиск RSS-каналов, подписываться на них и читать их содержимое непосредственно в обозревателе.

Динамическое обеспечение безопасности решает две основные задачи обеспечения безопасности:

1. Защита от вредоносных программ. Корпорация Microsoft поставила себе цель обеспечить пользователям безопасность действий при просмотре веб-узлов и предотвратить установку вредоносных и других нежелательных программ. Вредоносной программой по определению Microsoft является любой вредоносный код или нежелательное программное обеспечение, включая программы-черви, вирусы, программы для показа рекламы и программы-шпионы.
2. Обеспечение безопасности личных данных. Обозреватель **Internet Explorer** помогает защитить пользователей от фишинга и предотвратить хищение данных пользователей мошенническими веб-узлами. Он позволяет повысить безопасность и надежность электронных коммерческих операций, не допуская непреднамеренного разглашения личных данных.

Защита от атак с применением междоменных сценариев. При атаках с применением междоменных сценариев сценарий из одного домена Интернета управляет информацией другого домена. Для защиты от такого вида мошенничества обозреватель **Internet Explorer** учитывает доменное имя, с которого был получен каждый сценарий, и позволяет сценарию взаимодействовать только с окнами и содержимым из того же домена. Благодаря таким междоменным барьерам информация пользователя доступна только тем узлам, которым пользователь намеревается ее предоставить. Эта новая функция усиливает защиту от вредоносных программ за счет ограничения возможностей мошеннических веб-узлов по использованию ошибок в других узлах и загрузки нежелательного содержимого на ПК пользователя.

Исправление параметров безопасности – в обозревателе **Internet Explorer** стандартные параметры безопасности обеспечивают максимально удобную работу, одновременно поддерживая требуемый уровень безопасности. Однако для работы некоторых приложений необходимо снизить уровень безопасности по сравнению со стандартным. В таких случаях важно восстановить прежние значения, когда измененные параметры больше не используются.

Совместное использование **Internet Explorer** и **Защитника Windows** повышает уровень безопасности и защиты личных сведений. **Защитник Windows** усиливает защиту от вредоносных программ на уровне обозревателя и не допускает проникновения вредоносных программ на компьютер путем комбинированной загрузки, которая часто используется для распространения шпионских программ и их незаметной установки вместе с другими приложениями.

Защита от фишинга**.** Владельцы мошеннических веб-узлов используют технологию, известную как фишинг, заключающуюся в имитации надежных лиц или организаций в Интернете с целью получения личной информации пользователей. В 2006 году было зафиксировано рекордное количество случаев фишинга. Кража удостоверяющих сведений представляет значительную угрозу финансовой безопасности пользователей.

В отличие от прямых атак, когда злоумышленники взламывают систему, чтобы получить сведения о счете, для фишинга не требуются сложные технические средства. Такие атаки становятся возможными в результате неосознанных действий пользователей, которые с готовностью сообщают личные сведения, например пароли к банковскому счету. Чтобы предотвратить такие атаки, пользователи должны знать о проблеме и понимать ее суть, а не просто ограничиться установкой обновлений для приложения. Качественно сделанные поддельные веб-узлы могут обмануть даже опытных профессионалов. Злоумышленники становятся все более изощренными и реагируют все быстрее, стремясь избежать обнаружения.

В последние годы для защиты информации пользователей стали использоваться технологии шифрования связи и протокол **SSL**. С появлением значительного количества веб-узлов физических лиц и малых предприятий, продающих товары в широком диапазоне цен, пользователи стали чаще сталкиваться с неизвестными организациями, которые просят сообщить им финансовые сведения. Сочетание этих факторов создает обширное поле деятельности для злоумышленников. Для решения этой проблемы в **Internet Explorer** используются понятные и хорошо заметные цветные значки, информирующие пользователей о безопасности и надежности веб-узла.

В строке состояния безопасности отображаются сведения о сертификатах SSL высокой надежности (**НА**), которые используются для более надежной идентификации безопасных узлов, например веб-узлов банков. Сертификат SSL высокой надежности указывает, что центр сертификации провел всестороннюю проверку такого веб-узла. При работе с узлами, прошедшими такую проверку, цвет адресной строки в обозревателе

Internet Explorer изменяется на зеленый и отображается название соответствующей организации.

Если **Internet Explorer** обнаруживает какие-либо ошибки в сведениях сертификата узла, цвет фона меняется на красный. Пользователи получают дополнительную наглядную информацию для распознавания сомнительных веб-узлов. Для веб-узлов с правильными учетными данными, напротив, в адресной строке отображается имя владельца сертификата на золотом фоне, что указывает на более высокий уровень безопасности и надежности.

Фильтр фишинга **Internet Explorer** обращается к интерактивной информационной службе, которая обновляется несколько раз в час. Служба располагает последними сведениями о мошеннических веб-узлах и предупреждает пользователей о любых подозрительных веб-узлах. Принцип работы данного фильтра основан на динамическом получении и частом обновлении информации.

Фильтр фишинга сочетает средства поиска признаков подозрительных веб-узлов на клиентском компьютере с обращением к дополнительной интерактивной службе. В фильтре предусмотрены три способа защиты от мошенников:

1. сравнение адресов веб-узлов, на которые пользователь пытается зайти, со списком надежных узлов, который хранится на компьютере пользователя;
2. анализ веб-узлов, которые хочет посетить пользователь, с целью выявления типичных признаков мошеннических узлов;
3. отправка адреса (без указания каких-либо личных сведений) веб-узла, на который хочет зайти пользователь, в интерактивную службу Майкрософт для немедленной проверки в регулярно обновляемом списке мошеннических узлов.

В случае поучения подтверждения того, что адрес принадлежит известному мошенническому узлу, обозреватель **Internet Explorer** оповещает об опасности (красным цветом) и автоматически направляет пользователя на нейтральную страницу с предупреждением. Если веб-узел не относится к числу известных мошеннических узлов, но совершает действия, свойственные таким узлам, адресная строка окрашивается в желтый цвет и предупреждает пользователя о подозрительном поведении. Пользователь может сообщать о мошеннических веб-узлах и ложных срабатываниях фильтра фишинга непосредственно из обозревателя. Фильтр фишинга в только что установленном обозревателе не включен и не отключен. При первом после установки запуске обозревателя пользователь должен вклю-

чить или отключить этот фильтр. Для включения или отключения фильтра фишинга достаточно одного щелчка мыши.

Защита отображения адресов URL. Злоумышленники часто пытаются вызвать у пользователей ощущение, что они имеют дело с информацией из известного и надежного источника. Возможность скрывать от пользователей сведения об истинном адресе URL и доменных именах служила ценным подарком для злоумышленников. Два мощных визуальных инструмента **Internet Explorer** помогают пользователям избежать обмана: адресная строка в каждом окне и поддержка многоязычных доменных имен (IDN).

• Адресная строка в каждом окне. Во всех окнах обозревателя отображается адресная строка. Злоумышленники часто пользуются допустимыми действиями со всплывающими окнами для отображения изображений и данных, обманным путем убеждающих пользователей загрузить и установить вредоносные программы. Поэтому наличие в каждом окне адресной строки, доступной только для чтения, позволяет пользователям получить сведения об истинном источнике той информации, которую они видят.

• Отображение многоязычных доменных имен. Интернет охватывает все мировое сообщество, и поэтому обозреватели должны уметь работать не только с латинскими символами и доменными именами. Операторы мошеннических веб-узлов использовали проблемы с отображением символов на различных языках в качестве механизма для совершения фишинговых атак против пользователей. Также этим способом можно было скрыть настоящее доменное имя веб-узла. Эта проблема возникает вследствие использования различных наборов символов. В некоторых языках многие символы (например, английская буква а) могут напоминать совершенно иные символы других языков (например, букву а в кириллице). В результате злоумышленник мог зарегистрировать доменное имя, похожее на имя существующего надежного узла, чтобы обманным путем вынуждать пользователей отправлять свои данные на поддельный узел. Предыдущие версии Internet Explorer не имели поддержки многоязычных доменных имен и были уязвимы для такого вида атак.

Новый **Internet Explorer** полностью поддерживает возможности и отображение доменных имен, содержащих символы различных языков. Если какая-либо часть доменного имени содержит символы из различных наборов, правила использования IDN в **Internet Explorer 7** автоматически включают отображение доменного имени в кодировке Punycode. При этом учитываются языковые параметры операционной системы. Только симво-

лы из наборов символов, выбранных в системе языков, отображаются в своем естественном виде. Определенные сочетания языков, например сочетание английского с китайским, всегда отображаются в кодировке Punycode. Такие сочетания не могут использоваться в одном компоненте доменного имени, потому что эти языки используют разные наборы символов. Другие сочетания языков, например французского с английским, могут отображаться как в кодировке Punycode, так и в своем естественном виде. Это зависит от того, какие языки разрешены в операционной системе пользователя. Например, если включена поддержка и французского, и английского языков, адрес URL www.microsôft.com будет правильно отображаться в адресной строке, потому что оба эти языка используют общий набор основных символов. Однако если один из языков не поддерживается, адрес будет отображен в кодировке Punycode, поскольку в одном компоненте доменного имени используются и французские, и английские символы. При этом в адресной строке будет отображен адрес *www.xn--microsft-03a.com*. Таким образом, пользователь будет предупрежден о подозрительном адресе URL.

Удаление из обозревателя всех личных сведений крайне важно для сохранения конфиденциальности в Интернете. Особенно это относится к компьютерам, с которыми работают многие пользователи. В **Internet Explorer** предусмотрена функция удаления журнала обозревателя, которая позволяет пользователям удалять все личные сведения одним щелчком мыши.

Родительский контроль. **Internet Explorer** непосредственно взаимодействует со службой родительского контроля. Это обеспечивает удобный доступ к сведениям журнала и единый интерфейс управления настройками. Служба родительского контроля также может блокировать загрузку файлов – еще один способ предотвратить проникновение вредоносных программ на компьютер.

Улучшение возможностей администрирования. При использовании **Internet Explorer 7** и **8** администраторы получают возможность централизованного управления настройками на основе групповой политики службы каталогов **Microsoft Active Directory**, что значительно облегчает управление в организации. Обновленный пакет администрирования дает возможность развертывать полностью настроенные копии обозревателя.

Защитник Windows

Защитник Windows осуществляет непрерывный мониторинг системных папок, отслеживая изменения, которые свидетельствуют о присут-

ствии программ-шпионов, и, помогая точно оценить степень безопасности ПО, установленного на компьютере пользователя. По умолчанию **Защитник Windows** проверяет компьютер на наличие программ-шпионов ежедневно в 2:00, если не настроено другое расписание. В процессе проверки, в зависимости от настроек пользователя, могут автоматически выполняться действия над объектами высокой, средней или низкой важности.

Кроме того, нажав кнопку сканирования, можно в любое время быстро проверить папки, которые заражаются чаще всего (например, с программными файлами или объектами модуля поддержки Internet Explorer). При быстром поиске обычно выявляются наиболее распространенные программы-шпионы. Полное сканирование системы занимает больше времени, но при его выполнении на наличие признаков присутствия программ-шпионов проверяется весь компьютер и применяется более полный набор описаний. В раскрывающемся списке можно самостоятельно выбрать для проверки конкретные папки и диски, например съемный носитель.

Всего существует четыре возможных действия при обнаружении угрозы.

1. **Пропустить.** Никакие действия не выполняются, но потенциальная угроза будет выявляться при проведении проверок в будущем.
2. **Отправить на карантин.** В безопасном месте создается архивная копия программы, а сама она удаляется. Запустить программу нельзя, но в случае необходимости ее можно восстановить.
3. **Удалить.** Полное удаление программы с компьютера.
4. **Пропускать всегда.** Программа добавляется в список разрешенных объектов и не будет обнаруживаться при проведении проверок в будущем.

Защитник Windows выполняет поиск и удаление вредоносного программного обеспечения даже в том случае, когда пользователь не имеет полномочий администратора. По умолчанию обычный пользователь может удалять, отправлять на карантин и пропускать обнаруженные программы.

При сканировании удаляются только известные программы-шпионы. Для борьбы с новыми и неизвестными угрозами в состав **Защитник**а **Windows** включены агенты мониторинга, обеспечивающие защиту в режиме реального времени. Несколько агентов безопасности осуществляют мониторинг важных компонентов компьютера, которые могут быть изменены программами-шпионами: папка автозагрузки, системные настройки, надстройки Internet Explorer, параметры настройки Internet

Explorer, загружаемые файлы Internet Explorer, службы и драйверы, параметры выполнения и регистрации приложений, а также надстройки **Windows**. Через эти компоненты программы-шпионы чаще всего проникают на компьютер.

Поскольку программы-шпионы непрерывно совершенствуются, **Защитник Windows** перед выполнением запланированной проверки автоматически ищет самые последние описания. Обновленные описания создаются аналитиками корпорации Майкрософт при поддержке сообщества активных пользователей **Защитник**а **Windows**, присылающих отчеты о программах-шпионах.

Обнаружив в ходе запланированной проверки компьютера подозрительные операции или присутствие известной программы-шпиона, **Защитник Windows** оповещает об этом пользователя. Форма оповещения зависит от уровня серьезности потенциальной угрозы. Если обнаруженные изменения вызваны программой, которая еще не была классифицирована, на панели задач появляется небольшое уведомление. В случае обнаружения нежелательного или вредоносного программного обеспечения появляется желтое или красное окно оповещения, поскольку требуется немедленное реагирование (желтое окно отображается для нежелательных программ, красное – для вредоносных, а также при одновременном обнаружении нежелательных и вредоносных программ).

Архивация и восстановление данных

Несмотря на то, что архивация основных файлов и системной информации является чрезвычайно важной для обеспечения безопасности, не все пользователи находят время для выполнения этой операции. Как показали исследования, пользователям нужны более удобные способы выбора файлов и папок для архивации, а также встроенная поддержка распространенных резервных носителей, таких как CD- и DVD-диски.

Средство архивации и восстановления данных **Windows** с успехом решает эти задачи и позволяет пользователям выбирать файлы для архивации. А благодаря поддержке работы с CD- и DVD-дисками архивировать и восстанавливать важные файлы стало еще удобнее. Пользователи могут восстанавливать файлы с одного CD- или DVD-диска или набора дисков, даже если доступны не все входящие в архив диски.

Все функции архивации восстановления объединены в одной унифицированной панели архивации и восстановления, называемой центром архивации и восстановления.

Теневое копирование. Случайное удаление или изменение файла является распространенной причиной потери данных. В состав **Windows** входит новая полезная функция теневого копирования, которая помогает избежать потери данных. Эта функция, доступная в выпусках Business, Enterprise и Ultimate, автоматически создает текущие копии файлов, с которыми работает пользователь, что позволяет быстро и просто восстанавливать предыдущие версии случайно удаленного документа.

Функция теневого копирования **Windows** включается автоматически и периодически создает копии измененных файлов. Для хранения теневых копий используется минимум места на жестком диске, поскольку сохраняются только новые изменения. Чтобы получить доступ к этой функции, достаточно щелкнуть правой кнопкой мыши файл или папку и выбрать команду **Восстановить прежнюю версию**. Файлы можно просмотреть в режиме «только для чтения», чтобы решить, какую версию файла следует восстанавливать. Чтобы полностью восстановить файл, достаточно перетащить его в любую папку или выделить его и выбрать команду **Восстановить** для восстановления файла в исходное местоположение.

Эта функция позволяет восстанавливать предыдущие версии, как отдельных файлов, так и целых папок. При восстановлении файла отображаются все предыдущие версии, отличные от активной версии на жестком диске. При доступе к предыдущей версии папки пользователи могут просматривать иерархию папок, существовавшую на момент сохранения теневой копии папки

Восстановление системы

Функция восстановления системы была усовершенствована. Теперь для хранения измененных добавочных копий системных файлов применяется теневое копирование. Это позволяет ускорить процесс восстановления и сделать его более надежным. Если компьютер находится в аварийном состоянии и на нем невозможно запустить функцию восстановления системы, можно воспользоваться средой восстановления **Windows** для запуска этой функции с установочного диска **Windows**. Теперь в функции восстановления системы предусмотрена возможность отмены – перед изменением системы автоматически создается точка восстановления на тот случай, если после восстановления системы потребуется отменить это действие.

Методы повышения производительности

Благодаря новым технологиям значительно повышается скорость выполнения на ПК повседневных задач. Усовершенствованные функции загрузки и режима сна позволяют настольным и мобильным компьютерам быстрее возвращаться в рабочее состояние. Кроме того, **Windows** эффективнее управляет памятью и устройствами ввода-вывода, за счет чего компьютер более оперативно реагирует на запросы пользовательских приложений.

В **Windows** входит компонент панели управления **Счетчики и средства производительност**, позволяющий оценивать показатели производительности и устранять возникшие трудности. Некоторые компоненты **Windows** и сторонние приложения могут работать только на компьютерах, оборудование которых соответствует определенным требованиям. Новый **индекс производительности Windows** позволяет определить степень этого соответствия и выяснить, будут ли функционировать такие компоненты и приложения.

Технология **Windows ReadyBoost** (прежнее название – **EMD**) обеспечивает значительный рост производительности за счет использования флэш-накопителей без необходимости расширения ОЗУ. Поддерживается и технология гибридных жестких дисков **Windows ReadyDrive** (прежнее название – гибридный жесткий диск), которая предназначена для повышения надежности и производительности, а также увеличения срока работы от батареи.

Пользоваться функцией **Windows ReadyBoost** очень просто. При первой установке съемного запоминающего устройства, например флэш-накопителя USB или карты памяти SD, в соответствующий порт операционная система **Windows** проверяет, обладает ли это устройство достаточным быстродействием для поддержки технологии **Windows ReadyBoost.** Если это так, появляется запрос на использование устройства для повышения производительности системы. Для этой цели можно выделить часть памяти устройства USB, а в другой части, как обычно, хранить файлы. Запоминающее устройство, используемое функцией **Windows ReadyBoost,** можно в любое время отключить – это не приведет к потере данных и не окажет негативного воздействия на систему. Данные на запоминающем устройстве шифруются для предотвращения несанкционированного доступа.

Для использования функции **Windows ReadyBoost** объем съемной флэш-памяти должен быть не менее 256 МБ и она должна иметь достаточное быстродействие. Скорость прямого чтения блоков размером 4 КБ

должна составлять не менее 2,5 МБ/с, скорость прямой записи блоков размером 512 КБ – не менее 1,75 МБ/с.

Автоматическое управление памятью при переходе в режим сна осуществляется с помощью нового диспетчера памяти **Windows SuperFetch**. Эта новая технология управления памятью позволяет добиться высокой скорости реагирования на запросы пользовательских приложений за счет управления памятью с использованием нового алгоритма, который назначает пользовательским приложениям более высокий приоритет, чем фоновым процессам, и адаптируется к стилю работы каждого пользователя. Фоновые процессы продолжают выполняться, пока компьютер бездействует. Однако по завершении фонового процесса **SuperFetch** восстанавливает в памяти данные, с которыми пользователь работал до запуска этого процесса.

Кроме того, для повышения быстродействия ПК функция **SuperFetch** определяет наиболее часто используемые приложения и предварительно загружает их в память. За счет этого увеличивается скорость работы системы при запуске или переключении на другой профиль пользователя.

Низкоприоритетный ввод-вывод при работе с диском для фоновых процессов – в соответствии с ним фоновые процессы имеют более низкий приоритет при доступе к жесткому диску, чем пользовательские приложения. Если программа поддерживает низкоприоритетный ввод-вывод, то она может выполняться одновременно с пользовательским приложением, не замедляя его работы. Именно так функционирует ряд служб **Windows**, включая индексирование поиска, автоматическую дефрагментацию диска и ежедневную проверку системы **Защитник**ом **Windows.**

Автоматическая дефрагментация жесткого диска может выполняться в фоновом режиме по расписанию без снижения быстродействия приложений.

Работа в сети

Центр управления сетями и общим доступом позволяет полностью управлять работой в сети, включая проверку состояния подключения, просмотр графической схемы сети и устранение неполадок с подключением. Центр управления сетями и общим доступом информирует пользователя о сети, к которой подключен компьютер, и проверяет возможность доступа к **Интернету**. Информация может быть представлена даже в графической форме – в виде так называемой карты сети, с помощью которой можно быстро определить состояние подключения к локальной сети и

Интернету. Если установленное компьютером подключение к **Интернету** разрывается, то на изображении можно найти это подключение, а затем с помощью средств диагностики определить причины и возможное решение проблемы.

Сетевой Проводник. После настройки сети, данные о подключенных к ней компьютерах, устройствах и принтерах можно просматривать точно так же, как папки на локальном ПК. Для выполнения этой задачи служит новый сетевой Проводник **Windows.**

Карта сети. Если к сети проводным и беспроводным способом подключено несколько компьютеров и устройств, иногда бывает трудно понять, как они связаны между собой. **Windows Vista** содержит новый компонент **Карта сети**, с помощью которого можно в графическом виде наглядно представить все компоненты сети и связи между ними.

Служба сведений о подключенных сетях, входящая в **Windows Vista**, сообщает совместимым приложениям об изменении параметров сети. Это позволяет оптимизировать доступ к ресурсам и их использование. При подключении к другой сети приложения, поддерживающие эту функцию, производят соответствующую настройку параметров.

XPS-документы

XPS-документ – это файл, который сохранен в формате XML Paper Specification. **Windows Vista** при установке задает по умолчанию печать на устройство **Microsoft XPS Document Writer**, в результате создается документ **XPS** в папке, указанной пользователем.

Рекомендуется использовать средство создания **XPS**-документов в тех случаях, когда требуется создать, отослать, сделать общими или опубликовать документы, сделав их недоступными для изменения, или когда необходимо напечатать или отобразить документ в Интернете в том же виде, что и на экране локального компьютера. Кроме того, рекомендуется создавать **XPS**-версии для файлов, содержащих графику или иллюстрации, которые в противном случае могут отображаться при печати иначе, чем в Интернете, или на компьютерах с другими мониторами.

Просматривать и печатать **XPS**-документы можно только при помощи средства просмотра **XPS**, которое является компонентом обозревателя **Internet Explorer 7 Windows Vista.**

Новый формат документов **XPS** открывает перед пользователями новые возможности печати и обеспечивает решение проблем с точностью цветопередачи, размером файлов, содержащих сложные цветовые эффекты, и скоростью печати таких файлов.

Центр обновления Windows

Пользователи лицензированных копий **Windows Vista** имеют право получать обновления для операционной системы, периодически вы пускаемые корпорацией Майкрософт. К ним относятся обновления для отдельных функций **Windows Vista**, обновления, повышающие надежность и производительность, а также обновления, защищающие от вредоносных программ и других видов нежелательного программного обеспечения. Кроме того, некоторые обновления, предназначенные для повышения надежности и производительности, могут поставляться производителем компьютера для других программных и аппаратных компонентов.

Центр обновления **Windows** определяет, применимы ли обновления на компьютере пользователя, и при условии соответствующей настройки загружает и устанавливает их автоматически, обеспечивая безопасность ПК и актуальность программного обеспечения.

Конференц-зал Windows

Конференц-зал **Windows** – это новое средство поддержки совместной работы, входящее в состав **Windows Vista**. Это простой, но мощный инструмент, который помогает небольшим группам пользователей **Windows Vista** работать совместно в любом месте и в любое время.

Конференц-зал **Windows** поддерживает взаимодействие от двух до десяти человек через проводную или беспроводную локальную сеть либо временную беспроводную сеть типа компьютер-компьютер. С помощью этой программы можно выполнять самые разные задачи – например, проводить презентацию **Microsoft Office PowerPoint** или вносить исправления в электронную таблицу. Соединения между компьютерами участников устанавливаются быстро, просто и безопасно. Один из участников инициирует сеанс в конференц-зале **Windows.** А другие участники могут присоединиться к сеансу, обмениваться файлами, следить на экране за одним и тем же окном приложения или рабочим столом и совместно работать в режиме реального времени.

Учетные записи пользователей и групп, профили пользователей

Учетные записи пользователей или группы пользователей определяют права и разрешения при работе на определенном компьютере. Каждая учетная запись имеет свое имя (имя пользователя), пароль, а также содержит список групп пользователей, членом которых является данная

учетная запись. Учетные записи пользователей могут храниться на локальном компьютере, либо в службе каталогов (Active Directory) на компьютере-сервере, который является контроллером домена, если компьютер работает в локальной сети и присоединен к данному домену (например, главный домен АГТУ имеет имя ASTU). Управление учетными записями в первом случае осуществляется с помощью программного компонента локального компьютера *Учетные записи пользователей* Панели управления, либо оснастки *Управление компьютером – Локальные пользователи и группы*, во втором случае – администратором на компьютере, являющемся контроллером домена в оснастке *Active Directory – пользователи и компьютеры*.

В папке ***Пользователи***, размещенной в оснастке *Локальные пользователи и группы* консоли управления Microsoft, или в разделе *Учетные записи пользователей -Управление учетными записями* отображаются как учетные записи пользователей, создаваемые по умолчанию при установке системы, так и учетные записи, создаваемые пользователем. При установке Windows создаются встроенные учетные записи **Администратор** (пароль задается при установке системы) и **Гость** (при установке системы отключена, без пароля). Первая из них имеет неограниченные права, вторая – самые ограниченные права среди встроенных групп, основными из которых являются: **Администраторы**, **Опытные пользователи**, **Пользователи**, **Гости**, **Операторы печати**, **Операторы архива**, **Операторы настройки сети** и др. (для ОС Vista домашних версий только Администратор и Обычный доступ).

При создании новой учетной записи она автоматически включается в группу Пользователи, далее можно ее изменить или добавить другие группы.

Для изменения прав локальной группы или пользователя следует использовать оснастку *Локальные параметры безопасности – Локальные политики – Назначение прав пользователя*, где задаются права для групп и пользователей.

Профиль пользователя – папка, содержащая сведения о настройках для конкретного пользователя (рабочего стола, сетевых подключений, параметров приложений…) и личные папки для хранения документов, изображений, музыки, видео и пр.

Профили пользователей предоставляют следующие преимущества:

- Один компьютер может служить нескольким пользователям. При входе в систему пользователь получает настроенную рабочую среду, которая была сохранена для него при выходе из системы.

- Настройка рабочей среды одним пользователем не оказывает воздействия на параметры рабочей среды другого.
- Профили пользователей могут храниться на сервере и применяться для входа пользователя в сеть с любого компьютера.

Профили пользователей для Windows 2000 и XP располагались в папке *systemdrive*\Documents and Settings (например, C:\Documents and Settings), для Windows Vista и 7 – в папке Пользователи (см. рисунок 3.25).

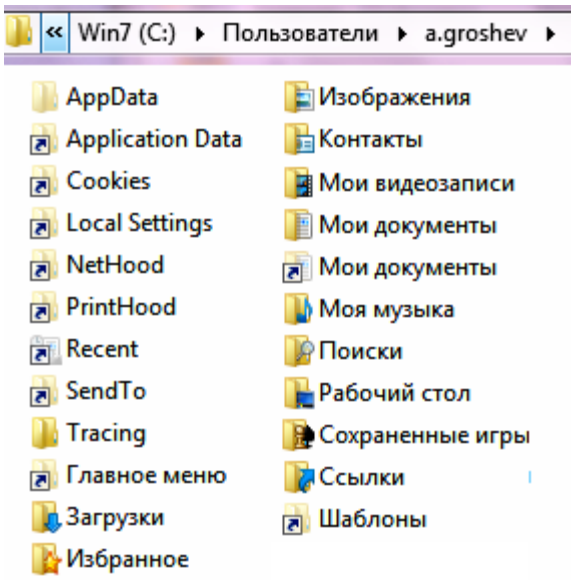


Рисунок 3.25. Папки профиля пользователя в Windows 7

Следует учитывать при работе, что файлы и папки, которые Вы размещаете на рабочем столе и в личных папках, сохраняются в Вашем профиле пользователя!

В Windows 7 в профиле пользователя кроме папки *Мои документы* появились дополнительные папки: *Изображения*, *Мои видеозаписи*, *Моя музыка*, которые являются основой для соответствующих библиотек пользователя.

Назначение системных папок профиля пользователя:

- *Application Data* – программные данные (например, словарь пользователя). Помещаемые в эту папку данные определяются разработчиками приложений.
- *Cookies* – сведения о пользователе и его параметры настройки для Интернет-сайтов.
- *Рабочий стол* – элементы рабочего стола, включая файлы, ярлыки и папки.
- *Избранное* – ярлыки избранных веб-узлов Интернета.
- *Local Settings* – файлы данных приложений, файлы журналов и временные файлы.
- *Недавние документы* (*Recent*) – ярлыки последних используемых документов и папок.
- *NetHood* – ярлыки для элементов сетевого окружения.
- *PrintHood* – ярлыки для элементов папки принтеров.
- *SendTo* – ярлыки служебных программ по работе с документами.

- *Главное меню* – ярлыки программ.
- *Шаблоны* – шаблоны пользователя.

Управление доступом

Управление доступом заключается в предоставлении пользователям, группам и компьютерам определенных прав на доступ к объектам на локальном компьютере или по сети. Ключевыми понятиями управления доступом являются разрешения, владение объектами, наследование разрешений, права пользователя и аудит объектов.

Разрешения

Разрешения определяют тип доступа к объектам Windows или их свойствам, предоставляемый пользователю или группе (вкладка *Безопасность* окна свойств папки или файла).

Чтобы просмотреть или изменить разрешения на доступ к файлу или папке, достаточно в Проводнике щелкнуть правой кнопкой мыши на имени файла или папке и выбрать пункт ***Свойства***. На вкладке ***Безопасность*** показаны текущие разрешения, которые можно изменить, если Вы обладаете необходимыми для этого правами.

Например, студентам группы 555 могут быть предоставлены разрешения на создание, удаление, чтение и запись файлов в папке C:\Мои документы\xt-5 и запрещен доступ в другие вложенные папки для C:\Мои документы.

С помощью управления доступом можно установить разрешения NTFS для таких объектов, как файлы, объекты Active Directory, объекты реестра или системные объекты, например, процессы. Разрешения могут быть предоставлены любому пользователю, группе или компьютеру. Целесообразно установить разрешения для групп, поскольку эту увеличивает быстродействие системы при проверке прав доступа к объекту.

Для любого из этих объектов можно предоставить разрешения:

1. локальным группам и пользователям на том компьютере, на котором находится объект;
2. группам и пользователям данного или любого доверенного домена;
3. группам, пользователям и другим объектам с идентификаторами безопасности домена.

Разрешения, назначаемые объекту, зависят от его типа. Например, разрешения, которые могут быть назначены для файла, отличаются от разрешений, допустимых для раздела реестра. Однако некоторые разрешения

являются общими для большинства типов объектов. Общие разрешения перечислены ниже.

- Чтение
- Изменение
- Смена владельца
- Удаление

При установке разрешений определяется уровень доступа для групп и пользователей. Например, одному пользователю можно разрешить читать содержимое файла, другому – вносить изменения в файл, а всем остальным пользователям вообще запретить доступ к этому файлу. Так же можно устанавливать разрешения на доступ к принтерам, чтобы одни пользователи могли настраивать принтер, а другие – только печатать.

Владение объектами

При создании объекта ему назначается владелец. По умолчанию владельцем объекта назначается тот, кто его создал. Какие бы разрешения ни были установлены для объекта, владелец объекта всегда может изменить эти разрешения.

Наследование разрешений

Механизм наследования облегчает администраторам задачи назначения разрешений и управления ими. Благодаря этому механизму разрешения, установленные для контейнера, автоматически распространяются на все объекты этого контейнера. Например, файлы, создаваемые в папке, наследуют разрешения этой папки. Наследуются только отмеченные для наследования разрешения.

Права пользователя

Права пользователей позволяют пользователям и группам в компьютерной среде пользоваться особыми привилегиями и правами на вход в систему. Администраторы могут назначать права учетным записям групп или отдельных пользователей. Эти права позволяют пользователям выполнять конкретные действия, такие как интерактивный вход в систему или архивирование файлов и каталогов.

Права пользователей отличаются от разрешений тем, что применяются к учетным записям пользователей, а не к объектам. Хотя права пользователей и могут применяться к учетным записям отдельных пользователей, удобнее администрировать права пользователей на основе групп.

Пользовательский интерфейс управления доступом не поддерживает прав пользователей, однако их предоставлением можно управлять с помощью оснастки «Локальные параметры безопасности».

Аудит объектов

Обладая правами администратора, можно контролировать успешные и неудавшиеся попытки доступов пользователей к объектам. Можно выбрать объект, аудит доступа к которому выполняется с помощью пользовательского интерфейса управления доступом, но сначала необходимо включить политику аудита, выбрав доступ к объекту аудита в элементе «Локальные политики» в оснастке «Локальные параметры безопасности». Впоследствии события, связанные с безопасностью, можно просмотреть в журнале безопасности в окне программы «Просмотр событий».

Общий доступ к папкам и файлам

В **Windows 7** реализовано два способа предоставления общего доступа к файлам: можно предоставить общий доступ к файлам, находящимся в группе папок *Общие* (*Public*), или к файлам любой папки на компьютере, задавая общий доступ к папке в её свойствах (рисунок 3.26).

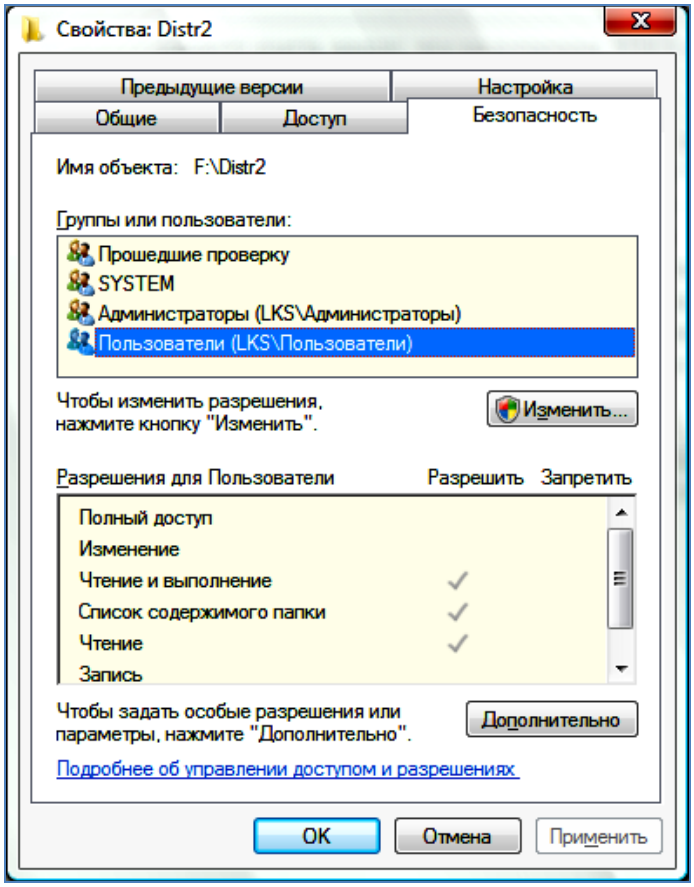


Рисунок 3.26. Окно свойств папки

Папки *Общая музыка, Общие видео, Общие документы, Общие изображения* в папке *Общие* являются удобным средством для обмена файлами, хранящимися на компьютере. Можно обмениваться файлами из этих папок с другими пользователями компьютера или же с пользователями других компьютеров, находящихся в од-

ной сети. Любой файл или папка, помещенные в эти папки, автоматически становятся доступными для других пользователей.

Папку *Общие* следует использовать, если Вы хотите хранить все файлы, к которым необходимо предоставить совместный доступ, в одном месте на компьютере и нет необходимости настраивать разрешения для отдельных пользователей сети.

Второй способ следует использовать, если Вы хотите предоставить доступ к файлам другим пользователям прямо из места их размещения, при этом необходимо настроить разрешения на доступ к файлам для отдельных пользователей компьютера – вкладка *Безопасность* окна свойств или сети – вкладка *Доступ* окна свойств (см. рисунок 3.26). Окончательные разрешения на общий доступ к папке определяются с учетом как набора разрешений NTFS (*Безопасность),* которые можно также установить для файлов, так и для общего ресурса (*Доступ*).

Панель управления **Windows Vista** содержит новый компонент, называемый **Центром управления сетями и общим доступом**. С помощью этого центра можно включить или отключить сетевое обнаружение и общий сетевой доступ к файлам и папкам компьютера.

Улучшенные стандартные приложения Windows 7

Впервые за всю историю Windows в новой версии были значительно переработаны такие приложения, как Калькулятор, WordPad, Paint.

В Калькуляторе кроме режимов *Обычный* и *Инженерный* появились новые варианты *Программист* и *Статистика*.

Интерфейс программ WordPad, Paint использует вместо меню ленты с вкладками, по стилю похожие на Microsoft Office 2010. Возможности этих двух программ расширены.

WordPad может работать с форматами RTF, Office Open XML (*.docx) и простым текстом в кодировках Windows, MS-DOS и Юникод. При загрузке в эту программу документа Microsoft Office Word 2007 или 2010 могут возникнуть некоторые искажения при отображении шрифтов, рисунков и других внедренных объектов (см. рисунок 3.27).

Новый WordPad может форматировать текст по ширине, работать со списками, рисунками и с внедренными объектами (формулы Microsoft Equation, диаграммы Microsoft Graph, при установленнлй системе Microsoft Office – с листами и диаграммами Excel, презентациями и слайдами PowerPoint и пр.).

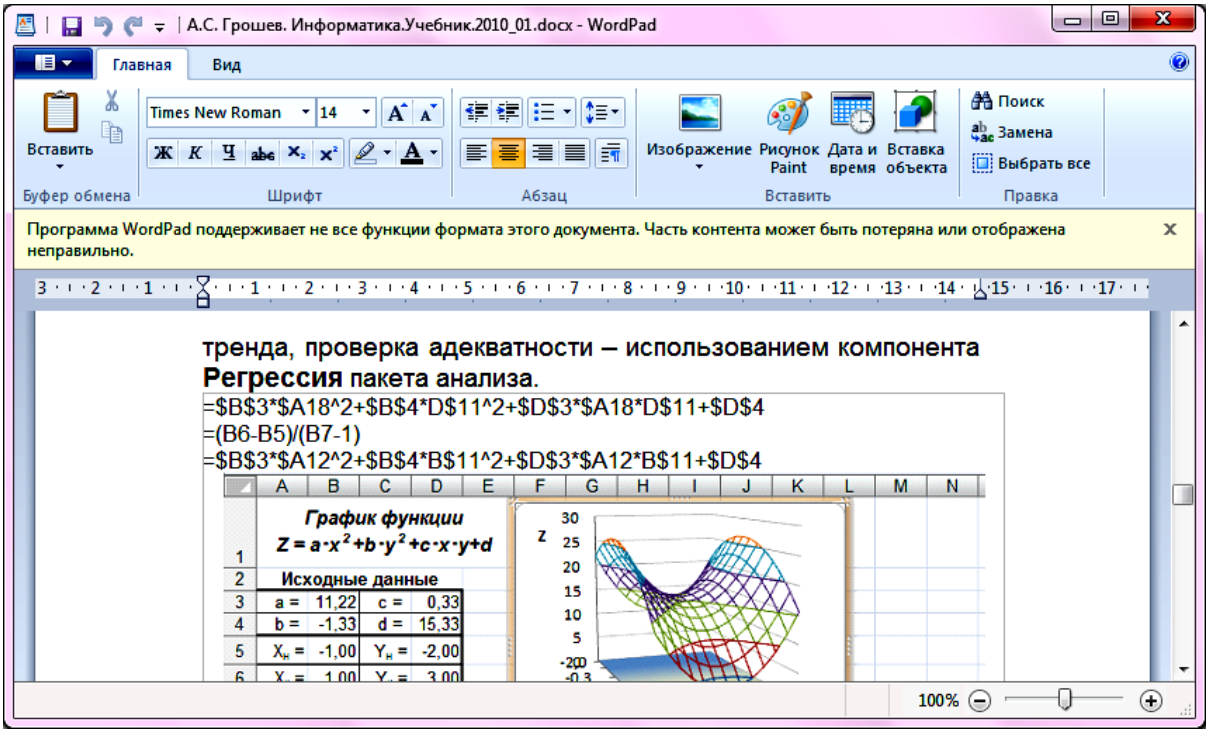


Рисунок 3.27. Текстовый редактор WordPad системы Windows 7

Графический редактор Paint имеет много новых возможностей, которые присутствуют на вкладке ***Главная*** ленты (см. рисунок 3.28), таких как 9 видов кисти, 23 вида фигур для вставки в рисунок и пр., а также дополнительные возможности контекстного меню для всего рисунка или области выделения (обрезать, повернуть, изменить размер, обратить цвета и пр.).

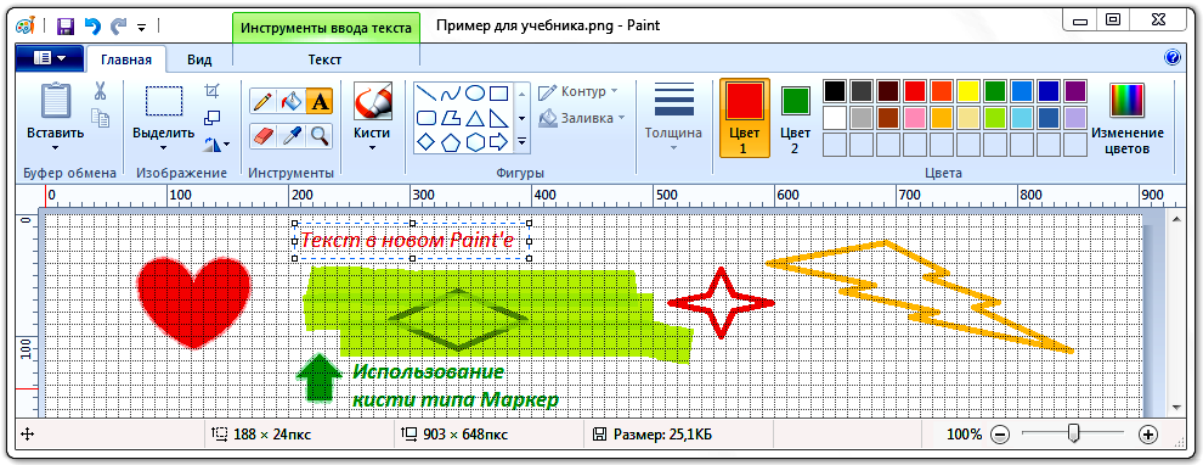


Рисунок 3.28. Графический редактор Paint системы Windows 7

Дополнительные компоненты на сайте Windows Live

В состав **Windows 7** не входит почтовая программа и программа видеомонтажа. Скачать их и другие приложения можно бесплатно с сайта *http://download.live.com/*.

Некоторые программы, доступных для загрузки:

1) ***Live* Messenger** – обмен мгновенными сообщениями;
2) ***Почта Windows Live*** – работа с учетными записями электронной почты Hotmail, Gmail и других служб;
3) ***Редактор блогов (Live Writer)*** – оформление блогов и их рассылка на сайты;
4) ***Фотоальбом Windows Live*** – редактирование и упорядочение фотографий и фильмов, публикация их в сети или отправка по электронной почте;
5) ***Киностудия Windows Live*** – простейшая программа видеомонтажа с сохранением фильмов в формате HD *.wmv или с размещением их на YouTube;
6) ***Семейная безопасность*** – родительский контроль за посещением сайтов и за общением в Интернете.

3.2 Сервисные программы

Назначение сервисных программ – обеспечение надежной работы компьютеров и их сетей: тестирование и оптимизация, обеспечение резервного копирования и восстановления, защита от вредоносных, шпионских и мошеннических вторжений.

В последнее время наиболее крупные фирмы – поставщики подобного программного обеспечения (**Symantec, Vcom, Acronis, Aladdin Systems, PGSoft, PowerQuest** и пр.), выпускают комплексы сервисных пакетов, предназначенных для домашних компьютеров, малого и среднего бизнеса, крупных предприятий.

По данным фирмы **Symantec Corporation** наибольшая угроза для компьютерных систем возникает при работе с информацией по протоколу HTTP в Интернете. Меньшая опасность – электронная почта, службы быстрых сообщений (ICQ и пр.), локальная сеть.

3.2.1 Сервисные программы корпорации Symantec

В качестве примера рассмотрим продукты корпорации **Symantec** (www.Symantec.ru).

Фирма предлагает 3 группы продуктов:

1) для дома:
   - **Norton 360** – комплексная автоматическая система безопасности, тестирование и оптимизация работы компьютера;
2) для малого и среднего бизнеса:
   - антивирусная защита:
     - Multi-tier Protection
     - Endpoint Protection
     - Endpoint Protection Small Business Edition
     - Mobile Security for Symbian
     - Mobile AntiVirus for Windows Mobile
   - защита конечных систем
     - Multi-tier Protection
     - Network **Access** Control Starter Edition
     - Endpoint Protection Small Business Edition
     - Mobile AntiVirus for Windows Mobile
   - управление клиентами
     - Ghost Solution Suite
     - pcAnywhere
   - резервное копирование и восстановление
     - Backup Exec for Windows Servers
     - Backup Exec for Windows Small Business Server
     - Backup Exec System Recovery Server Edition
     - Backup Exec System Recovery Desktop Edition
     - Backup Exec System Recovery Windows Small Business Server Edition
     - Enterprise Vault
   - защита систем доставки сообщений
     - Mail Security for Microsoft Exchange
     - Mail Security with Premium AntiSpam for Windows Small Business Server
     - Mail Security for Domino
     - Mail Security Enterprise Edition
     - Mail Security for SMTP
     - Hosted Mail Security for Small Business
     - IM Manager
   - управление безопасностью
     - Control Compliance Suite
     - Enterprise Vault

3) для крупных предприятий – список компонентов в несколько раз длиннее, чем предыдущий.

Комплексная программа для домашних компьютеров **Norton 360** обеспечивает антивирусную защиту, защиту от программ-шпионов, защиту от фишинга, обнаружение руткитов, проверку подлинности веб-сайтов, двусторонний брандмауэр, автоматическое резервное копирование и восстановление, 2 ГБ памяти в безопасном сетевом электронном хранилище корпорации Symantec.

Важнейшее место среди сервисных программ простых пользователей занимают средства для борьбы с компьютерными вирусами, средства для архивации файлов, программы для работы с оптическими дисками.

3.2.2 Защита от вирусов

Компьютерные вирусы

Компьютерный вирус – вредоносная программа, способная к саморазмножению и распространению при передаче информации с одного компьютера на другой.

Официально считается что термин «компьютерный вирус» впервые употребил сотрудник Лехайского университета (США) Ф. Коэн в 1984 г. на 7-й конференции по безопасности информации, проходившей в США.

В настоящее время кроме классических компьютерных вирусов существуют также сетевые вирусы (черви и троянские программы), а также прочие вредоносные программы.

Типы вирусов:

- файловые;
- загрузочные;
- макро-вирусы;
- скриптовые;
- сетевые.

Файловые вирусы различными способами внедряются в исполняемые файлы, либо создают файлы-двойники.

Пример, показывающий, насколько сложными являются подобные программы – вирус WinNT.RemEx (Remote Explorer). При запуске файла вируса его код копируется в системный каталог WinNT System32\Drivers под именем IE403R.SYS и запускается как системный сервис, который один раз в 10 минут запускает одну из своих подпрограмм – либо заражения, либо «заметания сле-

дов». Процедура заражения сканирует случайно выбранные локальные и доступные сетевые папки, ищет EXE-файлы и заражает их. При заражении вирус компрессирует файл-жертву, записывает в него свой код и добавляет компрессированный первоначальный код файла в его конец. При запуске зараженного EXE-файла вирус копирует его во временный файл, распаковывает и выполняет исходный файл программы. Вирус компрессирует файлы методом «deflate» и использует для этого встроенную в свой код библиотеку GZIP. В зависимости от случайного счетчика вирус также портит случайно выбранные файлы: компрессирует их тем же методом и затем шифрует крипто-алгоритмом средней сложности. Процедура «заметания следов» запускается следом за заражением и уничтожает следы жизнедеятельности вируса. Вирус закрывает сообщения об ошибках, произошедших при заражении файлов, и выполняет другие действия (по данным Лаборатории Касперского *http://www.kav.ru*).

Загрузочные вирусы записывают себя в загрузочный сектор диска (boot-сектор), либо в сектор, содержащий системный загрузчик винчестера (Master Boot Record), либо меняют указатель на активный boot-сектор. Данный тип вирусов был достаточно распространён в 1990-х годах, но практически исчез с переходом на 32-разрядные операционные системы и отказом от использования дискет как основного способа обмена информацией. Теоретически возможно появление загрузочных вирусов, заражающих оптические диски и USB-флэш-память, но на текущий момент такие вирусы не обнаружены.

Макро-вирусы заражают файлы данных различных популярных программ. Многие текстовые, табличные и графические редакторы, системы проектирования, имеют свои языки программирования для работы с различными объектами, присутствующими в файлах данных этих систем (например, Visual Basic for Application (VBA)в системе **Microsoft Office**). Текст программ на языке VBA сохраняется в модулях процедур (макросах), хранящихся в файлах документов (см. в меню системы **Microsoft Office** 2003 раздел Сервис - Макрос). Макро-вирусы являются программами на языках, встроенных в такие системы обработки данных. Например, при открытии документа Word проверяет его на наличие процедуры с именем AutoOpen, если она присутствует, то Word выполняет ее. Для своего размножения вирусы этого класса используют возможности языков этих систем и при их помощи переносят себя из одного зараженного файла (документа, таблицы или другого файла данных) в другие. По данным Лаборатории Касперского количество макровирусов – несколько сотен.

Пример опасного макро-вируса: Macro.Word.KillDll. Содержит единственный макрос: AutoOpen. При открытии зараженного файла вирус заражает область глобальных макросов системы Word. После этого при

открытии в системе Word незараженного файла заражает его и уничтожает все файлы по маске *.D?? в папке C:**WINDOWS** \SYSTEM\ (т. е. файлы динамических библиотек и драйверов **Windows**).

Скриптовые вирусы написаны на различных скрипт-языках (VBS – Visual Basic Script, JS – Java Script, BAT, PHP и пр.), которые широко используются в ОС **Windows** и Web-программировании. Они либо заражают другие скрипт-программы, присутствующие в виде программных файлов или фрагментов текста html-страниц, либо являются частями многокомпонентных вирусов.

Пример: Virus.VBS.Redlof – написан на языке VBS и зашифрован. При первом запуске создает файл со своим исполняемым кодом в системном каталоге **Windows** с именем Kernel.dll. Кроме того, копирует себя во все каталоги на других дисках зараженного компьютера в виде файла настройки отображения файлов и папок – folder.htt. Зараженный файл folder.htt получает управление и копируется вирусом во все папки при их открытии при помощи Проводника. Вирус дописывает себя во все HTM файлы, находящиеся в каталоге **Windows** \web и таким образом он также получает управление при открытии данных файлов.

Сетевые вирусы используют для своего распространения протоколы и команды компьютерных сетей и электронной почты.

Троянская программа – любая программа, которая втайне от пользователя выполняет на компьютере какие-либо нежелательные для него действия.

По своим функциям подразделяются на:

- бэкдор (Backdoor) – несанкционированный доступ к удалённому администрированию компьютера;
- PSW-троянцы – похищение паролей;
- Trojan Clicker – открывает на компьютере веб-ссылки без ведома пользователя;
- Trojan Downloader – загрузка из интернета каких-либо файлов без ведома пользователя;
- Trojan-Dropper – инсталлирует без ведома пользователя другие вредоносные программы на зараженный компьютер и запускает их;
- Trojan Proxy – запускает без ведома пользователя прокси-сервер на зараженном компьютере;
- прочие троянцы.

Сетевые черви подразделяются на почтовые (Email-Worm), использующие интернет-пейджеры типа ICQ (IM-Worm), использующие IRC-

средства Интернет-общения (IRC-Worm), заражающие Интернет-серверы и прочие. Почтовые черви после инсталляции в системе могут рассылать себя по всем адресам, обнаруженным в адресной книге **Windows** и в файлах электронной почты. Другие виды сетевых червей могут рассылать URL-ссылку на себя или другие сайты по адресам, обнаруженным в базах Интернет-пейджеров или IRC-клиентов, открывать полный сетевой доступ на диски компьютера и пр.

К вредоносным программам относятся также **руткиты** (rootkits) – утилиты, используемые хакерами для сокрытия своей вредоносной деятельности на зараженном компьютере. Для этого они модифицируют операционную систему компьютера. **Руткиты** не только скрывают свое собственное присутствие, но также и действия, которые предпринимает злоумышленник. Р**уткит** может скрывать присутствие других вредоносных программ на компьютере, изменяя файловые данные, ключи реестра или активные процессы.

Каждый файловый или сетевой вирус существует в какой-либо одной или нескольких ОС – **DOS**, **Windows 9×**, **Windows NT**, **OS/2, Unix** и т.д.

Особенности алгоритмов работы вирусов:

- резидентность;
- использование стелс-алгоритмов;
- самошифрование и полиморфичность;
- использование нестандартных приемов.

Резидентный вирус при инфицировании компьютера оставляет в оперативной памяти свою резидентную часть, которая затем перехватывает обращения операционной системы к объектам заражения и внедряется в них. Резидентные вирусы находятся в памяти и являются активными вплоть до выключения компьютера или перезагрузки операционной системы.

Резидентными можно считать макро-вирусы, поскольку они постоянно присутствуют в памяти компьютера на все время работы зараженного редактора.

Нерезидентные вирусы не заражают память компьютера и сохраняют активность ограниченное время. Некоторые вирусы оставляют в оперативной памяти небольшие резидентные программы, которые не распространяют вирус. Такие вирусы считаются нерезидентными.

Использование стелс-алгоритмов позволяет вирусам полностью или частично скрыть себя в системе. Наиболее распространенным стелс-

алгоритмом является перехват запросов ОС на чтение/запись зараженных объектов. Стелс-вирусы при этом либо временно лечат их, либо «подставляют» вместо себя незараженные участки информации. В случае макровирусов наиболее популярный способ – запрет вызовов меню просмотра макросов.

Самошифрование и полиморфичность используются практически всеми типами вирусов для того, чтобы максимально усложнить процедуру детектирования вируса. Полиморфные вирусы (polymorphic) – это достаточно труднообнаруживаемые вирусы, не имеющие сигнатур, т.е. не содержащие ни одного постоянного участка кода. В большинстве случаев два образца одного и того же полиморфного вируса не будут иметь ни одного совпадения. Это достигается шифрованием основного тела вируса и модификациями программы-расшифровщика.

По деструктивным возможностям вирусы подразделяются на:

- безвредные, т.е. никак не влияющие на работу компьютера (кроме уменьшения свободного места на диске в результате своего размножения);
- неопасные, влияние которых ограничивается уменьшением свободного места на диске и графическими, звуковыми и пр. эффектами;
- опасные вирусы, которые могут привести к серьезным сбоям в работе компьютера;
- очень опасные, в алгоритм работы которых заведомо заложены процедуры, которые приводят к порче программ, уничтожению файлов.

Полную информацию о многочисленных известных вирусах можно найти на сайтах фирм, производителей антивирусных программ, например, на сайте Лаборатории Касперского *http://www.kav.ru*.

Антивирусные программы

Для борьбы с компьютерными вирусами используются антивирусные программы. Так как компьютеры простых пользователей, не подключенные к Интернету, вскоре перестанут существовать, первой программой, устанавливаемой на ПК после установки операционной системы, должна быть антивирусная система. Интернет в настоящее время – источник повышенной вирусной опасности, особенно при посещении хакерских и развлекательных сайтов!

В состав современных антивирусных систем входят, как правило, несколько программных компонентов:

- **сканер** – программа проверки заданных дисков, папок или файлов по указанию пользователя, по расписанию или по команде монитора, программы проверки электронной почты или скриптов;
- **монитор**, запускающий программу-сканер для проверки всех запускаемых на ПК программ и открываемых файлов данных, начиная с момента запуска ОС;
- программы проверки файлов электронной почты и скриптов в файлах Интернета, поступающих на компьютер;
- программа обновления антивирусных баз;
- центр управления, позволяющий задать расписание сканирования папок, обновления баз и прочие параметры работы.

В антивирусных сканерах для поиска известных вирусов используются два основных метода: сигнатурный поиск и эвристический анализ.

Сигнатурой или маской вируса является некоторая постоянная последовательность кода, специфичная для этого конкретного вируса. Все сигнатуры размещены в антивирусной базе – специальном хранилище, в котором программа-антивирус хранит характерные коды вредоносных программ.

Метод эвристического анализа проверяет не код подозрительного файла, а его действия. Для того чтобы размножаться, вирус должен копировать свое тело в память, открывать другие исполняемые файлы и записывать туда свое тело, записывать данные в секторы жесткого диска и т.д. Есть характерные действия и у сетевого червя – доступ к адресной книге и сканирование жесткого диска на предмет обнаружения адресов электронной почты. Благодаря этому эвристический анализатор способен обнаружить даже те вредоносные коды, сигнатуры которых еще неизвестны.

Антивирусные сканеры можно разделить на две категории – универсальные и специализированные. Универсальные сканеры рассчитаны на поиск и обезвреживание всех типов вирусов вне зависимости от операционной системы, на работу в которой рассчитан сканер. Специализированные сканеры предназначены для обезвреживания ограниченного числа вирусов или только одного их класса, например макро-вирусов. Специализированные сканеры, рассчитанные только на макро-вирусы, часто оказываются наиболее удобным и надежным решением для защиты систем документооборота в средах MMicrosoft Word и Excel.

К достоинствам универсальных сканеров относится большое количество вирусов, которые они знают и могут обнаруживать и обезвреживать. Недостатки – относительно невысокая скорость работы и большой

размер антивирусной базы, которую необходимо систематически обновлять (желательно каждый день или чаще).

Существуют также CRC-сканеры, принцип работы которых основан на подсчете контрольных сумм файлов и системных секторов. Эти CRC-суммы затем сохраняются в базе данных этой программы, как, впрочем, и некоторая другая информация: длины файлов, даты их последней модификации и т.д. При последующем запуске CRC-сканеры сверяют данные, содержащиеся в базе данных, с реально подсчитанными значениями. Если информация о файле, записанная в базе данных, не совпадает с реальными значениями, то CRC-сканеры сигнализируют о том, что файл был изменен или заражен вирусом.

Если на компьютере найден вирус, необходимо сделать следующее:

1. Если компьютер подключен к сети, необходимо отключить его от сети и проинформировать системного администратора. Это защитит сервер и другие рабочие станции от заражения. Если же вирус уже заразил сервер, отключение от сети не позволит ему вновь проникнуть на компьютер после лечения. Подключение к сети возможно лишь после того, как будут вылечены все серверы и рабочие станции.

2. При помощи антивирусной программы нужно вылечить зараженные файлы и затем проверить их работоспособность. Если возможно, следует восстановить зараженные файлы из архивной копии, предварительно проверив архив на отсутствие вирусов. После лечения от вируса желательно выполнить сканирование всех дисков компьютера.

3. Если обнаружен резидентный файловый вирус, следует убедиться в том, что программе-сканеру данный вирус известен, и она обезвреживает его резидентную часть. Полезно ознакомиться и с другими характеристиками вируса: типами заражаемых вирусом файлов, действиями вируса и прочее. Источником подробной информации может служить «Вирусная энциклопедия» Лаборатории Касперского (*http://www.viruslist.com/ru/viruses/encyclopedia*).

В локальных сетях предприятий могут использоваться специальные меры борьбы с проникновением вирусов: запрет пользователям, не обладающим административными правами, использовать устройства для работы с оптическими дисками, флэш-картами, USB-флэш памятью, дискетами и пр.; защита компьютеров межсетевым экраном (брандмауэром); настройка прокси-сервера с доступом пользователей только к разрешенным ресурсам локальной сети и Интернета. Стратегическим компонентом защиты данных от вирусов является систематическое резервное копирование информации.

Сертификацией антивирусных программ занимается **ICSA** – International Computer Security Association – Международная ассоциация по компьютерной безопасности (основана в 1992 году). В тестированиях, проводимых **ICSA Labs**, используется вредоносный код как из собственной коллекции, так и из списка **WildList** (список вирусов неформальной международной организации *WildList Organization International*). По результатам исследований продуктам выдаётся сертификат **ICSA** – его удостаиваются те антивирусы, которые способны обнаружить 100% вирусов из списка **WildList**, выпущенного за месяц до испытаний, и не менее 90% вирусов из собственной коллекции **ICSA**.

В списке сертифицированных **ICSA** антивирусов в настоящее время большое количество программ более чем 20-ти различных производителей, некоторые из программ приведены в таблице 3.8.

Таблица 3.8. Антивирусные программы, сертифицированные **ICSA** для **Windows 7** (*http://www.icsalabs.com/products?tid[]=4216*)

Продукт	Производитель
AntiVirus with Spy Sweeper	*Webroot Software, Inc.*
eScan Internet Security	*MicroWorld Technologies, Inc.*
ESET NOD32 Antivirus	*Eset spol. s.r.o.*
Microsoft Forefront Client Security	*Microsoft Corporation*
Microsoft Security Essentials	*Microsoft Corporation*
Norman Security Suite	*Norman ASA*
Panda for Desktops	*Panda Security*
Panda Internet Security 2010	*Panda Security*
PC Tools Spyware Doctor with AntiVirus	*PC Tools Software*
Webroot Internet Security Essentials	*Webroot Software, Inc.*

Сравнение различных антивирусных программ можно найти, например, на сайте *http://www.av-comparatives.org*. Места по проценту обнаруженных вирусов по данным этого сайта показаны в таблице 3.9.

Антивирусные программы постоянной совершенствуются, поэтому в их подсистемах обновления может быть задан параметр загрузки не только обновлений антивирусных баз, но и программы антивирусного сканера.

В настоящее время наиболее крупные фирмы, производители антивирусных программ, выпускают их в составе комплексных систем обеспечения компьютерной безопасности.

Одна из лидирующих компаний в этой области в нашей стране – Лаборатория Касперского (*http://www.kaspersky.ru/business_products*).

Таблица 3.9. Итоговые места по тестированию антивирусов (*www.av-comparatives.org*, ноябрь 2009 г.)

Место	Программа
1.	AVIRA
2.	G DATA
3.	Kaspersky
4.	ESET NOD32
5.	F-Secure
6.	Avast, BitDefender, eScan
7.	AVG, TrustPort
8.	McAfee
9.	Symantec
10.	Sofos
11.	Norman, Kingsoft

Комплекс продуктов **Kaspersky Open Space Security** – решения для защиты всех типов сетевых узлов – от мобильных устройств до серверов; контроль всех входящих и исходящих потоков данных на компьютере – электронной почты, интернет-трафика и всех сетевых взаимодействий, включает в себя 4 выпуска:

- **Kaspersky Work Space Security** (*1-уровневая защита)* – решение для централизованной защиты ***рабочих станций***, в том числе ноутбуков в корпоративной сети и за ее пределами от всех видов современных интернет-угроз: вирусов, шпионских программ, хакерских атак и спама.
- **Kaspersky Business Space Security** (*2-уровневая защита)* - защищает рабочие станции и файловые серверы от всех видов вирусов, троянских программ и червей, предотвращает вирусные эпидемии, а также
- обеспечивает сохранность информации и мгновенный доступ пользователей к сетевым ресурсам.
- **Kaspersky Enterprise Space Security** (*3-уровневая защита*) защищает рабочие станции, файловые и почтовые серверы от всех видов современных интернет-угроз.
- **Kaspersky Total Space Security** (*многоуровневая защита*) защищает все узлы корпоративной сети любого масштаба и сложности от современных интернет-угроз.

Компоненты системы Kaspersky Total Space Security (http://www.kaspersky.ru/enterprise_space_security):

- *Антивирус Касперского для Windows Workstation*
- *Антивирус Касперского для Linux Workstation*
- *Антивирус Касперского для Windows Server*
- *Kaspersky Mobile Security Enterprise Edition*
- *Антивирус Касперского для Windows Servers Enterprise Edition*

- *Антивирус Касперского для Linux File Server*
- *Антивирус Касперского для Samba Server*
- *Антивирус Касперского для Novell NetWare*
- *Антивирус Касперского для Microsoft Exchange 2000/2003*
- *Kaspersky Security для Microsoft Exchange Server 2003*
- *Kaspersky Security для Microsoft Exchange Server 2007*
- *Антивирус Касперского для Lotus Notes/Domino*
- *Антивирус Касперского для Linux Mail Server*
- *Kaspersky Mail Gateway*
- *Kaspersky Anti-Spam*
- *Антивирус Касперского для Microsoft ISA Server*
- *Антивирус Касперского для Proxy Server*
- *Антивирус Касперского для Check Point FireWall-1*
- *Kaspersky Administration Kit*

В таблице 3.10 приведен список защищаемых объектов различных выпусков этой системы.

Таблица 3.10. Возможности различных выпусков системы Kaspersky Open Space Security

Защищаемые объекты	Kaspersky Work Space Security	Kaspersky Business Space Security	Kaspersky Enterprise Space Security	Kaspersky Total Space Security
Рабочие станции	×	×	×	×
Файловые серверы		×	×	×
Почтовые серверы			×	×
Интернет-шлюзы				×
Смартфоны				×

Некоторые фирмы выпускают бесплатные антивирусные программы для домашнего использования или пробные версии с ограниченной функциональностью. Например, немецкая фирма AVIRA предлагает скачать бесплатную версию – программу AntiVir Personal Edition Classic (*http://www.free-av.com*), эта программа защищает пользователя не только от вирусов, червей и троянов, но также от программ-шуток, руткитов и фишинга. Шпионские и рекламные модули эта версия обнаруживать не умеет, подобная защита имеется только в платной версии AntiVir Personal Edition Premium и версиях продукта для рабочих станций и серверов. Эти

версии также можно бесплатно скачать для предварительного ознакомления с ними в течение 30 дней.

Подобная ситуация с антивирусными программами Avast! – версию для домашнего использования ***Home Edition*** можно загрузить через Интернет (*http://www.avast.ru/Free_avast_home_edition.htm*) и использовать бесплатно после регистрации пользователя и получения ключа для использования через электронную почту. Другие версии – Avast! Professional Edition, Avast! Server Edition и пр. являются платными.

3.2.3 Архивация файлов

В операционной системе **Windows** XP и более старших версиях имеются собственные средства архивации файлов (создание ZIP-архива) командой **Переместить** – **Папка архива** в контекстном меню или пункте **Файл** главного меню для группы выбранных файлов или папок.

Однако наиболее известные программы-архиваторы **WinRar** и **WinZip** предоставляют пользователю дополнительные возможности, например, для **WinRar**:

- создание архивов в форматах RAR и ZIP;
- разархивация для архивов форматов RAR, ZIP, ACE, ARJ, CAB, ISO, JAR, LZH, TAR и др.;
- поддержка технологии перетаскивания при работе с архивами (drag & drop);
- возможность использования интерфейса командной строки;
- задание степени сжатия (без сжатия, скоростной, быстрый, обычный, хороший, максимальный);
- присутствие в контекстном меню работы с файлами и папками команд программы WinRar (добавить в архив, добавить в архив <имя>, добавить в архив и отправить по e-mail, добавить в архив <имя> и отправить по e-mail);
- поддержка кодировки Unicode в именах файлов;
- тестирование архивов после упаковки;
- кроме того, возможности только для формата RAR:
  - занесение в архив только файлов для заданных условий их изменения (с любым временем; старше, чем; новее, чем; измененные до; измененные после);
  - создание многотомных архивов, состоящих из нескольких файлов заданного размера для одного архива (имена файлов имеют вид «имя_тома.partNNN.rar», где NNN – номер тома);

- создание самораспаковывающихся обычных и многотомных SFX-архивов (exe-файлов);
- создание непрерывных (solid) архивов – архив RAR, упакованный специальным способом, при котором все сжимаемые файлы рассматриваются как один последовательный поток данных;
- выбор – архивировать или нет открытые пользователями файлы;
- возможность сохранения данных о правах доступа (для **NTFS**);
- возможность восстановления физически поврежденных архивов;
- шифрования данных и имён файлов при задании пароля архива;
- задание параметров сжатия текста, аудио, графики, исполняемых файлов и структурированных таблиц;
- добавления комментариев в архив, ведения протокола ошибок и пр.

3.2.4 Работа с оптическими дисками

Программы записи дисков

В операционной системе **Windows XP** и более старших версиях имеются собственные средства записи оптических дисков. **Windows Vista** и **Windows 7** имеют поддержку файловой системой **UDF** для работы с CD и DVD дисками. На рисунке 3.29 показано окно подготовки оптического диска к записи, которое появляется после двойного щелчка на иконке DVD-привода с чистым диском, в котором присутствует 2 режима записи: **LFS** – Live File System (ранее и обозначалась **UDF)** и **Mastered** (ранее обозначалась **ISO 9660**).

Формат **LFS** позволяет копировать на диск и удалять на перезаписываемых дисках каждый файл по отдельности, чего нельзя сделать для формата **Mastered**. **LFS** имеет несколько версий: 1.02, 1.50, 2.00, 2.01 и 2.50. Последняя версия предназначена только для Windows Vista и более поздних ОС. Версии 1.02, 1.5 и 2.00 можно использовать для записи видео на диски, используемые на бытовых DVD-плерах. Формат **Mastered** не позволяет записывать на диск файлы с размером более 2 Гб.

В выпусках **Windows Home Premium** и **Ultimate** присутствует компонент ***Windows Media Center*** с большими возможностями по записи радио и телепередач, создания из аудио и видеофайлов большого размера DVD-дисков для обычных бытовых DVD-плееров (с разбивкой на необходимое количество файлов *.vob с размером 1,048 Гб и созданием необходимых информационных файлов).

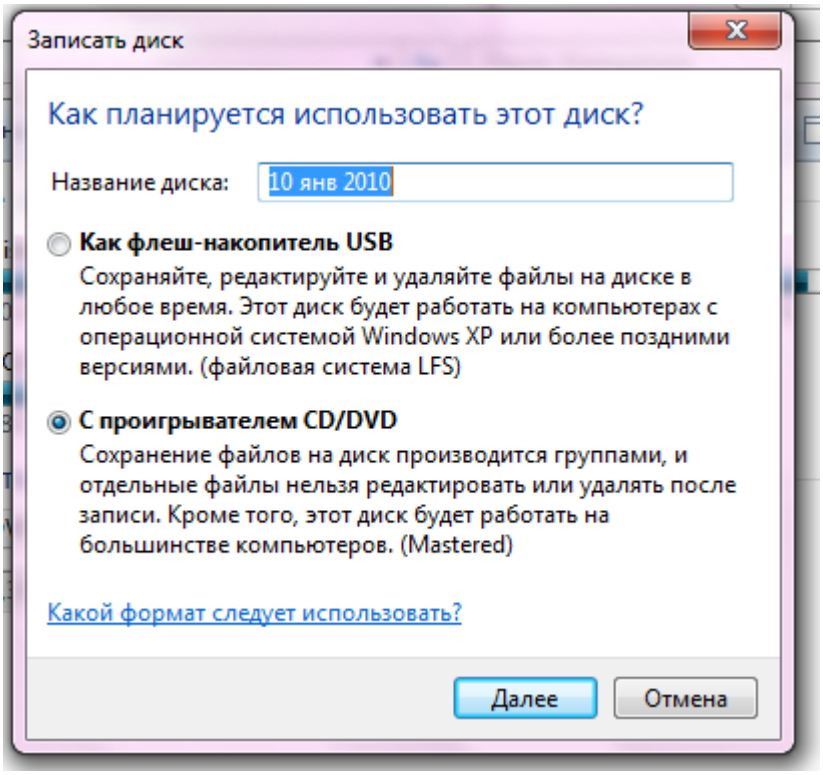


Рисунок 3.29. Подготовка диска к записи в Windows 7

Специализированные программы, предназначенные для записи CD и DVD дисков, часто имеют дополнительные возможности. Наиболее распространенной программой этого типа на текущий момент можно считать **Nero** фирмы **Ahead**. Последние версии ее (7, 8) кроме режимов записи дисков (Nero Burning ROM, Nero Express, Nero StartSmart) имеют в своем составе программы управления домашним медиацентром (запись теле- и радиовещания с тюнера и из Интернета); резервного копирования и восстановления (Backup) файлов, папок, логических дисков (на винчестер или оптический диск); записи и редактирования видео, фото и звука; просмотра DVD-фильмов; тестирования и настройки приводов оптических дисков и пр. (см. рисунок 3.30).

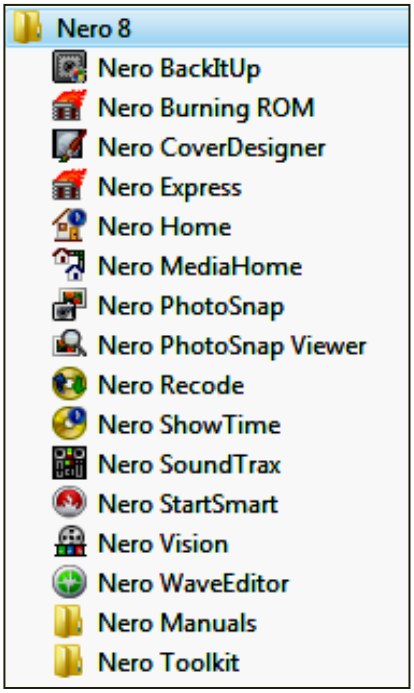


Рисунок 3.30. Меню системы Nero 8

При записи дисков возможны режимы:

- CD
  - ISO-диск (стандарт ISO 9660);
  - Аудио-CD

- Mixed-CD (файлы и аудио-записи)
- Копирование CD и создание образа диска
- Видео-CD
- Загрузочный CD
- UDF CD
- UDF/ISO CD

° DVD
- DVD-ROM (ISO)
- Копирование DVD
- DVD-видео
- DVD-ROM загрузочный
- DVD-ROM (UDF)
- DVD-ROM (UDF/ISO)

° HD DVD
- HD DVD-ROM (UDF, форматы 1.02, 1.50, 2.00, 2.01,2.50, 2.60)

° Blu-ray DVD
- Blu-ray Disc (UDF, форматы 1.02, 1.50, 2.00, 2.01,2.50, 2.60)

Программы воспроизведения DVD-фильмов

Поддержка воспроизведения DVD-фильмов появилась только в **Windows Vista Home Premium** и **Ultimate**. В предыдущих версиях **Windows** *Универсальный проигрыватель* (*Windows Media Player*) мог проигрывать DVD-диски только после установки дополнительных кодеков.

Наиболее широко для этой цели используется программа **Power DVD Player** корпорации **CyberLink**. Программа предназначена для воспроизведение мультимедиафайлов с CD, DVD дисков и с винчестера. Поддерживается воспроизведения видео с разрешением 720×480 для NTSC и 720×576 для PAL, 8-ми канальный звук, технология 24/96 LPCM вывода 24 битного звука с частотой выборки 96 кГц. Функция pan&scan позволяет просматривать видео формата 2.35:1 и 1.85:9 на полном экране с соотношением сторон 4:3 с сохранением пропорций изображения.

3.2.5 Создание и просмотр специальных форматов документов

К подобным форматам относятся широко используемые **PDF**-документы, **DjVu**-формат и новый формат **Windows Vista** – XPS (XML Paper Specification, см. раздел 3.1.3). Эти форматы создаются путем преобразования документов, подготовленных в текстовых редакторах, в новый формат, предназначенный, в основном, для чтения этих файлов. Чтение

файлов данных форматов (*.pdf, *.djvu, *.xps) может выполняться в специальных программах или в интернет-обозревателях (при установленных для этого дополнительных программных компонентах).

PDF-формат (Portable Document Format, формат документов корпорации Adobe Systems) – платформонезависимый формат, предназначенный для представления в электронном виде полиграфической продукции. Позволяет сохранять точный внешний вид исходного документа за счет внедрения необходимых шрифтов для текста, векторных и растровых изображений, форм и мультимедиаобъектов. Поддерживает цветовые схемы RGB и CMYK, несколько типов сжатия растровой информации.

Документы могут быть зашифрованы с заданием пароля на открытие документа или на его изменение, печать, копирование фрагментов и пр. (см. рисунок 3.31). Однако в Интернете легко можно найти программы, снимающие парольную защиту.

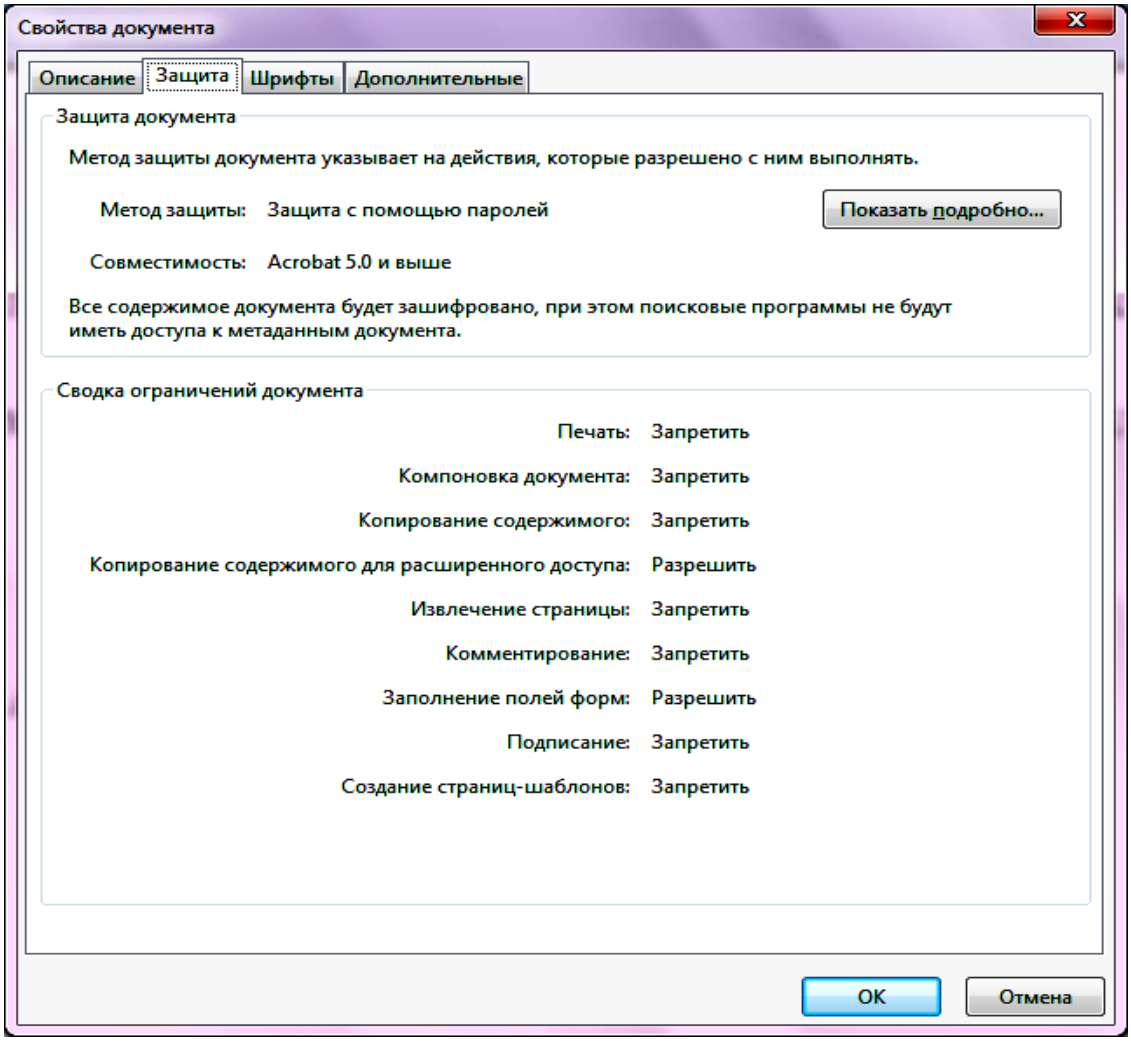


Рисунок 3.31. Параметры защиты **PDF**-документа

Для просмотра **PDF**-документов можно использовать бесплатную программу **Adobe Reader** или программы других разработчиков (например, **Foxit Reader**). Чтобы создавать, добавлять, пересматривать, редактировать и коллективно использовать информацию, содержащуюся в PDF-файлах следует использовать программы **Adobe Acrobat Professional** и **Adobe Acrobat Standard**.

DjVu – формат электронных документов, использующий программную технологию, сочетающую повышенную компрессию и высокое качество изображения. **DjVu** удобен для распространения сканированных документов, фотографий и прочих изображений через Интернет.

Существуют компьютерные программы просмотра файлов формата **DjVu** для различных платформ, в том числе в виде плагинов к браузерам, программных приложений для **PDA** устройств, многие из них являются бесплатными.

Степень сжатия изображений в **DjVu** значительно выше, чем в JPEG. Так же как и **PDF**, **DjVu** документ может содержать текстовый слой, что позволяет производить такие привычные операции как поиск по странице, копирование и вставка фрагментов текста и пр.

В настоящее время документы многих онлайн-библиотек, архивов технической документации и любительских Интернет-сайтов представлен в **DjVu**-формате.

3.3 Microsoft Office System

3.3.1 Общая характеристика компонентов системы

Система **Microsoft Office System** включает в себя серверы, службы, решения и приложения. Эти четыре компонента созданы для совместной работы при решении широкого круга бизнес-задач предприятий.

В системе представлены структурные блоки для создания решений, которые позволяют выполнять следующие действия:

- предоставляют сотрудникам подразделений удобный доступ к данным предприятия;
- обеспечивают эффективную и гибкую совместную работу групп и организаций;

- дают возможность большому количеству сотрудников подразделений организации участвовать в создании бизнес-сред растущей сложности.

Система **Microsoft Office System** имеет в своем составе следующие **приложения**:

Microsoft Office Access – СУБД для работы с собственным форматом реляционной базы данных и с сервером баз данных ***Microsoft SQL Server***;

Microsoft Office Communicator – позволяет пользователям легко общаться и работать с другими пользователями в различных географических пунктах с помощью обмена мгновенными сообщениями, общиго доступа к рабочему столу, голосовой и видеосвязи;

Microsoft Office Excel – табличный процессор для выполнения математических вычислений и оформления результатов;

Microsoft Office FrontPage – создание веб-узлов и управление ими, разработка веб-страниц;

Microsoft Office Groove – обеспечение совместной работы пользователей за счет создания общих рабочих областей групп; позволяет синхронизировать процесс обработки данных участниками таких групп при работе в Интранете и Интернете, либо при подключении к рабочей области после оффлайновой работы;

Microsoft Office InfoPath – создание многофункциональных динамических бизнес-форм для сбора данных и обмена ими внутри широкого круга бизнес-процессов различной сложности на основе языка *XML* и программирования на Jscript или VBScript. Источниками данных могут быть база данных системы **Microsoft** SQL Server, **Microsoft Access**, файл XML-данных или веб-служба;

Microsoft Office OneNote – обеспечивает возможность записи данных в различной форме на переносном, настольном или планшетном компьютере, включая вводимые с помощью клавиатуры и рукописные заметки, нарисованные от руки схемы, звукозаписи, фотографии и рисунки из Интернета, а также данные из других приложений.

Microsoft Office Outlook – диспетчер связей и органайзер: работа с электронной почтой, календарь для планирования встреч и совещаний, база адресов, список задач и заметок;

Microsoft Office PowerPoint – дизайнер презентаций;

Microsoft Office Project – управление корпоративными проектами; использование эффективных средств анализа данных и создания отчетов для оптимизации использования ресурсов, определения приоритетных направлений работы и обеспечения соответствия проектов общим бизнес-целям;

Microsoft Office Publisher – создание публикаций для печати, веб-узлов и электронной почты на основе шаблонов;

Microsoft Office SharePoint Designer – создание и настройка веб-узлов **Microsoft SharePoint** и разработки интерактивных решений по организации рабочих процессов с помощью технологий SharePoint – создание настраиваемых представлений данных, отчетов, средств, в том числе отображающих данные из нескольких источников;

Microsoft Office Visio – работа с бизнес-диаграммами и техническими диаграммами;

Microsoft Office Word – текстовый процессор; может также использоваться для редактирования HTML и XML файлов.

Дополнительные приложения:

- **Microsoft Office Picture Manager** – просмотр, редактирование и печать файлов изображений форматов JPEG, GIF, BMP, PNG, TIFF, WMF, EMF;
- **Microsoft Office Document Imaging** – сканирование и просмотр изображений. Для настройки и выполнения сканирования использует компонент **Microsoft Office Document Scanning**.

Система **Microsoft Office System** имеет в своем составе следующие **серверы**:

- **Microsoft Office Forms –** расширяет взаимодействие с клиентами, партнерами и поставщиками с помощью простых в использовании электронных форм;
- **Microsoft Office Groove Server** – предоставляет возможности для определения присутствия и немедленного обмена сообщениями, пе-

редачи файлов, организации видео- и аудиоконференций, ведения журналов, архивирования.

- **Microsoft Office Project Server** – платформа для поддержки функций управления корпоративными проектами и ресурсами, а также совместной работы в рамках решений для корпоративного управления проектами. Решение для корпоративного управления проектами – **Microsoft Office Enterprise Project Management** (EPM) включает дополнительный компонент **Microsoft Office Project Web Access**.
- **Microsoft Office SharePoint Server** – позволяет организовать централизованный доступ ко всей ключевой деловой информации и приложениям, а также обеспечить распределенное использование информации файловых серверов, баз данных, общих папок, интернет-сайтов, а также веб-сайтов на базе Microsoft SharePoint Team Services.

Корпорация Microsoft поставляет следующие основные дистрибутивные наборы приложений Office System 2007 (*http://office.microsoft.com/en-us/products/FX101635841033.aspx*):

Приложения Microsoft Office System 2007	Дистрибутивные наборы, их цена и состав					
	Home and Student	**Standard**	**Small Business**	**Professional**	**Ultimate**	**Enterprise**
	$149.95	$399.95	$449.95	$499.95	$679.95	-
Word	●	●	●	●	●	●
PowerPoint	●	●	●	●	●	●
Excel	●	●	●	●	●	●
OneNote	●				●	●
Outlook		●				
Outlook with Business Contact Manager			●	●	●	●
Access				●	●	●
Accounting Express			●	●	●	
Publisher			●	●	●	●
Groove					●	●
InfoPath					●	●
Communicator						●

Система **Microsoft Office System** имеет в своем составе следующие **сервисы**:

- **Microsoft Office Live Meeting** – сервис для проведения веб-конференций. Позволяет в реальном времени общаться с партнерами, коллегами и заказчиками, в группах до 2000 участников.

- **Microsoft Office Online** – веб-ресурс для продуктов **Microsoft Office System**. Содержит статьи, материалы справочной системы, обновления и исправления для приложений, описание интегрированных Интернет-служб партнеров, наборы картинок, шаблонов, бланков деловых документов, пошаговые обучающие курсы и многое другое.

Пользовательский интерфейс приложений **Microsoft Office** до версии 2007 показан на рисунке 3.32.

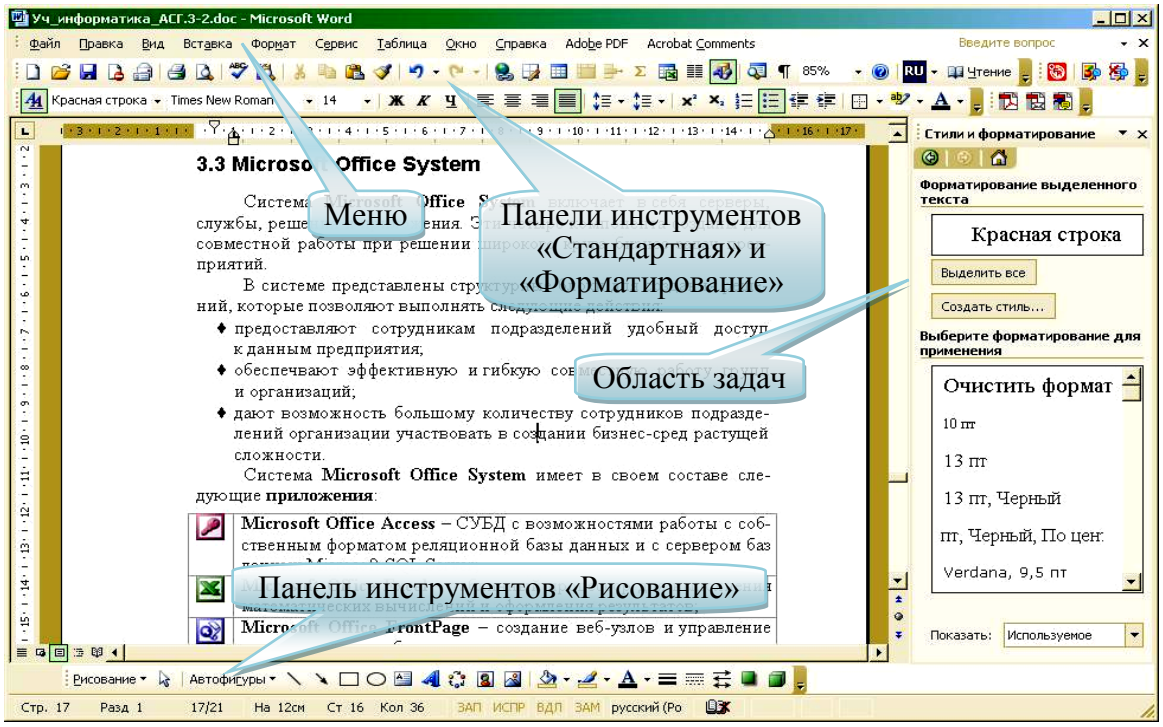


Рисунок 3.32. Интерфейс Microsoft Office Word 2003

Во время работы использовались системы меню, панели инструментов, области задач и диалоговые окна. При этом существовала избыточность интерфейса: некоторые команды могли выполняться с помощью панели инструментов и меню системы, а также контекстного меню (все они были настраиваемыми).

Версия **Microsoft Office 2007** кардинальным образом отличается от предыдущих версий по пользовательскому интерфейсу. В новой версии дублирование команд убрано. Традиционные меню и панели инструментов заменены лентой (см. рисунок 3.33).

Лента – новый вид интерфейса, в виде набора вкладок, каждая из которых содержит набор сгруппированных элементов для различных команд и управления настройками при работе.

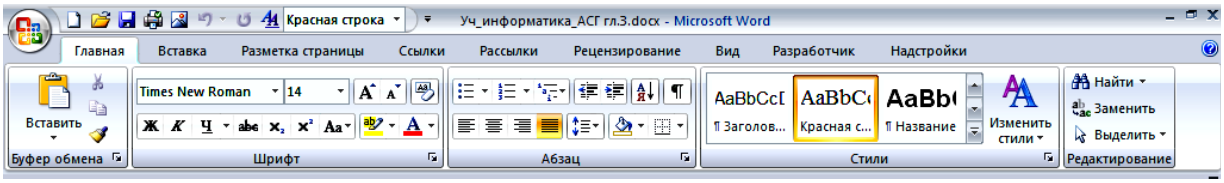


Рисунок 3.33. Вкладка ленты Главная Microsoft Office Word 2007

В **Word 2007** имеются постоянные вкладки ленты **Главная**, **Вставка**, **Разметка страницы**, **Ссылки**, **Рассылки**, **Рецензирование**, **Вид**, **Разработчик** для набора текста, вставки, разметки страницы, работы со ссылками, почтовых рассылок, рецензирования документов и работы с формами, макросами и скриптами. В **Excel 2007** имеет похожий набор вкладок ленты с командами для работы с электронными таблицами: создание новых листов, добавление таких объектов, как диаграммы и рисунки, разметка страницы, работа с формулами, управление данными и рецензирование. По замыслу разработчиков системы, новый интерфейс должен увеличить простоту и скорость доступа к элементам управления работой и оформлением документов. Однако, переход со старых версий, если пользователь длительное время пользовался ими, к новой, достаточно сложен, т. к. привычное расположение команд настройки полностью изменилось. Для освоение нового интерфейса требуется достаточно много времени, после чего его преимущества действительно становятся реальны.

Приблизиться к некоторым привычным элементам в интерфейсе позволяет настраиваемая панель быстрого доступа, которая может располагаться в заголовке окна справа от главной круглой кнопки окна (которая содержит все же меню с важными командами!) или под лентой.

В настройках панели быстрого доступа, показанных на рисунке 3.34, добавлены кнопки для часто используемых команд **Создать**, **Быстрая печать**, **Вставить рисунок из файла**, **Стили**, а также **список Стиль**, отображающий текущий стиль и позволяющий быстро задать новый.

К привычным элементам управления и настройки относятся контекстные меню, появляющиеся при щелчке правой кнопкой мыши на элементах в окне системы.

К новым элементам интерфейса относится мини-панель инструментов, которая автоматически появляется полупрозрачной над выделенным фрагментом текста и становится более четкой при наведении на неё курсора. Её можно использовать для форматирования выделенного фрагмента: применения полужирного начертания или курсива, изменения размера и цвета шрифта и т. д. При удалении от нее курсора мини-панель плавно исчезает.

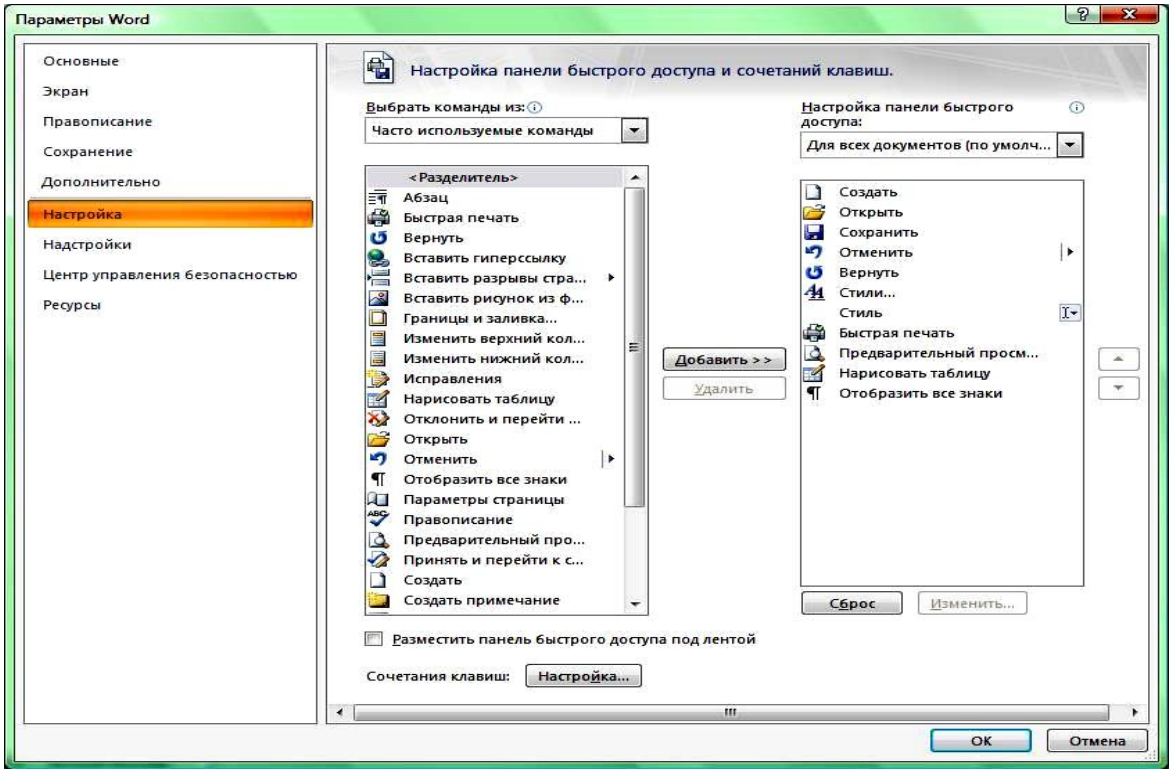


Рисунок 3.34. Настройка панели быстрого доступа

Кроме набора основных вкладок ленты, новая версия **Office** имеет контекстные вкладки, появляющиеся при работе с объектами, которые могут быть созданы с использованием вкладки ленты **Вставка**:

- Работа с таблицами (**Конструктор**, **Макет**);
- Работа с рисунками (**Формат**);
- Средства рисования (**Формат**);
- Работа с рисунками SmartArt (**Конструктор**, **Формат**);
- Работа с диаграммами (**Конструктор**, **Макет**, **Формат**);
- Работа с колонтитулами (**Конструктор**);
- Работа с надписями (**Формат**);
- Работа с объектами WordArt (**Формат**);
- Работа с формулами (**Конструктор**).

Например, при работе с таблицами, появляется надпись на заголовке окна **Работа с таблицами** над двумя новыми вкладками – **Конструктор** и **Макет** (см. рисунок 3.35).

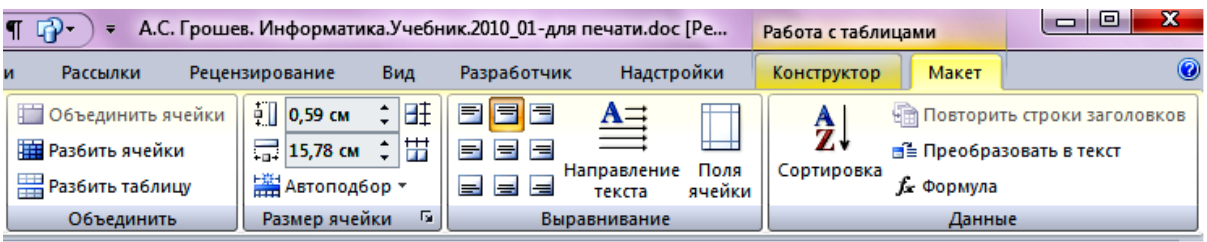


Рисунок 3.35. Контекстно-зависимые ленты работы с таблицами

Вместо сложных диалоговых окон с многочисленными параметрами настройки различных объектов (стилей, таблиц, диаграмм и пр.) при работе с документом, электронной таблицей, презентацией или базой данных предлагается использовать галереи для получения необходимых результатов. Представляя простой набор возможных вариантов, галереи упрощают создание профессионально оформленных работ. Пользователи, которым требуется большая степень контроля результатов работы, могут по-прежнему использовать традиционные диалоговые окна.

Благодаря новой технологии динамического предварительного просмотра пользователь может увидеть результаты изменения формата, наведя указатель мыши на вариант оформления, содержащийся в галерее. Эта новая динамическая возможность позволяет в процессе разметки, редактирования и форматирования затрачивать меньше времени на выбор необходимого варианта оформления (см. рисунок 3.36).

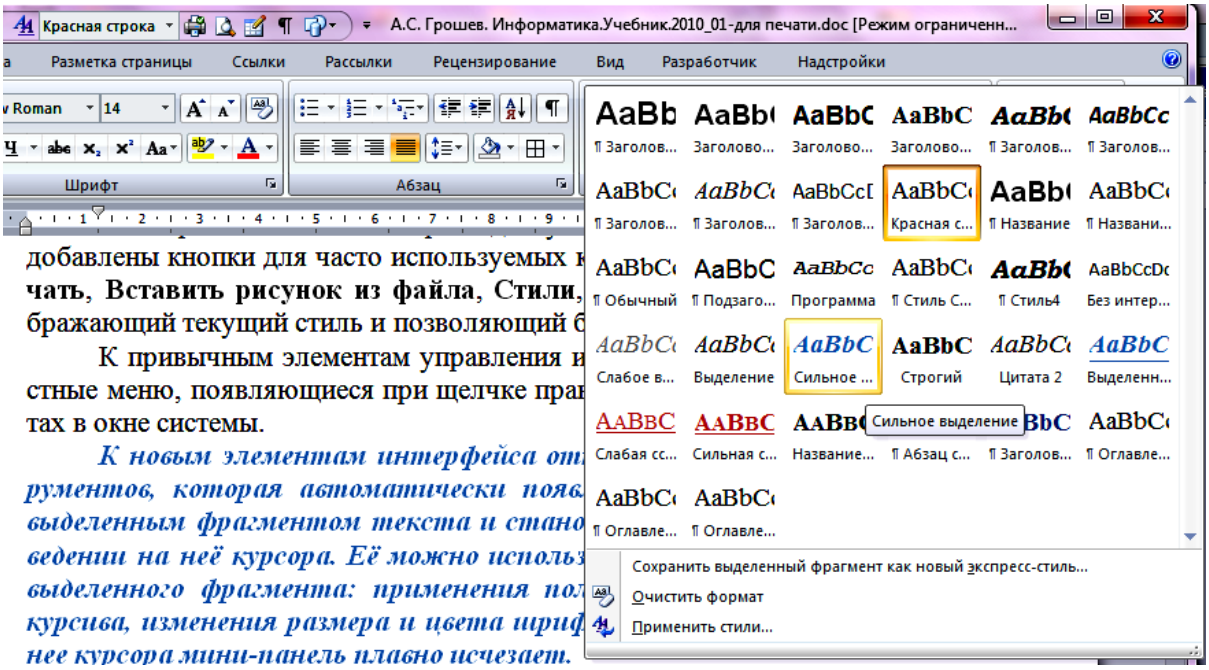


Рисунок 3.36. Галерея выбора стиля с динамическим изменением стиля текущего абзаца текста

3.3.2 Подготовка текстовых документов в системе Microsoft Office Word

Текстовый процессор **Microsoft Office Word** может быть использован для подготовки документов сложной структуры, состоящих из большого количества глав, разделов, подразделов, имеющих таблицы,

сложные математические, химические и пр. формулы, рисунки и диаграммы, которые могут создаваться в данной системе или добавляться, как объект другой системы с помощью технологии OLE.

Эти документы должны соответствовать определенным стандартам, обязательным при оформлении деловых документов и научных отчетов, или строгим правилам оформления статей для различных журналов, текстов научных монографий для издательств, стилю оформления веб-страниц различных сайтов. Требования стандартов на научные отчеты и требования различных издательств могут сильно отличаться друг от друга. В связи с этим высокую актуальность имеет правильная технология подготовки документов с использованием стилей оформления и шаблонов документов.

Стиль – именованный набор параметров оформления абзаца, знака или таблицы.

Стиль определяет вид шрифта, параметры абзаца, табуляции, границ; язык, текст без рамки или в рамке, абзац без нумерации или нумерованный и горячие клавиши задания данного стиля (см. рисунок 3.37).

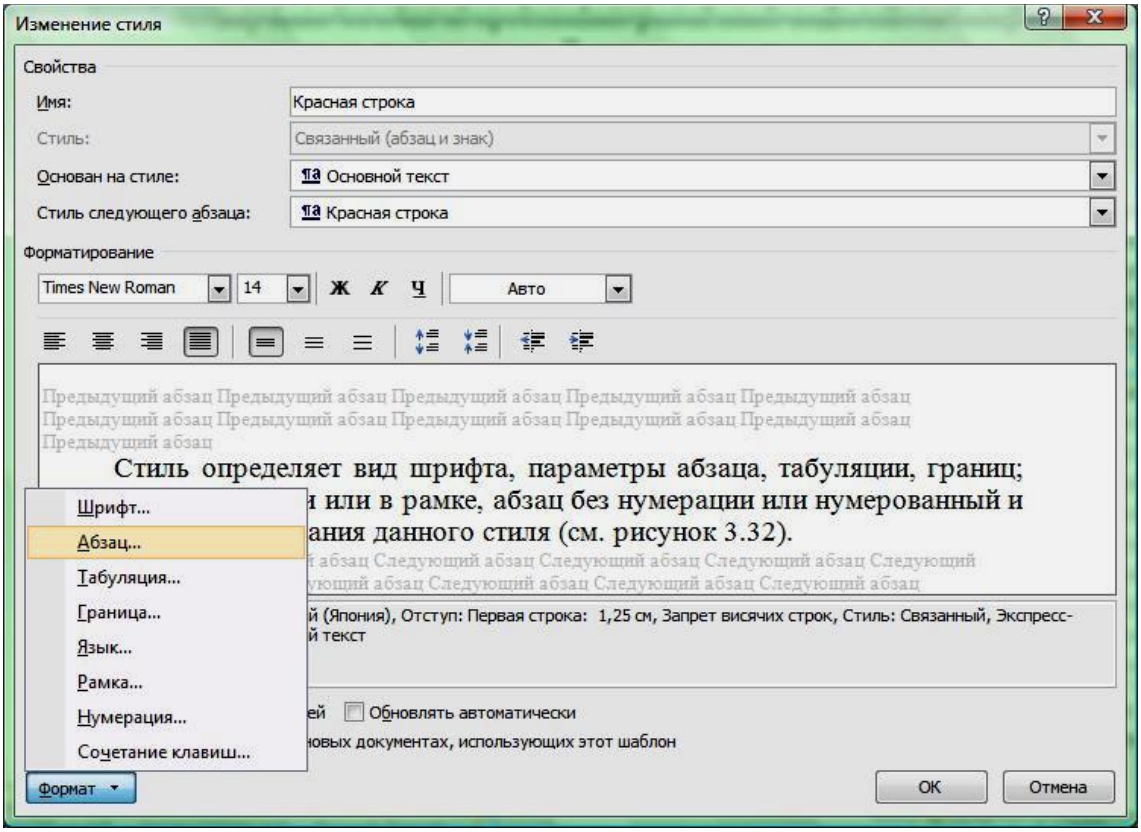


Рисунок 3.37. Окно настройки стилей

Следует знать и правильно применять основные параметры абзацев в стилях: выравнивание (по левому краю, по центру, по правому краю, по двум сторонам), отступы (слева, справа, первой строки), интервалы (перед абзацем, после него, межстрочный), а также положение на странице – запрет отрывать первую или последнюю строки, разрывать абзац между страницами, переносить слова по слогам, нумеровать строки; начинать абзац с новой страницы (см. рисунок 3.38).

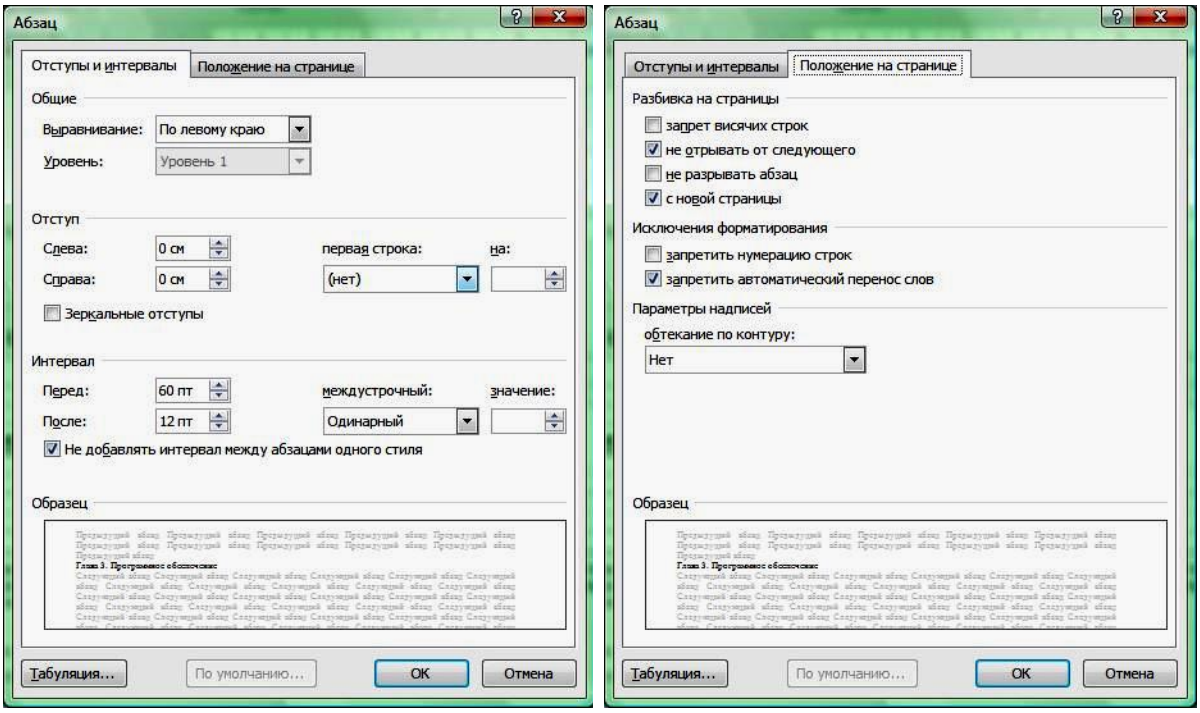


Рисунок 3.38. Окно настройки параметров абзаца (стиль **Заголовок 1**)

При создании нового документа в приложении **Microsoft Office Word 2007** команда круглой кнопки **Office ► Создать** открывает диалоговое окно с вариантами:

- новый документ;
- XML- документ;
- на основе шаблонов
  - отчеты
  - письма
  - резюме
  - факсы.

В **Office Word 2003** команда меню **Файл ► Создать…** предлагает в области задач следующие варианты:

- новый документ;
- XML- документ;
- веб-страница;
- сообщение электронной почты;

- из имеющегося документа;
- с использованием шаблонов на компьютере или на веб-узлах **Microsoft**.

После инсталляции **Microsoft Office Word 2003** имеет достаточно много шаблонов документов, которые сгруппированы по разделам:

- публикации (брошюра, диссертация, руководство, справочник);
- отчеты (изысканный, современный, стандартный);
- записки;
- письма и факсы;
- документы слияния;
- прочие.

Все шаблоны имеют собственные определенные стили оформления, которые при необходимости можно отредактировать. Можно изменить также вид самого шаблона и использовать его далее в своей работе над документами.

Значительный интерес для желающих научиться правильно работать со сложными документами представляют шаблоны системы **Microsoft Office 2003 Руководство**, **Брошюра** (правильнее назвать **Буклет**) и **Справочник** из раздела **Публикации**. Открыв эти шаблоны, можно познакомиться с основными инструкциями по использованию шаблонов и стилей в документах, оформлению буклета на одном листе бумаги, использованию символов в качестве рисунков, созданию оглавлений и алфавитных указателей.

Шаблоны раздела **Документы слияния** являются мастерами, которые помогают освоить оформление и рассылку писем или факсов по адресам, выбранным из таблиц существующих файлов или из адресной книги **Outlook**.

Шаблоны раздела **Отчеты** содержат краткую инструкцию по созданию оглавлений и указателей, маркированных и нумерованных списков, таблиц, колонтитулов, сносок.

В системе **Microsoft Office Word 2007** отсутствуют указанные выше шаблоны с краткими инструкциями по использованию стилей и других элементов оформления.

При оформлении документов может использоваться также компонент оформлением документов – **Тема документа**. В системе **Office** до 2007 эта настройка присутствует в меню **Формат ► Тема**, в 2007 – на ленте **Разметка страницы**.

Тема системы Office – набор унифицированных элементов, определяющих внешний вид документа с помощью цвета, шрифтов и графических объектов.

Тема документа представляет собой набор вариантов форматирования, включающих цветовую тему (набор цветов), тему шрифтов (стили заголовков и основного текста), тему эффектов (набор линий и заливок), настройки списков, гиперссылок и границ таблиц.

Такие приложения, как **Microsoft Office Word**, **Excel** и **PowerPoint** имеют несколько заранее определенных тем документов, но пользователь может также создавать и свои темы путем настройки существующих тем документов, а затем сохранять их в виде пользовательских тем документов. Темы документов используются приложениями **Office 2007** совместно, так что все документы **Office** могут иметь единообразный вид.

Методика подготовки текстового документа

1. Задать параметры страницы

На ленте **Разметка страницы** (версии до 2007 – в меню ***Файл ► Параметры страницы…***) следует задать для страницы размер, ориентацию и поля – левое, правое, верхнее и нижнее. При необходимости можно выбрать тему документа или задать количество колонок на листе, цвет страницы и текст подложки, границы страницы.

2. Определить параметры необходимых стилей

На ленте **Главная** открыть окно стилей (до 2007 – на панели **Форматирование** кнопка слева от раскрывающегося списка стилей открывает панель настройки стилей в области задач или меню **Формат ► Стили и форматирование…**). Вид окна стилей показан на рисунке 3.39.

Следует учитывать, что список стилей может быть неполный, в ссылке ***Параметры…*** можно выбрать режимы **Рекомендованные**, **Используемые**, **В текущем документе**, **Все**. (В версиях до 2007 список **Все** настраивался при выборе режима **Показать: Специальное**….)

Для выбранного стиля можно открыть список задач (см рисунок 3.39). Полные возможности по настройке стилей дает команда **Изменить**, диалоговое окно которой показано ранее на рисунке 3.37.

Кнопка **Создать стиль** позволяет определить параметры стиля с новым названием на основе текущего.

Для ского формирования лавления сложного документа (с использованием ленты **Ссылки** или в версиях до 2007 – пункта меню ***Вставка ► Ссылка ► Оглавление и указатели…***) следует определить параметры стилей **Заголовок 1** (уровень 1), **Заголовок 2** (уровень 2), **Заголовок 3** (уровень 3) и т. д., если необходимо.

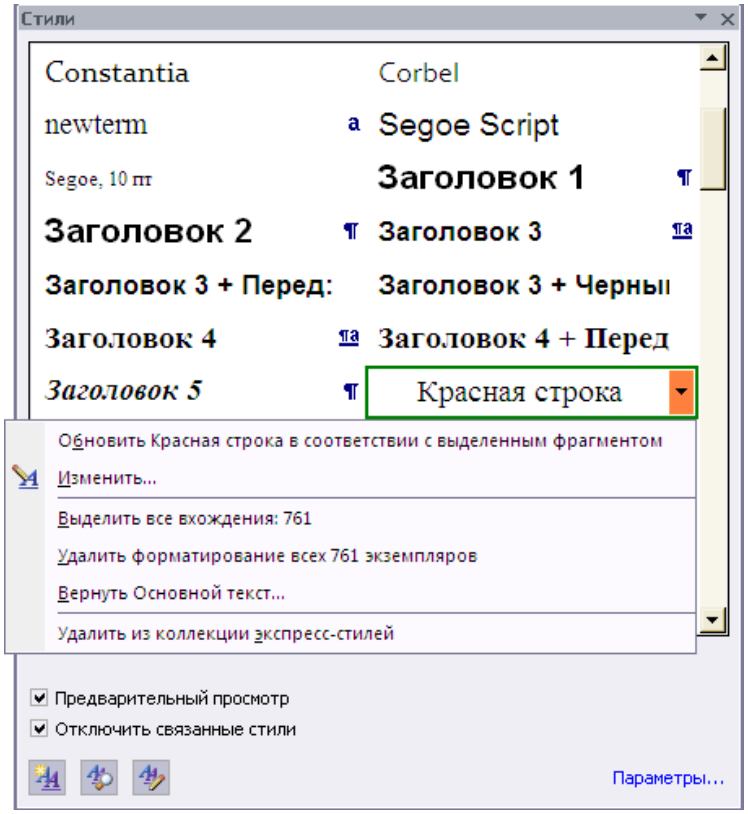


Рисунок 3.39. Окно «Стили»

Для основного текста следует определить параметры основного абзаца. При этом следует учитывать, что традиционный русский стиль – абзац с красной строкой (первая строка с отступом), без интервалов до и после абзаца, без отступов слева и справа. Стиль некоторых новых журналов, близкий к западным стилям – первая строка абзаца без отступа, но добавлен интервал после абзаца.

Не следует изменять стиль абзаца по умолчанию **Обычный**, который является основой стиля **Обычная таблица**, иначе в таблицах могут появиться нежелательные отступы первой строки, слева и справа; интервалы сверху и снизу в ячейках, увеличенный межстрочный интервал и пр.

Если удалить какой-либо стиль абзаца, то ко всем абзацам, которые были оформлены удаленным стилем будет применен стиль **Обычный.** Аналогично действует команда **Очистить формат.**

При определении внешнего вида документа может быть использована **Тема документа**. По умолчанию никакая **Тема** не используется, при задании темы параметры стилей документа будут переопределены.

Если в документе используется много таблиц, можно определить стиль таблицы, который будет использоваться по умолчанию (в меню ***Таблица ► Автоформат таблицы…*** или в области задач **Стили и формати-**

рование). После этого при создании таблицы с помощью кнопки **Вставить таблицу** панели инструментов **Стандартная** она будет иметь стиль таблицы по умолчанию.

3. Задать колонтитулы

Если необходимо, можно определить текст верхнего и нижнего колонтитулов. Для этого следует воспользоваться лентой Вставка или в версиях до 2007 командой меню ***Вид*** ► ***Колонтитулы***. Кроме постоянного текста колонтитулы могут содержать поля, показывающие информацию текущего файла: номер страницы, автор, дата создания, имя файла, всего страниц и пр.

Колонтитулы первой страницы, четных и нечетных страниц могут различаться, что определяется на контекстной ленте **Работа с колонтитулами** или до версии 2007 в настройках **Параметры страницы**.

Каждый раздел документа может иметь собственный колонтитул. Это можно использовать для нумерации страниц альбомной ориентации, оформленных как отдельные разделы (например, для широких таблиц или рисунков) при книжной ориентации основных страниц. При альбомной ориентации номер страницы должен иметь иную ориентацию, чем при книжной. Например, если обычная страница имеет автоматическую нумерацию в правом нижнем углу, альбомная страница того же документа (при печати на одной стороне бумаги) должна иметь номер в левом нижнем углу, развернутый по часовой стрелке на 90° (см. рисунок 3.40).

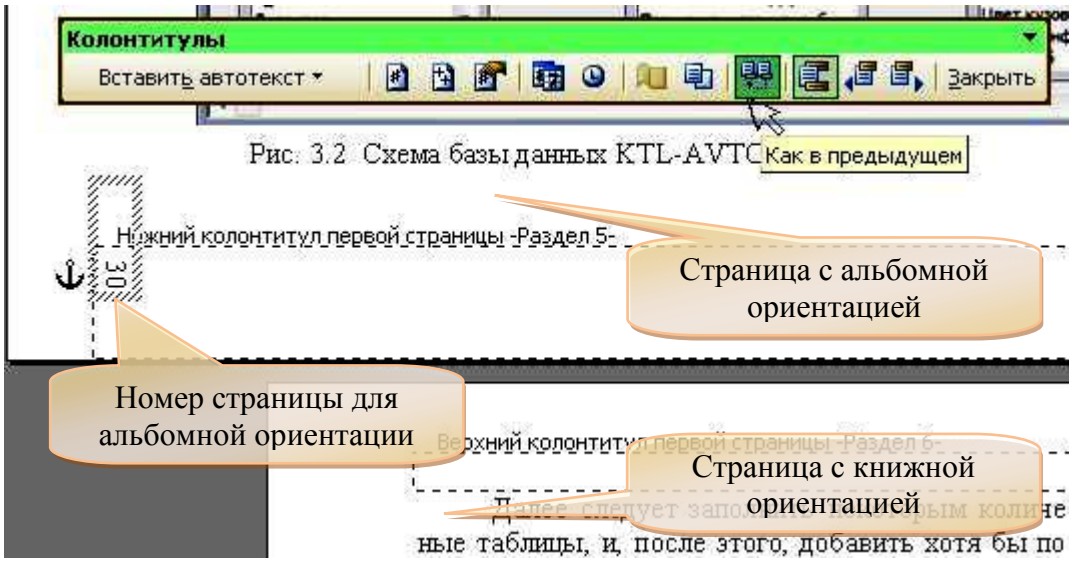


Рисунок 3.40. Номер страницы в колонтитуле для раздела с альбомной ориентацией в версии Word 2003

Это можно сделать, если для колонтитула нового раздела убрать свойство **Как в предыдущем** (кнопкой на панели инструментов **Колонтитулы**), после чего командой меню ***Вставка ► Номера страниц…*** вставить поля номеров страниц отдельно в разных разделах в нужном месте. Для колонтитулов альбомных страниц в меню (Word до 2007) командой ***Формат ► Направление текста*** задать направление сверху-вниз (см рисунок 3.40).

В версии **Word 2007** номер страницы командой ленты **Вставка – Номер страницы** размещается в колонтитуле без рамки. Поэтому для номеров с иной ориентацией, чем обычно, нужно сначала разместить в колонтитуле объект **Надпись**, потом в него (в текущее положение) вставить поле **Номер страницы**, далее можно развернуть номер в нужном направлении и убрать линии у рамки надписи (см. рисунок 3.41).

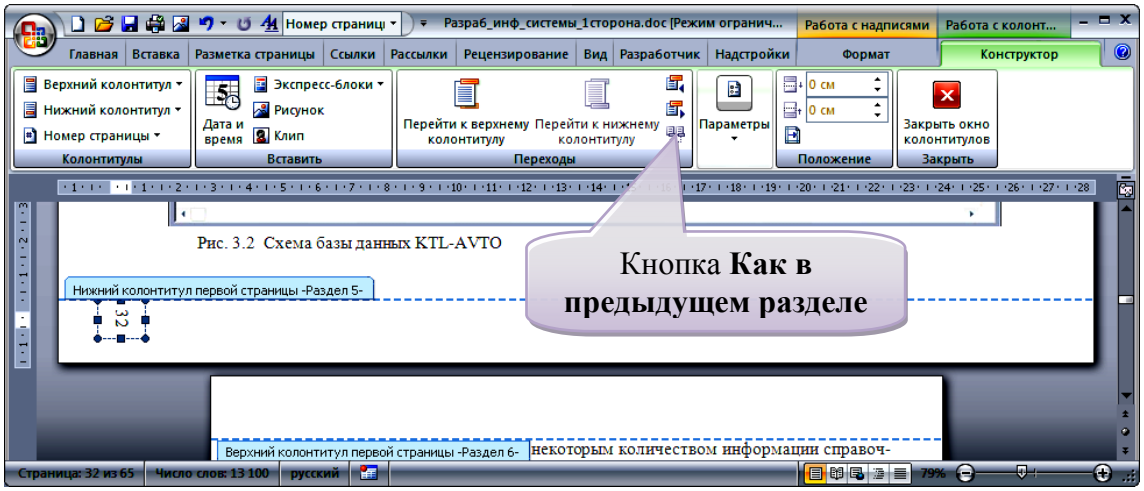


Рисунок 3.41. Работа с колонтитулами и задание номера страницы для раздела с альбомной ориентацией в версии Word 2007

4. Приступить к набору текста

Перед началом набора текста можно определить, будет система автоматически переносить слова по слогам при достижении границы строки в абзаце или нет (по умолчанию переносы отключены). Делается это в Word 2007 на вкладке ***Разметка страницы – Расстановка переносов***, в Word 2003 – меню ***Сервис – Язык – Расстановка переносов***.

Если документ начинается с титульного листа или обложки, на них не следует использовать стили заголовков 1 – 3 уровня, которые обычно входят в оглавление при его автоматическом формировании.

Основной текст документа обычно начинается с заголовка 1 уровня, стиль его лучше задать перед началом написания этого абзаца. В конце заголовков точек не ставят, в стилях заголовков следует задать запрет автоматического переноса слов. Если заголовок должен быть написан прописными буквами, это следует задать в настройках стиля для раздела **Формат ► Шрифт**. Часто для заголовков задают отступ слева и справа 1-1,5 см, обычно интервал до абзаца и после, выравнивание может быть по центру или по левому краю. Стиль заголовков 2 и 3 уровня может быть основан на стиле Заголовок 1, но несколько модифицирован.

Для абзацев основного текста следует задать стиль с отступом первой строки или без него, но с интервалом после абзаца. В тексте могут присутствовать абзацы иного стиля, чем основной текст (например, аннотация в начале статьи или выделенный текст).

При необходимости можно использовать оформление текста в несколько колонок для всего документа или для одного раздела.

При вводе текста параметры абзаца могут быть изменены с использованием лент **Главная ►** разделы **Абзац**, **Шрифт** и **Разметка страницы ►** раздел **Абзац.**

Таблицы в тексте могут занимать всю ширину страницы или часть ее с обтеканием текстом или без обтекания. Ширина таблицы может быть задана в процентах от ширины страницы (без полей) или в сантиметрах (или других единицах, определенных в параметрах **Office**). Следует помнить, что в свойствах таблицы можно задать поля для ее ячеек (верхнее, нижнее, левое и правое), вертикальное выравнивание в ячейке, направление текста (слева-направо, снизу-вверх или сверху-вниз) и параметры абзацев в ячейках таблицы. Для многих операций при работе с таблицей удобно использовать контекстное меню, однако полный набор операций присутствует в двух контекстных лентах – **Конструктор** и **Макет** или в версиях до 2007 – в пункте меню ***Таблица***.

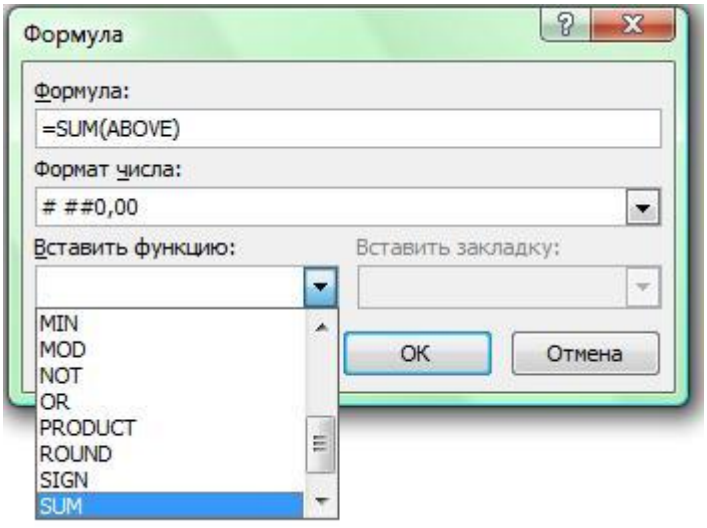


Рисунок 3.42. Окно построителя формул для таблицы Word

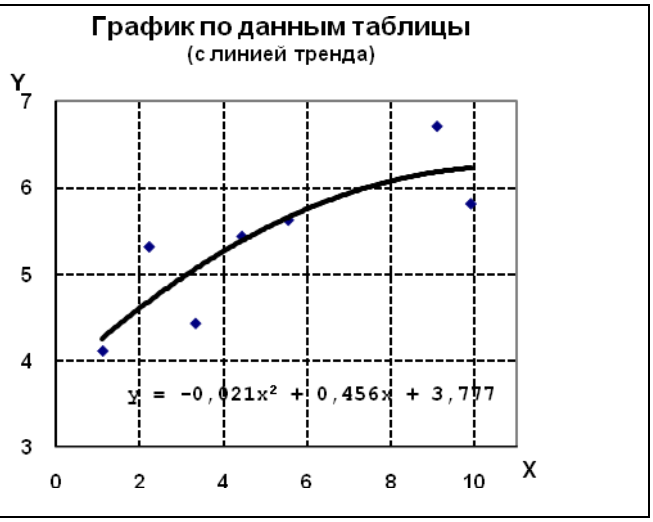
В ячейки таблицы можно вставлять поля с формулами для простейших математических вычислений по данным таблицы (лента **Вставка**

► группа **Текст** ► кнопка **Экспресс-блоки** ► команда **Поле**, в окне **Поле** кнопка **Формула** - см. рисунок 3.42). В версиях **Word** до 2007 команда **Поле…** присутствует в пункте меню **Вставка**.

В таблице 3.11 показано использование функций вычисления количества (**COUNT**), среднего значения (**AVERAGE**) и суммы по строкам (**SUM**). Если выделить столбцы числовых значений X и Y таблицы, по ним можно построить диаграмму в **Word 2003** командой меню **Вставка ► Объект**… ► **Диаграмма Microsoft Graph** (см. таблицу 3.11).

Таблица 3.11. Пример использования формул и **Microsoft Graph** в Word

X	Y	=SUM (LEFT)
1,11	4,11	5,22
2,22	5,32	7,54
3,33	4,43	7,76
4,44	5,44	9,88
5,55	5,63	11,18
9,11	6,72	15,83
9,92	5,82	15,74
7	5,35	15,83
=COUNT (ABOVE)	**=AVERAGE (ABOVE)**	**=MAX (ABOVE)**



В Word 2007 команда ленты **Вставка ► Диаграмма** открывает окно выбора типа диаграммы, после чего происходит временный переход в новое окно системы **Excel** с названием **Диаграмма в Microsoft Office Word**, предназначенное для задания (или расчета) исходных данных для диаграммы. После того, как мы закроем окно **Excel**, мы увидим диаграмму в исходном документе. Все дальнейшие настройки диаграммы выполняются с использованием трех лент группы **Работа с диаграммами ► Конструктор**, **Макет** и **Формат**. На первой из них есть кнопка **Изменить данные**, которая снова откроет при необходимости таблицу **Excel**. В контекстном меню для диаграммы, как и в предыдущих версиях **Word**, имеется команда Добавить линию тренда для различных типов аппроксимации и сглаживания экспериментальных данных. Те же возможности – на ленте **Макет**, кнопка **Анализ**.

Иногда таблицы без разлиновки используются для размещения структурированного текста на странице документа.

Рисунки могут быть добавлены в текст из файлов графических форматов или созданы с помощью команды ленты **Вставка ► Фигуры** или

панели **Рисование Word 2003** (рисунки векторного типа, состоящий из отдельных объектов). При создании рисунков может использоваться полотно, которое позволяет выполнять операции со всеми объектами, размещенными в нем (перемещение всего рисунка, изменение масштаба, задание положения полотна). Положение рисунков может быть в тексте, перед ним или за ним, с обтеканием текстом вокруг рамки, контура или сверху и снизу.

В системе Word 2003 рисунок в формате *.bmp, скопированный из редактора Paint в буфер обмена, можно вставить в документ через пункт меню ***Правка ► Специальная вставка***... с заданием **Вставить как Bitmap Image** (объект), в этом случае рисунок можно будет редактировать прямо в документе, куда он вставлен (в редакторе **Word** запускается **Paint**).

Если рисунок или таблица имеют название, их можно написать в виде объекта **Рамка**, либо воспользоваться командой ленты **Вставка ► Надпись** или до версии Word 2007 – команда меню ***Вставка ► Ссылка ► Название…*** (этот режим позволяет также выполнять автоматическую вставку надписей с нумерацией таблиц и рисунков).

Несколько небольших рисунков могут быть размещены в ячейках нелинованной таблицы, при этом название рисунка или группы рисунков также можно разместить в отдельных ячейках таблицы (или в одной ячейке, объединенной из нескольких по горизонтали).

Если предполагается автоматическое формирование списка таблиц и рисунков с помощью команды ленты **Ссылки ► Названия ►** кнопка **Список иллюстраций** или до версии 2007 – командой меню ***Вставка ► Ссылка ► Оглавление и указатели… ► Список иллюстраций***, не следует писать названия в графических объектах типа **Надпись** и пр., такие названия не попадут в сформированный список.

Формулы – математические, химические и другие могут быть вставлены в текст с использованием ленты **Вставка ►** группа **Символы ►** команда **Формулы**, или как объект **Microsoft Equation** (редактор формул) в **Word 2003** и более ранних (а также и в Word 2007). В версии **Word 2007** команда ***Вставка … Формулы*** содержит достаточно много примеров написания формул:

- ✓ бином Ньютона $$(x+a)^n = \sum_{k=0}^{n} \binom{n}{k} x^k a^{n-k}$$
- ✓ квадратное уравнение $$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

✓ ряд Фурье $$f(x) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos\frac{n\pi x}{L} + b_n \sin\frac{n\pi x}{L} \right)$$

и пр.

Лента **Работа с формулами ► Конструктор** и объект **Microsoft Equation** содержат многочисленные шаблоны для написания различных математических символов, дробей, индексов, интегралов, сумм и пр. Пример формулы, написанной с помощью **Microsoft Equation**:

$$\frac{d\psi}{d\varphi} = -\sqrt{\frac{\cos\varphi_0 - \cos\varphi_1}{b\left(ctg^2\varphi_0 - ctg^2\varphi_1\right)}} \frac{1}{\sin^2\varphi\sqrt{c - \cos\varphi}}$$

5. Вставить оглавление и указатели

Для сложного структурированного документа Word предоставляет полезные возможности формирования оглавления, списка таблиц и рисунков, предметного указателя.

Как указывалось выше, для автоматического формирования оглавления можно все заголовки, включаемые в оглавление, оформлять стилями соответствующего уровня. Если этого своевременно не сделано, в **Word 2003** можно задать уровни заголовков с помощью панели инструментов **Структура**, после чего сформировать оглавление (меню ***Вставка ► Ссылка ► Оглавление и указатели***, вкладка окна **Оглавление**). Есть и другие способы создания оглавления, описанные в разделе справки «Создание оглавления».

В **Word 2007** оглавление формируется с помощью группы **Оглавление** на ленте **Ссылки**, которая позволяет сформировать автособираемое оглавление выбранного стиля, создать ручное оглавление выбранного стиля или задать уровни заголовков для абзацев (**Добавить текст** ► Уровень 1, 2, 3).

Для формирования списка таблиц и рисунков можно использовать оформление их особыми стилями, например **Название таблицы** и **Название рисунка**.

Методы создания предметного указателя описаны в разделе справки «Создание предметного указателя», один из способов – пометить слова предметного указателя сочетанием клавиш ALT+SHIFT+X, затем создать указатель в нужном месте командой ленты **Ссылки ► Предметный ука-**

затель или в **Word 2003** командой меню ***Вставка*** ► ***Ссылка*** ► ***Оглавление и указатели***, вкладка окна **Указатель.**

6. Проверить правописание

В параметрах системы **Word** можно задать автоматическую проверку правописания и параметры проверки. Принудительно задать проверку можно в **Word 2007** – на ленте **Рецензирование** кнопкой **Правописание**, в **Word 2003** – в меню **Сервис ► Правописание.**

3.3.3 Выполнение вычислений в системе Microsoft Office Excel

Принципы организации вычислений во всех электронных таблицах примерно одинаковы (начиная с первых, таких систем для **MS DOS SuperCalc** и **Lotus 1-2-3**). В системах для операционной системы **Windows** изменились, в основном, возможности оформления документов, которые достаточны для размещения расчетных таблиц и диаграмм в деловых документах, научных отчетах, статьях.

Основным элементом электронной таблицы является ячейка (или клетка) **таблицы.**

В каждую ячейку может быть помещена информация различного типа:

- **текстовая** длиной до 32767 символов;
- **числовая;**
- **дата;**
- **время** и пр.

Для отображения в ячейках числовой информации, даты и времени могут быть заданы различные форматы (на ленте **Главная** группа **Число** или в **Excel** до 2007 меню ***Формат*** ► ***Ячейки…***, вкладка окна **Число**). Для форматирования ячейки можно также воспользоваться мини-панелью инструментов, которая появляется вместе с контекстным меню при щелчке правой кнопкой мыши на ячейке таблицы.

Ячейка электронной таблицы по умолчанию имеет стиль **Обычный** (лента **Главная**, группа **Стили** или в **Excel** до 2007 меню **Формат ► Стиль…**), который после инсталляции системы имеет параметры, представленные на рисунке 3.43. Параметры стиля **Обычный** можно изменить, тогда все ячейки таблицы этого стиля будут иметь новые свойства. Можно создать новый стиль, если для ячейки задать новые свойства (лента **Главная** или до версии 2007 меню ***Формат*** ► ***Ячейки***...), а затем в окне **Стиль**

задать ***Имя стиля*** и нажать кнопку **ОК** (или **Добавить** в **Excel** до 2007). Все стили, кроме стиля **Обычный**, могут быть удалены.

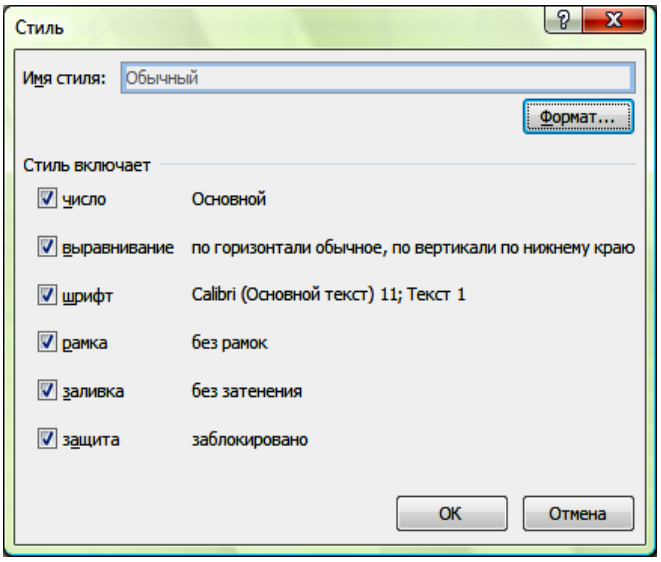


Рисунок 3.43. Окно редактирования стилей ячеек

В версии **Excel 2007** для работы со стилями ячеек используется **Галерея**, выбор в которой одного из вариантов приводит к динамическому изменению вида текущей ячейки (см. **Приложение 4**).

Каждая ячейка электронной таблицы имеет свой адрес, который используется при написании формул (ссылки на ячейки). Наиболее часто используется стиль ссылок с обозначением столбцов латинскими буквами и строк – числами, например, **A1** или **AB32**, однако стиль можно изменить (кнопка **Office ► Параметры Excel**, или в версиях до 2007 меню ***Сервис ► Параметры***, вкладка окна **Общие**) на обозначение столбцов и колонок числами. Ссылки предыдущего примера будут выглядеть, как **R1C1** и **R32C28**, здесь **R** – row (ряд), **C** – column (колонка).

На рисунке 3.44 продемонстрировано несколько принципиально важных моментов работы в электронной таблице:

- написание длинного текста в ячейке (**A1** или **R1C1**), этот текст будет виден справа наложенным на другие ячейки, пока не встретит непустую ячейку;
- использование редактора формул **Microsoft Equation** для оформления документа (**Вставка ► Объект…**);
- отображение текста в ячейке в несколько строк (параметр ячейки «**Переносить по словам**» в окне **Формат ячеек** или кнопка **Перенос текста** группы **Выравнивание** вкладки **Главная** в **Excel 2007**;
- объединение ячеек для написания в них текста с выравниванием по центру (кнопка **Объединить и поместить в центре**);
- использование относительных (**D4** или **RC[-1]**) и абсолютных (**B7** или **R7C2**) ссылок при написании формулы, которую предполагается копировать в другие ячейки таблицы y=f(x) (для копирования ячеек лучше всего использовать маркер в правом нижнем углу ячейки).

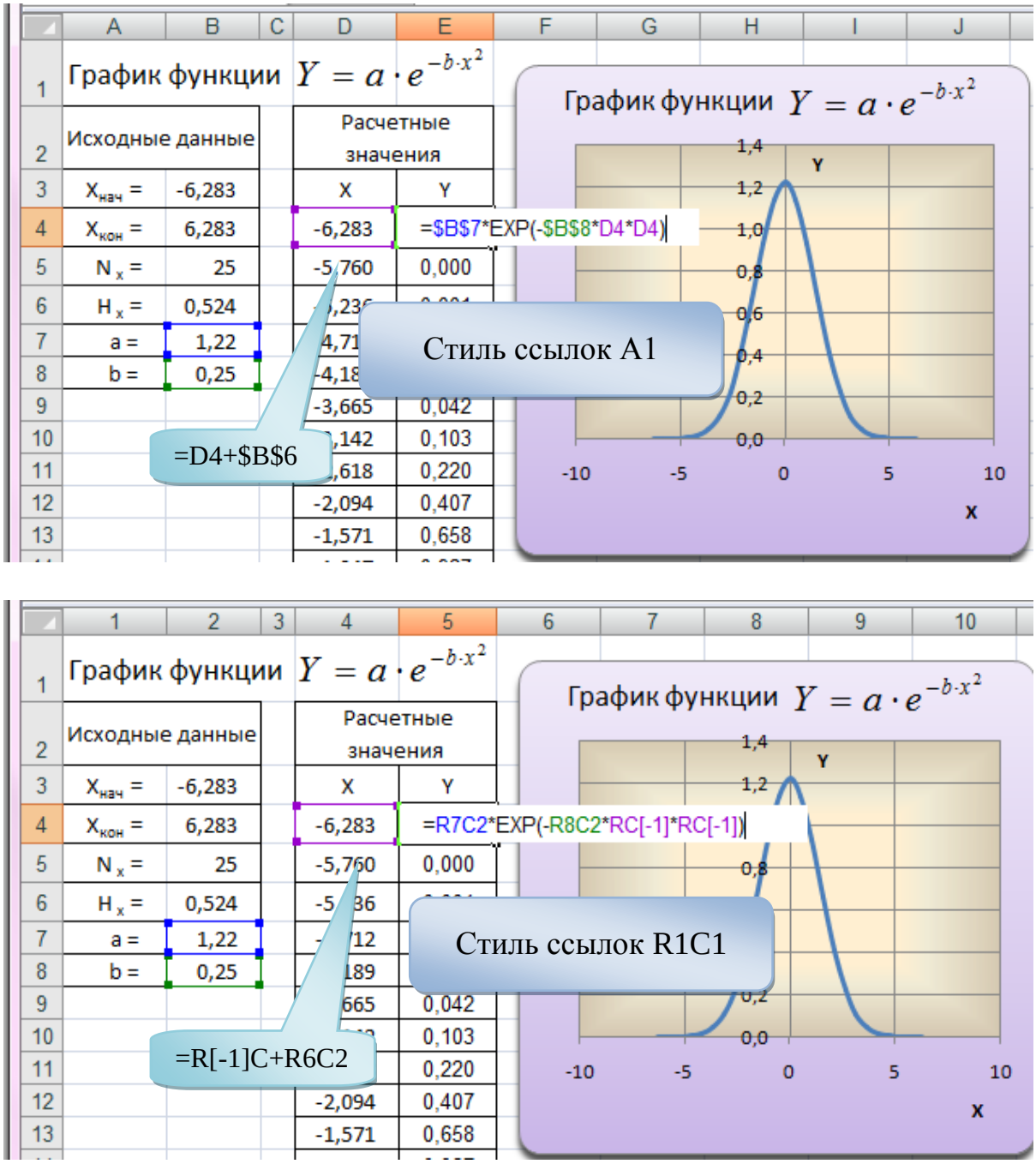


Рисунок 3.44. Два стиля ссылок в системе **Excel**

Никогда не следует писать в формулах ссылки на ячейки посимвольно. Правильно начинать написание формулы со знака равно (=), после этого ссылки заносить в формулу указанием мышкой нужных ячеек на листе таблицы. Более подробную инструкцию по вводу формул можно прочитать в разделе справки **Excel 2003** *«Создание формулы»*.

В одном файле системы **Excel** (в книге) может присутствовать много листов (по умолчанию создается 3 листа), ярлычки которых видны в нижней части книги, минимум – один видимый лист. На ярлычках можно задать имена листов (например, **Система лин.ур.**, **График**). Листы могут быть скрытыми (см. контекстное меню ярлычков страниц). Полная ссылка на ячейку (трехмерная ссылка) состоит из имени листа и через восклицательный знак – ссылки на ячейку: **Лист3!A1** или **График.D3**. Трехмерная ссылка имеет некоторые ограничения по использованию в формулах (см. раздел справки Excel 2003 «*Ссылки на ячейки и диапазоны ячеек*»).

Вместо ссылок на ячейки в формулах можно использовать имена ячеек и их групп. Имя ячейке или выделенной группе ячеек можно присвоить, задав его в поле **Имя**, находящемся в начале строки формул (см. рисунок 3.45). Для работы с **Именами** служит группа **Определенные имена** вкладки **Формулы**. Имя ячейки или их группы можно задать в **Диспетчере имен**, при этом можно определить область действия имен (**Книга** или какой-либо из ее листов), как показано на рисунке 3.45. Заданные имена ячеек можно использовать в формулах. Имя ячейки в формулах играет роль абсолютной ссылки. Имя столбца в формуле эквивалентно смешанной ссылке с неизменным обозначением столбца и относительным номером строки.

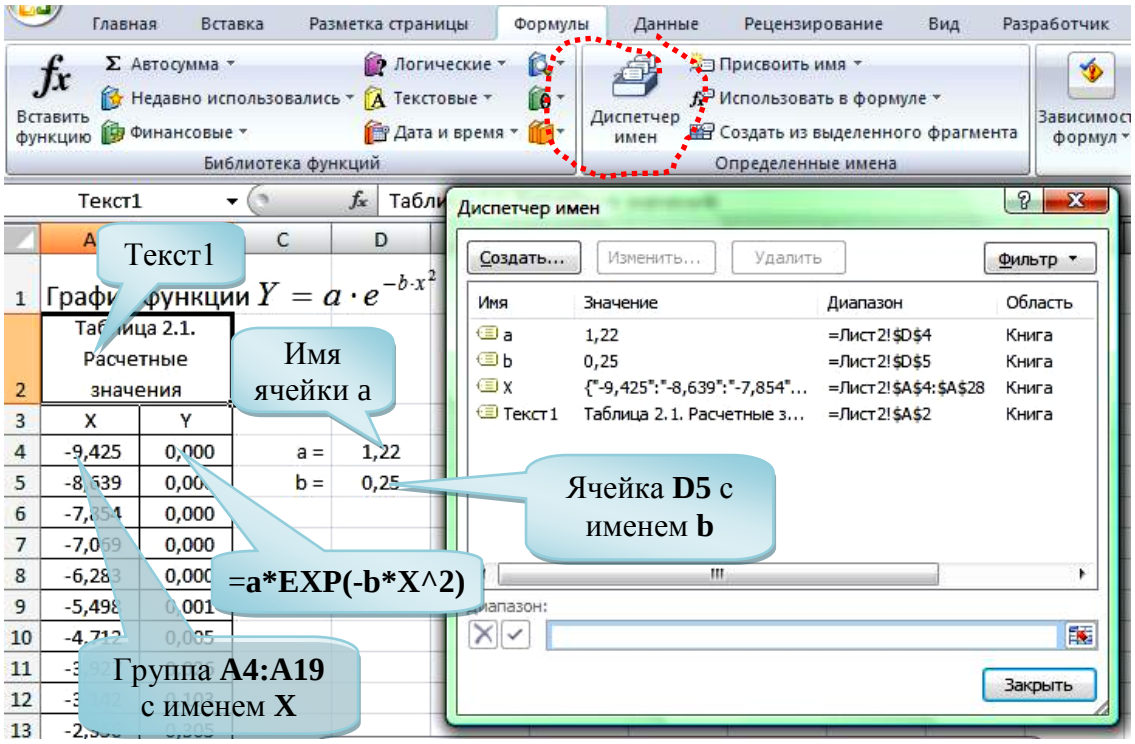


Рисунок 3.45. Использование имен ячеек в системе Excel

Копирование формул – характерный прием в электронных таблицах, который позволяет выполнить одни и те же операции с группой данных (в некоторой степени – аналог циклов при работе с массивами в программах на алгоритмических языках). Именно в операциях копирования большое значение имеет правильное использование абсолютных (не изменяются при копировании) и относительных ссылок (изменяются в соответствии с изменением позиции формулы). Ссылка на ячейку в формуле состоит из двух частей – обозначение колонки и строки, абсолютную адресацию можно использовать для одного из этих компонентов. Пример использования смешанных ссылок – расчет Z для поверхности вида Z = F(X, Y), где формула расчета написана один раз для $Z_1 = F(X_1, Y_1)$, а затем скопирована для всех остальных значений X и Y, т. е. вниз и вправо (см. рисунок 3.46).

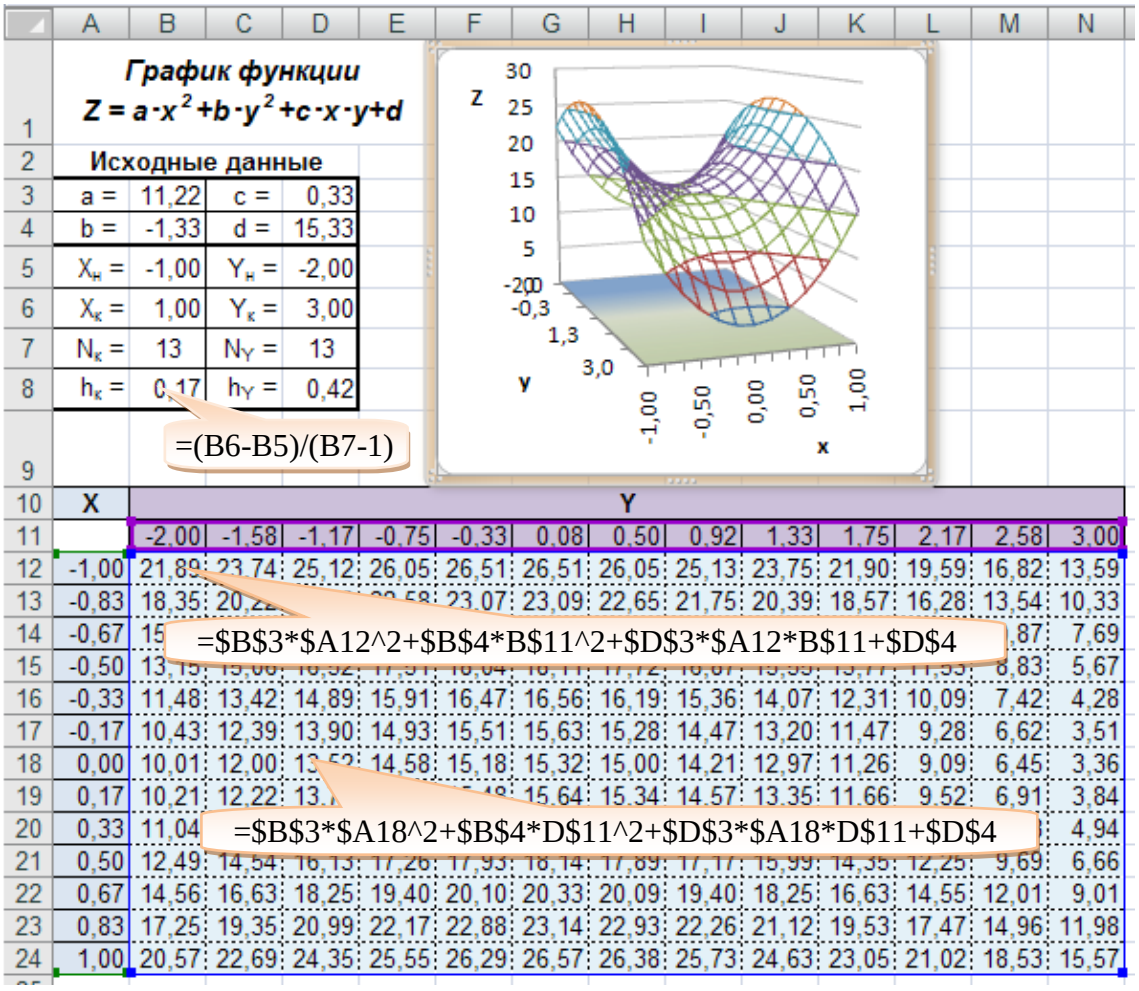


Рисунок 3.46. Использование различных типов ссылок в формулах

Система **Excel** имеет достаточно большой набор функций (см. ленту **Формулы** на рисунке 3.47).

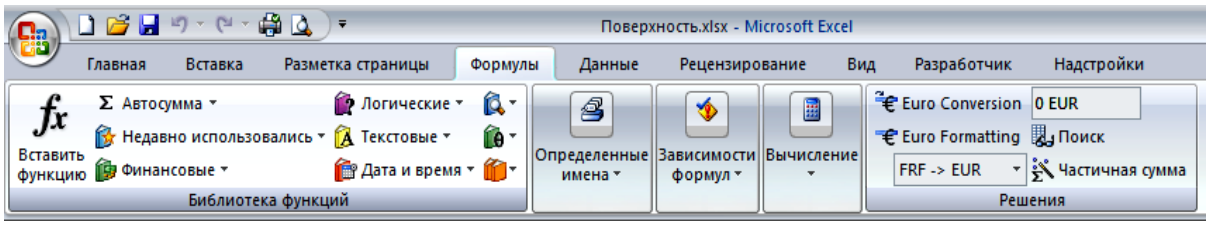


Рисунок 3.47. Вкладка ленты **Формулы** Excel 2007 с установленными надстройками *Поиск решения*, *Мастер суммирования, Пересчет в евро*

Функции сгруппированы по типу обрабатываемых ими данных:

- **текстовые** (ЛЕВСИМВ, ПРАВСИМВ, ПСТР…);
- **математические** (ABS, ACOS, LN, LOG10…);
  - в том числе матричные МОБР, МОПРЕД, МУМНОЖ;
- **финансовые** (БС, ВСД, КПЕР…);
- **статистические** (ДИСП, КВАДРОТКЛ, КОРРЕЛ…);
- **работы с датой и временем** (СЕГОДНЯ, ТДАТА, МИНУТЫ…).

Названия функций приведены для русской версии **Office**, в других версиях эти названия могут быть другие. В связи с этим, всегда следует при использовании функций (даже таких, как **SIN** и **LOG**), найти их в библиотеке системы. Для этого в процессе написания формул можно использовать раскрывающийся список функций слева в строке формул, или значок f_x в той же строке.

Кроме основных функций с системой **Microsoft Office** поставляются надстройки для **Excel** – вспомогательные программы, служащие для использования дополнительных возможностей (таблица 3.12), они устанавливаются из окна **Параметры Excel**, раздел **Надстройки**.

В частности, **пакет анализа** (см. рисунок 3.48) имеет следующие компоненты:

- описательная статистика;
- дисперсионный анализ:: однофакторный; двухфакторный с повторениями; двухфакторный без повторений;
- корреляционный анализ;
- регрессия;
- ковариационный анализ;
- экспоненциальное сглаживание;
- двухвыборочный f-тест для дисперсии;
- анализ Фурье;
- гистограмма;

- скользящее среднее;
- генерация случайных чисел;
- ранг и персентиль;
- выборка;
- t-тест;
- z-тест.

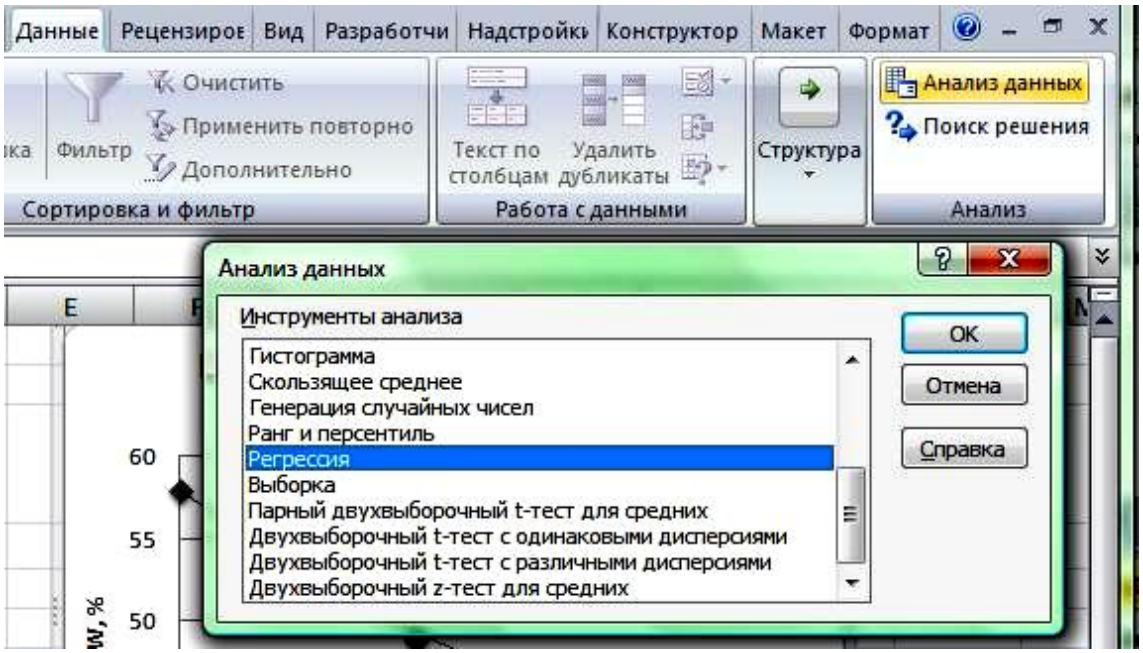


Рисунок 3.48. Excel 2007 с установленной надстройкой *Пакет анализа*

Таблица 3.12. Надстройки **Microsoft Office Excel**

Надстройка	**Описание**
Пакет анализа	Статистические, финансовые и инженерные функции
Пакет анализа VBA	Позволяет разработчикам использовать синтаксис пакета анализа данных
Мастер суммирования	Создает формулу суммирования с условием
Пересчет в евро	функция EUROCONVERT для пересчета
Помощник по Интернету	Позволяет размещать данные **Microsoft Excel** в Интернете, используя синтаксис помощника по Интернету.
Мастер подстановок	Создает формулу для поиска данных в диапазоне по другому значению в этом же диапазоне.
Поиск решения	Вычисляет решения для сценариев «что-если» на основе ячеек перебора и ячеек ограничений.

Описание использования этих компонентов присутствует в справке **Excel 2003**, которую можно вызвать из окна **Анализ данных**. В **Приложении 5** показаны примеры использования пакета анализа при статистической обработке результатов эксперимента – для определения погрешности результата измерения и дисперсионного анализа. Нахождение уравнений регрессии может быть выполнено путем построения линии тренда, проверка адекватности – использованием компонента **Регрессия** пакета анализа.

Методика построения диаграмм в **Excel 2003** отличается от **Word** наличием **Мастера диаграмм**, в **Excel 2007 Мастер** отсутствует, но есть множество **Экспресс-макетов** и **Экспресс-стилей**. Для построения диаграммы рекомендуется предварительно выбрать группу ячеек – источник данных, далее дать команду ***Вставка*** ► ***Диаграмма*** (или для **Excel 2003** нажать кнопку **Мастер диаграмм** на панели инструментов **Стандартная**).

Мастер диаграмм позволяет пошагово задать следующие основные свойства диаграммы:

1) тип диаграммы
   a) **Точечная** – позволяет показать зависимость $Y = F(X)$, даже если точкт по оси X заданы с неравномерным шагом (один из рядов – значения X, другие – Y_1, Y_2…), причем есть возможность показать кривую, проходящую по точкам, с ее сглаживанием или линию тренда (аппроксимация данных экспоненциальной, линейной, логарифмической, полиномиальной или степенной функциями);
   b) **Поверхность** – зависимость $Z = F(X, Y)$ (см. рисунок 3.48);
   c) **Гистограмма**, **Линейчатая**, **График** – диаграммы с равномерным распределением данных по оси X (ось категорий), все ряды – значения по оси Y (ось значений).
2) источник данных диаграммы – если заранее выбран диапазон данных, при построении будут использованы ссылки на эти данные, при этом следует учитывать следующее:
   a) ряды данных могут быть расположены в колонках или столбцах;
   b) при построении поверхности – если клетка в пересечении столбца значений X и строки значений Y пустая (как на рисунке 3.48) в область данных можно включать значения X и Y, которые будут использованы, как обозначения по этим осям;
3) параметры диаграммы – название, подписи по осям, показ линий сетки и пр.
4) размещение диаграммы – на текущем листе или на новом.

Без использования **Мастера** те же свойства можно определить в свойствах объекта **Диаграмма** в **Excel 2007** на контекстной ленте **Работа с**

диаграммами, которая имеет 3 вкладки: **Конструктор**, **Макет** и **Формат**). После создания диаграммы все ее элементы дополнительно можно отредактировать на самой диаграмме или настроить с помощью контекстной ленты **Работа с диаграммами**. На диаграмму можно добавить новые графические объекты ленты **Вставка** или в **Excel 2003** панели **Рисование**.

Векторные фигуры (ленты **Вставка ► Иллюстрации ► Фигуры**) или панели **Рисование** (кроме линий) могут содержать текст или ссылку на ячейку листа **Excel**, что можно использовать, например, в оформлении различных схем, отображающих числа таблицы исходных данных (см. рисунок 3.49).

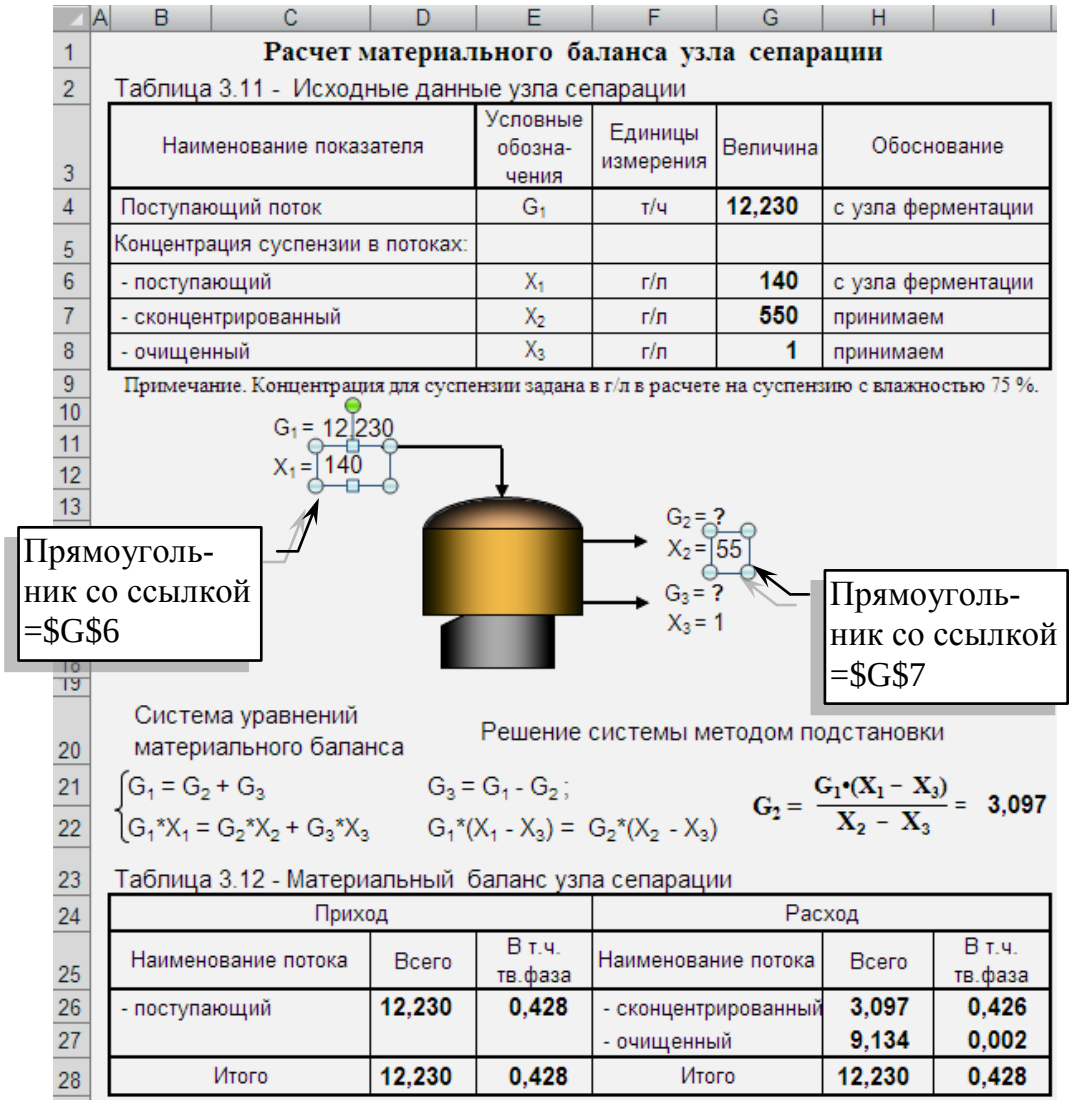


Рисунок 3.49. Использование ссылок в графических объектах

На рисунке 3.50 показан пример использования матричных функций и логической функции *ЕСЛИ(логическое_выражение; значение1; зна-*

чение2), которая возвращает (показывает в ячейке, где она находится) *значение1*, если *логическое_выражение* истинно или *значение2*, если *логическое_выражение* ложно.

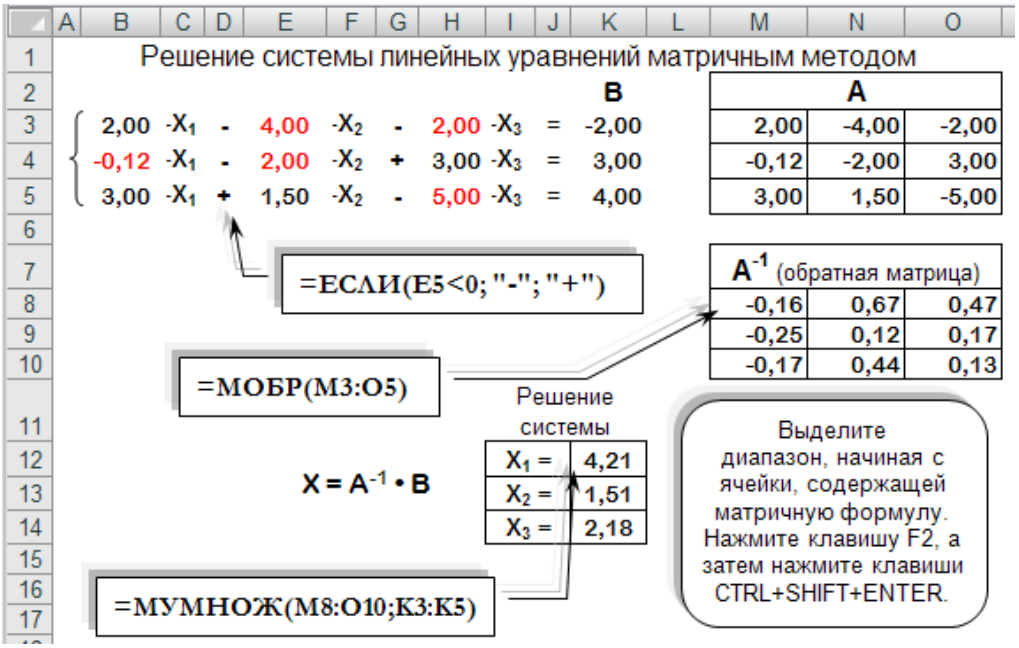


Рисунок 3.50. Использование матричных функций и функции *ЕСЛИ*

В примере функция *ЕСЛИ(E5<0;"-";"+")*, записанная в ячейке *D5* показывает в этой ячейке минус или плюс в зависимости от того, отрицательное или нет число записано в соседней справа клетке *E5* (для числа задан формат без знака).

В этом же примере показано использование двух матричных функций: вычисление обратной матрицы *МОБР(M3:O5)* и умножение матриц *МУМНОЖ(M8:O10;K3:K5)*. Особенность этих функций – результат вычисления имеет вид матриц (в первом случае двухмерная, во втором – вектор). Матричная функция записывается в одну клетку, затем решение разворачивается на группу клеток, как это описано на рисунке 3.50 с использованием сочетания клавиш *CTRL+SHIFT+ENTER*.

Защита листа – заключительная стадия подготовки электронной таблицы к дальнейшему использованию. Достаточно часто подготовленная электронная таблица является шаблоном и служит для расчетов с различными исходными данными. В этом случае доступными для редактирования должны быть только ячейки исходных данных, остальные необходимо защитить от редактирования. По умолчанию стиль **Обычный** имеет параметр **Заблокировано** (см. ранее рисунок 3.43). Поэтому в свойствах ячеек, которые предполагается изменять (окно Формат ячеек, вкладка **Защита**

следует снять пометку **Защищаемая ячейка** или на ленте **Главная** ► группа **Ячейки** ► список **Формат** ► выключить пункт **Блокировать ячейку**). Затем для **Excel 2003** выбрать пункт меню **Сервис** ► **Защита**, для **Excel 2007** в том же списке **Формат**, что и ранее для блокировки ячеек, выбрать команду **Защитить лист…** и в окне **Защита листа** оставить галочку только у пункта **Разрешить… Выделение незаблокированных ячеек**. После этого курсор на листе таблицы сможет перемещаться только по незащищенным ячейкам. При этом можно также задать пароль для снятия защиты, которая выполняется таким же образом, как и ее установка.

Печать листа. Система **Excel** предоставляет большие возможности по настройке печати. Для выбранного формата бумаги можно напечатать лист или заданную область печати в любом масштабе. Настройка выполняется в окне **Параметры страницы** или на вкладке **Разметка страницы Excel 2007** (см. рисунок 3.51).

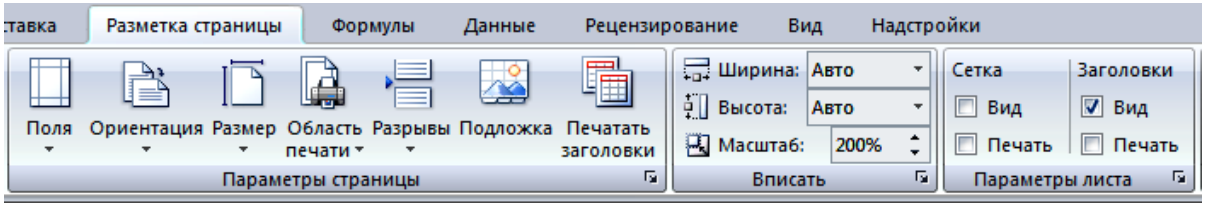


Рисунок 3.51. Настройка параметров страницы и масштаба печати

Например, для документа, занимающего лист формата **А4** при масштабе 100%, можно задать масштаб 200%, при этом документ будет напечатан на четырех листах формата **А4**, все изображение соответственно увеличится в 2 раза. Этот прием может быть использован для печати небольших плакатов с расчетными данными. И наоборот, если документ не помещается в необходимый формат, можно задать, на скольких листах бумаги нужно его разместить при печати, при этом масштаб изображения будет автоматически уменьшен.

3.3.4 Работа с базами данных в системе Microsoft Office Access

База данных – основа любой информационной системы, начиная от систем малых предприятий и кончая ERP-системами управления корпорациями. Понимание принципов организации и функционирования баз данных необходимо всем специалистам, работающим на современных предприятиях.

Система **Microsoft Office Access** имеет достаточно большие возможности по работе с базами данных, развитые средства создания баз данных

и визуального программирования приложений для работы с ними, которые может освоить любой пользователь.

Некоторые термины и определения, которые необходимо знать при работе с базами данных

Используемая терминология различна в теории реляционных баз данных, на стадии проектирования концептуальной модели и при практической работе с базой данных (с физической моделью), как это показано далее.

База данных (БД, database**)** поименованная совокупность структурированные данных, относящихся к определенной предметной области.

Предметная область – некоторая часть реально существующей системы, функционирующая как самостоятельная единица. Полная предметная область может представлять собой экономику страны или группы союзных государств, однако на практике для информационных систем наибольшее значение имеет предметная область масштаба отдельного предприятия или корпорации.

Система управления базами данных (СУБД) – комплекс программных и языковых средств, необходимых для создания и модификации базы данных, добавления, модификации, удаления, поиска и отбора информации, представления информации на экране и в печатном виде, разграничения прав доступа к информации, выполнения других операций с базой.

Реляционная БД – основной тип современных баз данных. Состоит из таблиц, между которыми могут существовать связи по ключевым значениям.

Таблица базы данных (table) – регулярная структура, состоящая из однотипных строк (**записей**, records), разбитых на столбцы (**поля**, fields).

В теории реляционных баз данных синоним таблицы – ***отношение*** *(relation),* строка называется ***кортежем***, столбец отношения называется ***атрибутом***.

В концептуальной модели аналогом таблицы является ***сущность (***entity***),*** с определенным набором свойств – ***атрибутов***, способных принимать определенные значения (набор допустимых значений – ***домен***).

Ключевой элемент таблицы (ключ, regular key) – такое ее поле или строковое выражение, образованное из значений нескольких полей (составной ключ), по которому можно определить значения других полей для одной или нескольких записей таблицы. На практике для использования ключей создаются **индексы** – служебная информация, содержащая упорядоченные сведения о ключевых значениях.

В реляционной теории и концептуальной модели понятие ***«ключ»*** применяется для атрибутов отношения или сущности.

Первичный ключ (primary key) – главный ключевой элемент, однозначно идентифицирующий строку в таблице. Могут также существовать альтернативный (candidate key) и уникальный (unique key) ключи, служащие также для идентификации строк в таблице.

В реляционной теории ***первичный ключ*** – минимальный набор атрибутов, однозначно идентифицирующий кортеж в отношении.

В концептуальной модели ***первичный ключ*** – минимальный набор атрибутов сущности, однозначно идентифицирующий экземпляр сущности.

Связь (relation) – функциональная зависимость между объектами. В реляционных базах данных между таблицами устанавливаются связи по ключам, один из которых (в главной таблице) – первичный, второй может быть тоже первичным для связи «один к одному» или не первичный (внешний, foreign) – связь «один ко многим». Информация о связях сохраняется в базе данных.

Хранимые процедуры (stored procedures**)** – программные модули, сохраняемые в базе данных для выполнения определенных операций с информацией базы.

Ссылочная целостность данных (referential integrity) – набор правил, обеспечивающих соответствие ключевых значений в связанных таблицах.

Триггеры (triggers) – хранимые процедур, обеспечивающие соблюдение ссылочной целостности данных в следующих операциях:

1) добавление записи или изменение внешнего ключа в дочерних (child) таблицах – проверяется наличие соответствующего первичного ключа в родительской (parent) таблице;
2) изменение значения первичного ключа в родительской таблице – возможно каскадное изменение внешних ключей дочерних таблиц,
3) удаление записи в родительской таблице – возможно каскадное удаление всех записей в дочерних таблицах с внешним ключом, соответствующим удаляемому первичному.

Объект (object) – элемент информационной системы, обладающий определенными свойствами (properties) и определенным образом реагирующий на внешние события (events).

Репликация базы данных – создание копий базы данных (реплик), которые могут обмениваться обновляемыми данными или реплицированными формами, отчетами или другими объектами в результате выполнения процесса синхронизации.

Транзакция – изменение информации в базе в результате выполнения одной операция или их последовательности, которое должно быть выполнено полностью или не выполнено вообще. В **СУБД** существуют специальные механизмы обеспечения транзакций.

Язык SQL (Structured Query Language) – универсальный язык работы с базами данных, включающий возможности ее создания, модификации структуры, отбора данных по запросам, модификации информации в базе и прочие операции манипулирования базой данных.

3.3.4.1 Общие принципы разработки информационной системы

Информационная система (**ИС**) – совокупность средств, способов и методов, направленных на создание и использование систем сбора, передачи, обработки и надежного хранения информации.

База данных – важнейший компонент ИС. Хорошо спроектированная база данных позволяет не только безпроблемно эксплуатировать систему и выполнять ее текущее обслуживание, но и модифицировать и развивать ее при развитии предприятия и изменении информационных потоков, законодательства и форм отчетности.

В настоящее время в эксплуатации на крупных предприятиях находятся комплексные ИС управления предприятиями (КИС, корпоративные системы, ERP-системы), такие, как R/3 фирмы SAP, Oracle E-Business Suite, InforERP. Среди российских разработок приближаются к системам класса ERP **Галактика**, **Флагман**, **Парус**.

Любая сложная система для обеспечения ее надежного функционирования строится, как иерархическая система, состоящая из отдельных подсистем и модулей, взаимодействующих между собой и использующих общую базу данных.

Разработка информационной системы, ведется в несколько этапов, на каждом из которых конкретизируются и уточняются элементы разрабатываемой системы.

Существует различные типы схем, иллюстрирующих жизненный цикл разработки ИС. На рисунке 3.52 показана каскадная схема с обратной связью.

При описании информационной системы предполагается, что она содержит два типа сущностей: операционные сущности, которые выполняют какую-либо обработку (некоторый аналог программы) и пассивные сущности, которые хранят информацию, доступную для пополнения, изменения, поиска, чтения (база данных).

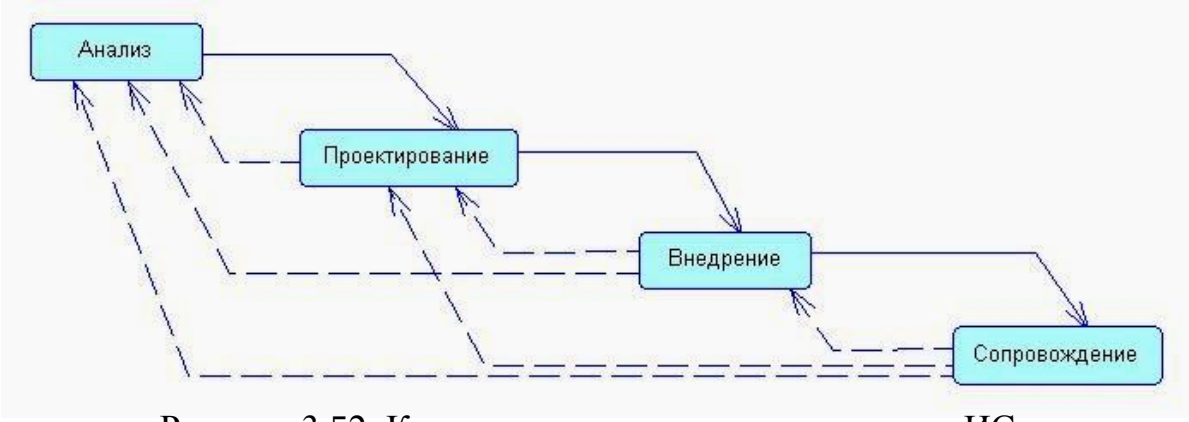


Рисунок 3.52. Каскадная схема жизненного цикла ИС

При проектировании сложных информационных систем используется метод декомпозиции – система разбивается на составные части, которые связаны и взаимодействуют друг с другом и образуют иерархическую структуру. Иерархический характер сложных систем хорошо согласуется с принципом групповой разработки. В этом случае деятельность каждого участника проекта ограничивается соответствующим иерархическим уровнем.

Классический подход к разработке сложных систем представляет собой структурное проектирование, при котором осуществляется алгоритмическая декомпозиция системы по методу **«сверху-вниз»**. Именно в этом случае можно построить хорошо функционирующую систему с общей базой данных, согласованными форматами использования и обработки информации на всех участках, с оптимальным взаимодействием всех подсистем.

Исторически сложилось так, что некоторые системы разрабатывались по методу **«снизу - вверх»** – вначале разрабатывались отдельные автоматизированные рабочие места (АРМы), затем предпринимались попытки объединения их в единую информационную систему. Подобные разработки для крупных систем не могут быть успешны.

При разработке проекта информационной системы следует определить:

1. объекты информационной системы (сущности в концептуальной модели);
2. их свойства (атрибуты);
3. взаимодействие объектов (связи) и информационные потоки внутри и между ними.

При этом очень важен анализ существующих бизнес-процессов, а также учет при проектировании нормативной базы (законов, постановлений правительства, отраслевых стандартов), определяющей объем и фор-

мат хранения и передачи информации. Если радикальной перестройки сложившегося информационного процесса не предвидится, следует учитывать имеющиеся формы хранения и обработки информации в виде журналов, ведомостей, таблиц и т.п. бумажных носителей.

Однако, предварительного необходимо выполнить анализ возможности перехода на новые системы учета, хранения и обработки информации, возможно, исходя из имеющихся на рынке программных продуктов - аналогов, разработанных крупными информационными компаниями и частично или полностью соответствующими поставленной задаче. Схема формирования информационной модели представлена на рисунке 3.53.

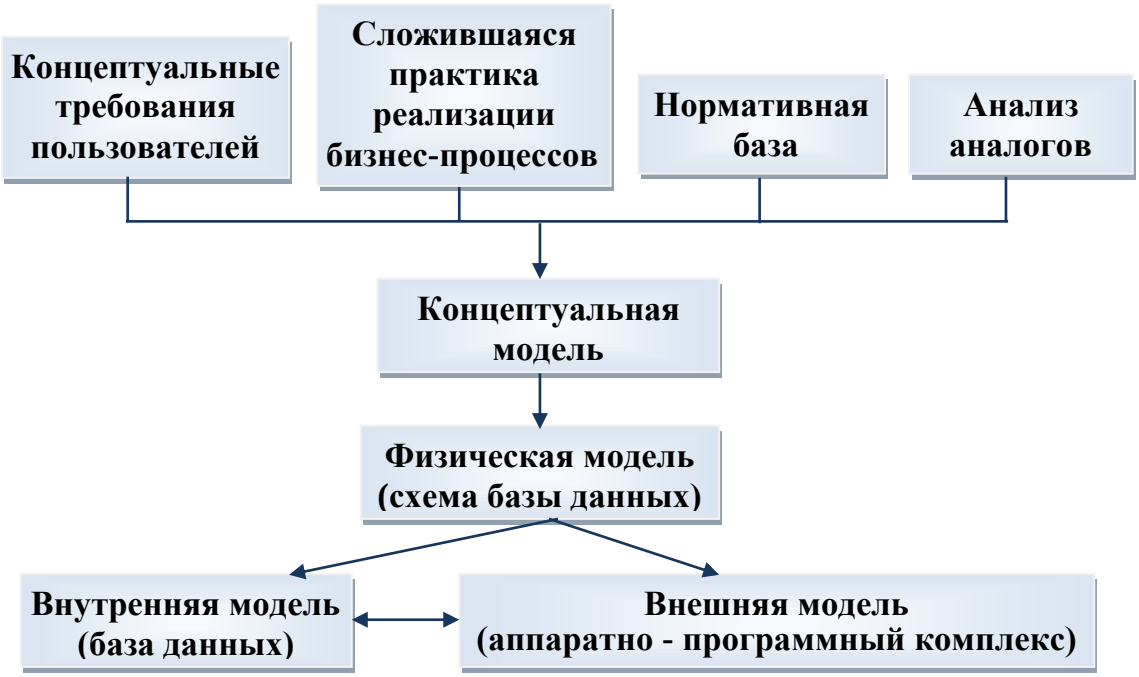


Рисунок 3.53. Схема формирования информационной модели

Концептуальная модель (см. рисунок 3.53) – отображает информационные объекты, их свойства и связи между ними без указания способов физического хранения информации (модель предметной области, иногда ее также называют информационно-логической или инфологической моделью). Информационными объектами обычно являются ***сущности*** – обособленные объекты или события, о которых необходимо сохранять информацию, имеющие определенные наборы свойств – ***атрибутов***.

Физическая модель – отражает все свойства (атрибуты) информационных объектов базы и связи между ними с учетом способа их хранения – используемой СУБД.

Внутренняя модель – база данных, соответствующая определенной физической модели.

Внешняя модель – комплекс программных и аппаратных средств для работы с базой данных, обеспечивающий процессы создания, хранения, редактирования, удаления и поиска информации, а также решающий задачи выполнения необходимых расчетов и создания выходных печатных форм.

На рисунке 3.54 приведена схема, показывающая взаимосвязь основных терминов, связанных с вопросами проектирования баз данных и работы с ними.

При разработке сложных информационных систем (при коллективе разработчиков в десятки и сотни человек) используются следующие технологии:

1. Визуальное моделирование с использованием **CASE**-средств (CASE – Computer Aided Software Engineering) *Case Studio, SyBase Power Disigner, ERWin Data Modeler* и др. Данные системы позволяют на основе визуально описанной модели базы данных генерировать программный код на языках SQL, VBScript, JScript, либо другие технологии для переноса модели в реальные СУБД, которыми могут быть Oracle, Microsoft SQL Server, IBM DB2, Informix, Microsoft Access и др.
2. Методы структурного анализа и проектирования **SADT** (Structured Analysis and Design Technique).
3. Методы функционального моделирования семейства **IDEF (**Icam DEFinition, ICAM – Integrated Computer-Aided Manufacturing – проект, который впервые был реализован для ВВС США в 1973 г.**)**. Их описание можно найти, например, в интернет-библиотеке Верникова (*http://www.vpg.ru/main.mhtml?PubID=6*).
4. Методы объектно-ориентированного анализа и проектирования с использованием унифицированного языка моделирования **UML** (Unified Modeling Language).
5. Методы языка моделирования бизнес-процессов **BPML** (Business Process Modeling Language).
6. Методы языка моделирования предприятий **UEML** (Unified Enterprise Modeling Language), соответствующего стандартам ISO 14258 Rules and Guidelines for Enterprise Models (Правила и руководящие принципы для моделей предприятия) и ISO 15704 Requirements for enterprise-reference architectures and methodologies (Требования и методологии по описанию архитектуры предприятия).

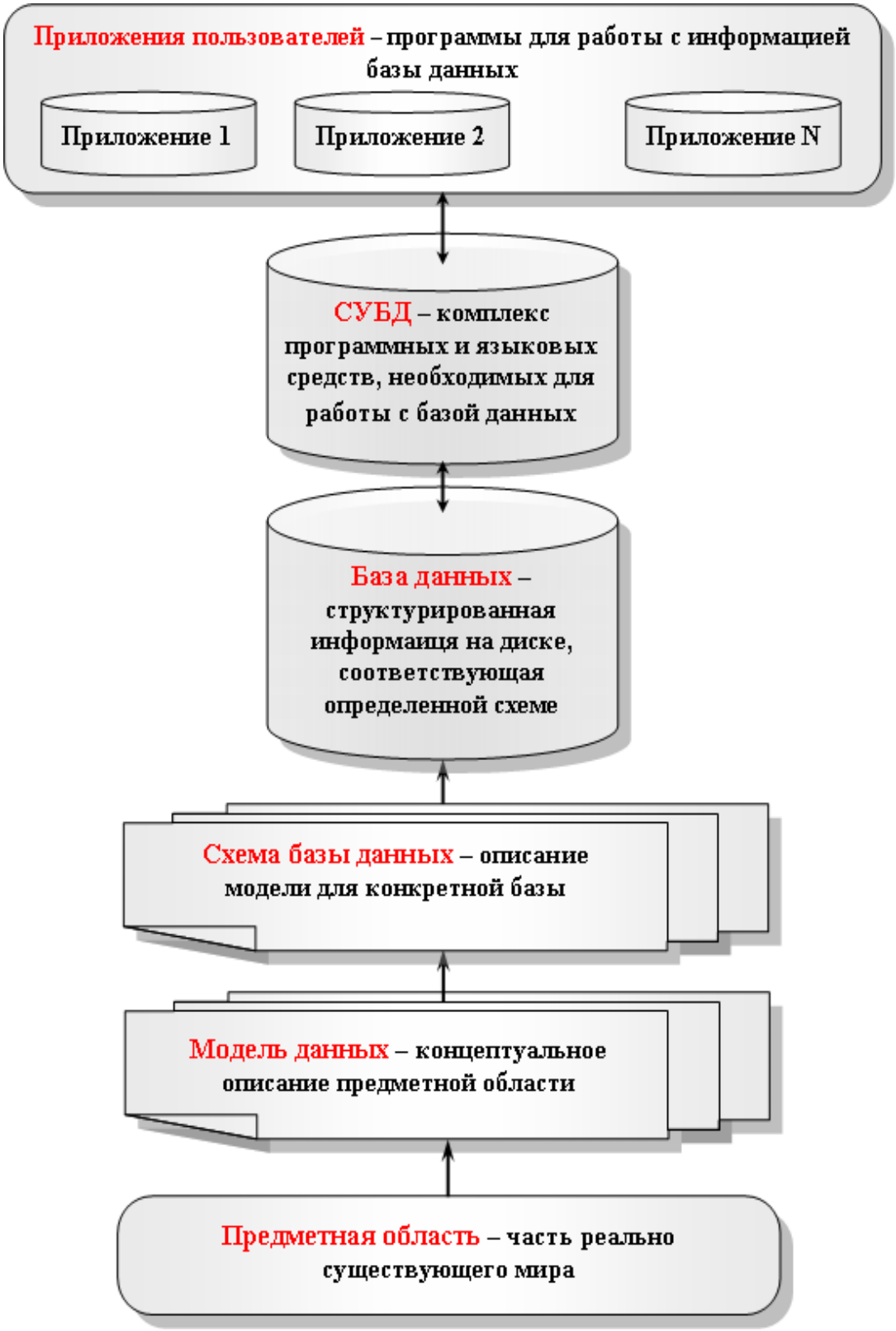


Рисунок 3.54. Взаимосвязь основных терминов

Выбор системы для разработки пользовательских приложений для работы с базой данных – сложное и ответственное решение. Возможность разработки таких программ имеют универсальные системы программирования, такие, как Microsoft Visual Studio (языки программирования C#, C++, Basic) и др. В связи с повсеместным распространением Интранета, Экстранета и Интернета многие системы имеют возможность разработки трехуровневой сервис-ориентированной архитектуры Web-приложений для работы с базами данных.

В качестве аппаратных средств наиболее часто используются персональные компьютеры с операционной системой **Microsoft Windows**, локальная сеть строится с использованием возможностей этой ОС, файловый сервер и сервер баз данных может использовать **Microsoft Windows** Server 2003, либо другую операционную систему для выделенных серверов (например, Unix или NetWare).

Разработка концептуальной модели базы данных на примере информационной системы «Контингент студентов университета»

Первоначальный этап – создание текстового описания моделируемой системы и ее подсистем.

Постановка задачи. Главная задача системы – сохранение в базе данных всех необходимых сведений о студентах и их успеваемости, формирование необходимых печатных форм для проведения зачетной и экзаменационной работы преподавателей, генерация сводных итогов по результатам сессии для руководящих работников деканатов, институтов и университета. При разработке системы следует учитывать, что она взаимодействует с системами «Абитуриент», «Стипендия» и «Кадры университета». Информация о студентах первоначально поступает из системы «Абитуриент» и редактируется на уровне деканатов. Она должна также удовлетворять требованиям бухгалтерского учета по начислению стипендий. Система должна использовать справочник специальностей, утвержденный в вышестоящих органах. Информация об успеваемости студентов накапливается постоянно и сохраняется за весь период обучения, после чего переносится в архивное хранилище данных. В системе должен использоваться единый справочник дисциплин (предметов) для всех подразделений университета.

Описание концептуальной модели. На концептуальном уровне данные информационной системы состоят их двух основных сущностей: **Студент** и **Успеваемость**. Сущность **Студент** связана с сущностями **Фа-**

культет и **Специальность**. Сущность **Успеваемость** связана с сущностью **Предмет.**

Минимальный состав атрибутов, их описание и особенности для сущности **Студент** представлены в таблице 3.13.

Таблица 3.13. Атрибуты сущности **Студент**

Имя атрибута	Описание, особенности использования
Номер зачетки	Уникальный номер, однозначно идентифицирующий студента университета, является первичным ключом
Фамилия, имя, отчество	Является простым атрибутом с точки зрения экземпляра сущности, при необходимости из общего поля можно выделить составляющие его фамилию, имя и отчество или фамилию и инициалы; на практике часто этот атрибут разделяют на 3 отдельных; первый вариант является более экономичным по необходимой общей ширине поля таблицы
Дата поступления в университет	В нашей стране наиболее часто используется формат работы с датой в виде дд.мм.гг, что совпадает с немецким (German) форматом дат. Количество цифр года – может быть два - для новых систем, поддерживающих заданный в **Microsoft Windows** годичный интервал (Панель управления – Язык и стандарты – Дата – «При вводе двух цифр года воспринимать их как год между…») или для систем, в которых аналогичный интервал может быть задан в программе, либо 4 цифры
Факультет (№ факультета)	Может быть сложным атрибутом (кроме названия может содержать и другие сведения); должен быть представлен в одинаковом виде для каждого факультета, поэтому в соответствии с принципами нормализации баз данных этот атрибут следует представить в виде номера, являющегося внешним ключом для новой сущности – **Факультет**, в которой каждому номеру, являющемуся первичным ключом, будут соответствовать название и прочие атрибуты этой сущности
Специальность (Код специальности)	Может быть сложным атрибутом, кроме того, необходимо использовать справочник министерства с утвержденными кодами специальностей, поэтому данный атрибут должен являться кодом специальности – внешним ключом для первичного ключа новой сущности **Специальность**
Курс	Число от 1 до 5
Номер группы	Трехзначное число
Номер паспорта	Состав и вид паспортных данных определяется требованиями отчетности перед налоговыми и др. органами
…	Прочие атрибуты, которых может быть достаточно много…

В табл. 3.14 – 3.17 представлены атрибуты сущностей **Успеваемость, Факультет, Специальность** и **Предмет.**

Таблица 3.14. Атрибуты сущности **Успеваемость**

Имя атрибута	Описание, особенности использования
Номер зачетки	Внешний ключ (к сущности **Студент**)
Номер семестра	Число от 1 до 10
Предмет (№ предмета)	Может быть сложным, его следует заменить на его номер (внешний ключ) и связать с новой сущностью **Предмет**, состоящий, как минимум из атрибутов **Номер предмета** (первичный ключ) и **Название предмета**
Оценка	Может быть представлена цифрами от 0 до 5 или 1 буквой – например «н» - неявка
Дата получения оценки	Формат даты обычно дд.мм.гг
Фамилия преподавателя	Данное поле может быть связано с сущностью **Преподаватель**. В данном учебном примере ограничимся простым атрибутом.
…	Могут быть добавлены и другие атрибуты, например, номер экзаменационной ведомости

Таблица 3.15. Атрибуты сущности **Факультет**

Имя атрибута	Описание, особенности использования
Номер факультета	Первичный ключ
Название факультета	Может быть достаточно длинным, но не более 255 символов
…	Могут быть добавлены и другие атрибуты, например, декан, номер комнаты деканата и т. д.

Таблица 3.16. Атрибуты сущности **Специальность**

Имя атрибута	Описание, особенности использования
Код специальности	Первичный ключ – значение из справочника министерства
Название специальности	Значение из справочника министерства
…	Могут быть добавлены и другие атрибуты

Таблица 3.17. Атрибуты сущности **Предмет**

Имя атрибута	Описание, особенности использования
№ предмета	Первичный ключ
Название предмета	Общий справочник университета
…	Могут быть добавлены и другие атрибуты

В физической модели каждой сущности будет соответствовать таблица базы данных, Каждому атрибуту будет соответствовать поле таблицы. Имена таблиц и полей лучше задавать с использование латинских букв и достаточно короткими для удобства использования при программировании и для совместимости с системами, не использующими кириллицу. Состав данных и связи в концептуальной и физической моделях показаны в табл. 3.18 и 3.19.

Таблица 3.18. Состав базы данных информационной системы

№ п/п	Сущности концептуальной модели	Таблицы физической модели	
		Название	Информация
1	Студент	Список	Список студентов
2	Успеваемость	Оценки	Оценки студентов
3	Факультет	Факультеты	Справочник факультетов
4	Специальность	Специальности	Справочник специальностей
5	Предмет	Предметы	Справочник предметов

Таблица 3.19. Связи между объектами базы данных

№ п/п	Концептуальная модель	Физическая модель
1	Студент – Успеваемость	Список – Оценки
2	Студент – Факультет	Список – Факультеты
3	Студент – Специальность	Список – Специальности
4	Успеваемость – Предмет	Оценки – Предметы

3.3.4.2 Общая характеристика системы Microsoft Office Access

Система **Microsoft Access** является одним из компонентов **Microsoft Office** и предназначена для работы с базами данных.

Особенность данной СУБД – вся информация базы данных хранится в одном файле (*.mdb или *.accdb для версии **Access** 2007). Кроме информации таблиц, в этом же файле сохраняются компоненты приложения для

работы с базой данных – экранные формы, отчеты, запросы, программные модули.

Для написания программных модулей обработки информации базы данных используется система **Microsoft Visual Basic for Applications** (**VBA**), являющаяся компонентом **Microsoft Office**.

Для работы с базой данных система использует **Microsoft** Jet database engine – систему управления базами данных, извлекающую и сохраняющую данные в пользовательских и системных задачах. Ядро базы данных **Microsoft Jet** можно рассматривать как компонент диспетчера данных, с помощью которого работают остальные компоненты системы доступа к данным, такие как **Microsoft Access** и **Microsoft Visual Basic for Applications**.

Основные возможности системы при работе с базами данных приведены в **Приложении 6**. Самые главные параметры системы, о которых не следует забывать:

- размер файла базы данных – не более 2 Гбайт;
- размер таблицы – не более 2 Гбайт;
- число одновременно работающих пользователей – не более 255;
- число знаков в текстовом поле – не более 255;
- число индексов в таблице – не более 32;
- число полей в индексе – не более 10;
- ширина формы или отчета – не более 55,87 см;
- высота всех разделов отчета в режиме конструктора – не более 508 см.

В табл. 3.20 приведены сведения о типах данных, которые могут иметь поля в таблицах.

Таблица 3.20. Типы данных системы **Microsoft Office Access 2007**

Тип данных полей таблицы	Тип данных в VBA	Использование	Размер
Текстовый	String	Текст, состоящий из любых символов в кодировке Unicode (2 байта на символ)	До 255 символов
Поле МЕМО	String	Текст в кодировке Unicode	До 64 000 символов
Логический	Boolean	Поля, содержащие только одно из двух возможных значений, таких как Да/Нет, Истина/Ложь, Вкл/Выкл.	1 бит

Продолжение таблицы 3.20

Тип данных полей таблицы	Тип данных в VBA	Использование	Размер
Числовой (Байт, Целое, Длинное целое, Одинарное с плавающей точкой, Двойное с плавающей точкой, Действительное, Код репликации)	Byte, Integer, Long, Single, Double, Decimal	Числовые данные	1, 2, 4 или 8 байтов. 16 байтов для кодов репликации (GUID)
Дата/время	Date	Дата и время	8 байтов (при активации поля всегда показывает полный формат даты)
Полный формат даты		31.12.04 23:55:59	
Длинный формат даты		31 декабря 2004 г.	
Средний формат даты		31-дек-04	
Краткий формат даты		31.12.04	
Длинный формат времени		23:55:59	
Средний формат времени		11:55	
Краткий формат времени		23:55	
Денежный	Currency	Значения валют. Денежный тип используется для предотвращения округлений во время вычислений. Предполагает до 15 символов в целой части числа и 4 - в дробной.	8 байтов
Счетчик		Автоматическая вставка последовательных (увеличивающихся на 1) или случайных чисел при добавлении записи.	4 байта. 16 байтов для кодов репликации (GUID)
Логический	Boolean	Поля, содержащие только одно из двух возможных значений, таких как Да/Нет, Истина/Ложь, Вкл/Выкл.	1 бит

Продолжение таблицы 3.20

Тип данных полей таблицы	Тип данных в VBA	Использование	Размер
Поле объекта OLE	String	Объекты (например, документы **Microsoft** Word, электронные таблицы **Microsoft** Excel, рисунки, звуки и другие двоичные данные), созданные в программах, использующих протокол OLE. Объекты могут быть связанными или.	До 1 гигабайта (ограничено объемом диска)
Гиперссылка	String	Поле, в котором хранятся гиперссылки. Гиперссылка может иметь вид пути UNC, либо URL-адреса.	До 64 000 символов
Вложение		Несколько любых файлов, сохраненных в одном поле.	
Мастер подстановок		Создает поле, позволяющее выбрать значение из другой таблицы или из списка значений, используя поле со списком. При выборе данного типа запускается Мастер для определения этого поля.	Тот же размер, который имеет первичный ключ - поле подстановок

Система **Microsoft Access** имеет собственные средства для разграничения прав доступа пользователей к базе данных.

Простейшим способом ограничения доступа к базе данных является **установка пароля для открытия базы данных**. После установки пароля при каждом открытии базы данных будет появляться диалоговое окно, в которое требуется ввести пароль. Этот способ достаточно надежен (**Microsoft Access** шифрует пароль, поэтому к нему нет доступа при непосредственном чтении файла базы данных), но он действует только при открытии базы данных. После открытия базы данных все объекты становятся доступными для пользователя (пока не определены другие типы защиты, описанные ниже в этом разделе). Для базы данных, которая совместно используется небольшой группой пользователей или на автономном компьютере, обычно оказывается достаточно установки пароля.

База данных может быть зашифрована. При шифровании базы данных ее файл сжимается и становится недоступным для чтения с помо-

щью служебных программ или текстовых редакторов. Дешифрование базы данных отменяет результаты операции шифрования.

Нельзя использовать установку пароля на базу данных, если предполагается выполнять репликацию базы данных. Реплицированные базы данных не могут быть синхронизированы, если установлен пароль базы данных.

Защита на уровне пользователей имеет большие возможности по разграничению прав. Этот способ подобен способам, используемым в большинстве сетевых систем.

При запуске **Microsoft Access** от пользователя требуется идентифицировать себя и ввести пароль. **Microsoft Access** по умолчанию создает две группы: администраторы (группа **Admins**) и простые пользователи (группа **Users**). Допускается определение других групп и пользователей.

Члены группы **Admins** имеют разрешения на доступ ко всем объектам базы данных. Другим группам и пользователям могут предоставляться разрешения на доступ только к отдельным объектам базы данных. Типовые разрешения на доступ для группы **Users** могут включать **Чтение данных** и **Обновление данных** для таблиц и запросов и **Открытие/запуск** для форм и отчетов.

3.3.4.3 Создание базы данных в Microsoft Office Access 2007

Процесс создания базы данных рассмотрим на примере разработки информационной системы «Контингент студентов университета», концептуальная модель которой была описана ранее[†].

Для создании базы данных в меню кнопки **Office** выберем команду **Создать**, после чего откроется окно **Приступая к работе…** с разделом **Новая база данных** (см. рисунок 3.55), а также списками локальных шаблонов баз данных и шаблон из Интернета.

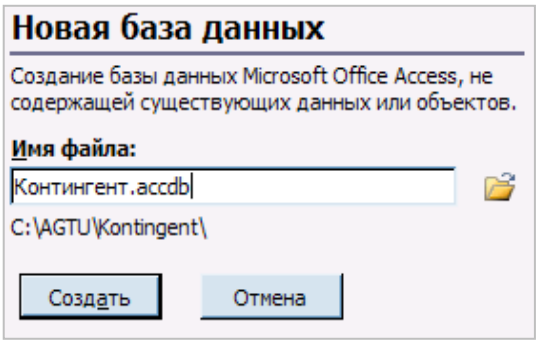


Рисунок 3.55. Окно создания новой базы данных

Для новой базы данных зададим Имя файла – **Контингент** и папку, где он будет размещен, например, C:\AGTU\Kontingent, нажмем на кнопку Создать (рисунок 3.55), после чего откроется окно базы данных в режиме работы с таблицами.

[†] Описание для версии Access 2003 см. А. С. Грошев. Базы данных. – Архангельск, 2005. – 124 с.

На ленте для группы вкладок **Работа с таблицами** выберем режим **Конструктор** (группа **Представления ► Вид**), зададим в появившемся окне **Сохранение** имя таблицы **Список** и опишем ее структуру в соответствии с данными таблицы 3.14, т.е. зададим имя, тип, размер или формат каждого поля таблицы, а также первичный ключ (если необходимо), индексные выражения и подписи (см. рисунок 3.56).

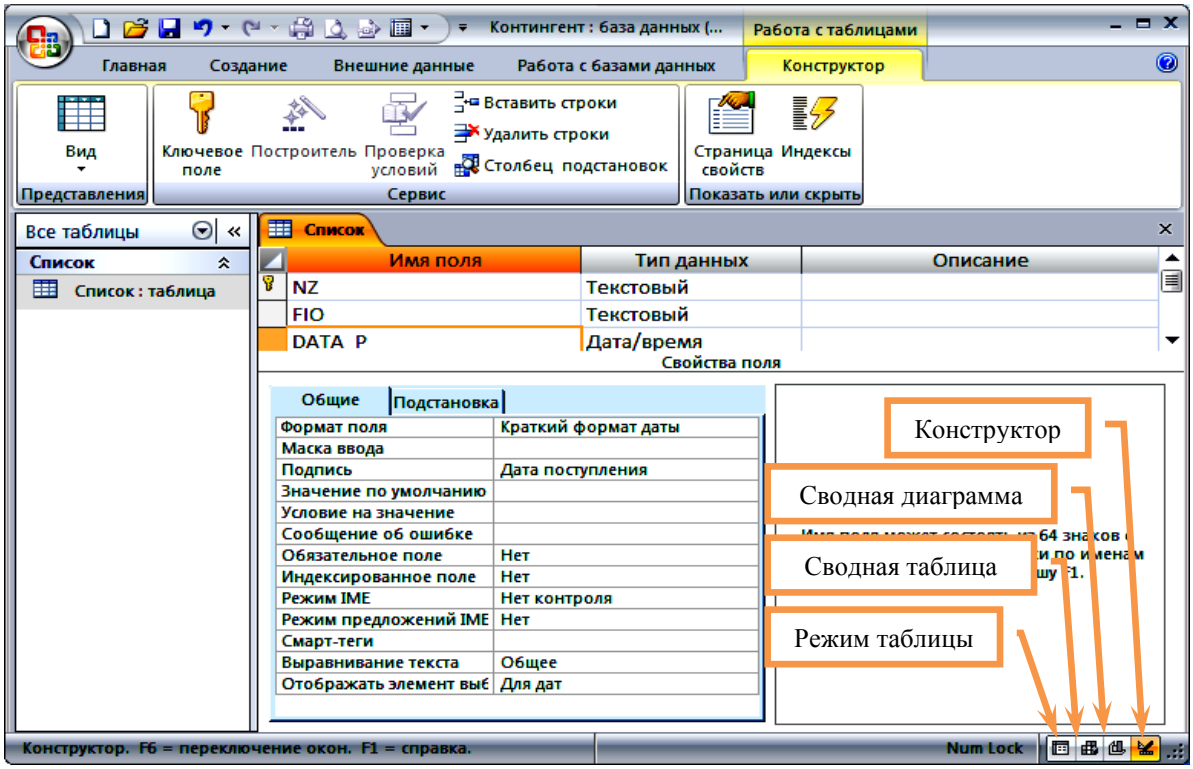


Рисунок 3.56. Работа с таблицей в режиме **Конструктор**

Имена полей лучше писать латинскими буквами, в одно короткое слово (можно использовать символ подчеркивания) – для удобства работы с ними в запросах и программах, работающих с базой данных; задание подписей для полей облегчает разработку экранных форм и отчетов.

В системе **Access** под термином **Ключевое поле** подразумевается первичный ключ, который однозначно идентифицирует запись в таблице. Для **Ключевого поля** система контролирует отсутствие дублирования его значений при вводе данных. Внешний ключ в этой системе обозначается, как **Индексированное поле, допускаются совпадения.**

В Параметрах **Access** (см. панель кнопки **Office**) для текущей базы задан показ окон в виде вкладок (в окне **Access**), как показано на рисунке 3.56. Возможна настройка работы в старом стиле – «перекрытие окон», при этом внутри главного окна системы **Access** открываются дочерние окна таблиц, форм, отчетов и т. д. Для удобства показа иллюстраций далее ис-

пользовался оконный режим. На практике можно использовать режим вкладок, где видны ярлычки всех открытых окон.

Структура таблицы **Список** приведена в табл. 3.21.

Таблица 3.21. Структура таблицы **Список**

Имя поля	Тип данных	Размер поля	Индексированное поле	Подпись
NZ	Текстовый	8	Да**, Ключевое поле**	№ зачетки
FIO	Текстовый	45	Нет	Фамилия, имя, отчество
DATA_P	Дата/время	Краткий формат даты	Нет	Дата поступления
N_FCLT	Числовой	Байт	Да (Допускаются совпадения)	Факультет
N_SPECT	Текстовый	9	Да (Допускаются совпадения)	Специальность
KURS	Числовой	Байт	Нет	Курс
N_GRUP	Текстовый	10	Нет	Группа
N_PASP	Текстовый	10	Нет	Номер паспорта

Для поля **NZ** следует задать свойство – **Ключевое поле**, т.к. номер зачетки – уникальный для каждого студента и однозначно его идентифицирует в таблице базы. По окончании описания структуры таблицы даем команду **Сохранить** (на панели быстрого доступа, в меню кнопки **Office** или при закрытии вкладки таблицы).

Далее на вкладке ленты **Создание** выбираем кнопку **Конструктор таблиц** и описываем структуру новой таблицы **Факультеты** в соответствии с концептуальной моделью, приведенной ранее в таблице 3.15. Аналогично создаются таблицы **Специальности, Оценки и Предметы.** Структура созданных таблиц базы данных показана в таблицах 3.22 – 3.25.

Таблица 3.22. Структура таблицы **Факультеты**

Имя поля	Тип данных	Размер поля	Индексированное поле	Подпись
N_FCLT	Числовой	Байт	Да**, Ключевое поле**	Номер факультета
NAME_F	Текстовый	120	Нет	Название факультета

Таблица 3.23. Структура таблицы **Специальности**

Имя поля	Тип данных	Размер поля	Индексированное поле	Подпись
N_SPECT	Текстовый	9	Да, **Ключевое поле**	Код специальности
NAME_S	Текстовый	120	Нет	Название специальности
N_SPEC2	Текстовый	9	Нет	Старый код

Таблица 3.24. Структура таблицы **Оценки**

Имя поля	Тип данных	Размер поля	Индексированное поле	Подпись
NZ	Текстовый	8	Да (**Допускаются совпадения**)	№ зачетки
SEMESTR	Числовой	Байт	Нет	Семестр
N_PREDM	Числовой	Целое	Да (Допускаются совпадения)	Предмет
BALL	Текстовый	1	Нет	Оценка
DATA_B	Дата/время	Краткий формат даты	Нет	Дата
PREPOD	Текстовый	45	Нет	Преподаватель

Таблица 3.25. Структура таблицы **Предметы**

Имя поля	Тип данных	Размер поля	Индексированное поле	Подпись
N_PREDM	Числовой	Целое	Да, **Ключевое поле**	Номер предмета
NAME_P	Текстовый	120	Нет	Название предмета

Далее задаем связи **Один ко многим** между таблицами в базе. Для этого на вкладке ленты **Работа с базами данных** выбрать в группе **Показать или скрыть** команду **Схема данных**, добавить в схему все таблицы и, перетаскивая название поля первичного ключа к аналогичному полю другой таблицы создать связи. При этом задаем в окне **Изменение связей** для всех связей между таблицами условия: обеспечения ссылочной целостности данных, каскадное обновление связанных полей и каскадное удаление связанных записей. Схема базы данных показана на рисунок 3.57.

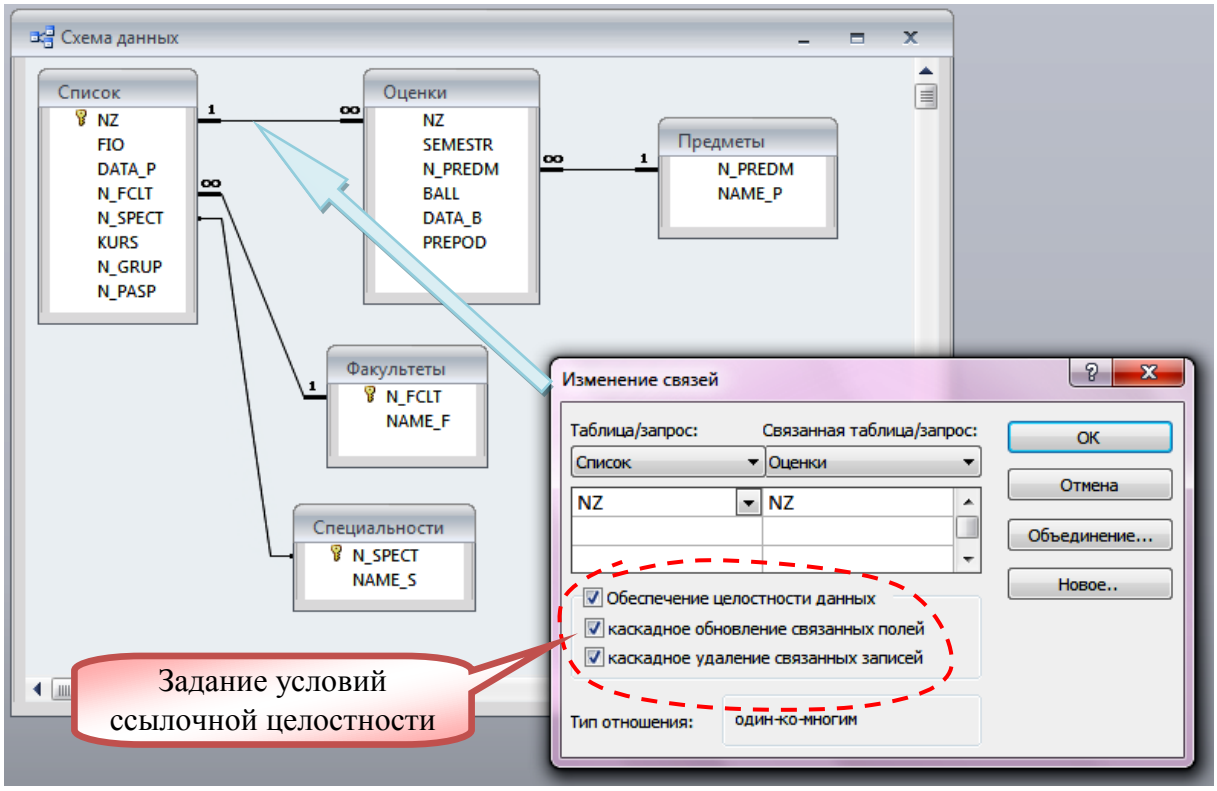


Рисунок 3.57. Схема базы данных и задание условий ссылочной целостности для связи Список – Оценки

Стандартный режим работы с таблицами

Долее можно занести некоторые данные в базу в стандартном режиме работы с таблицами, который в системе **Access** имеет достаточно большие возможности. Настройка внешнего вида таблиц выполняется в окне **Параметры Access ► Таблица**, которое вызывается с панели кнопки **Office**.

Заполнение базы информацией следует начинать со справочников – иначе при заполнении главных таблиц возникнут конфликты ссылочной целостности базы. Например, если в справочнике факультетов не будет номера факультета, указанного для студента, появится сообщение *Введенное значение не подходит для данного поля* и *Вы не сможете сохранить данные, пока не укажете правильное значение.*

Таблица **Предметы** в режиме работы с данными (двойной щелчок мышкой на ее названии в Области переходов или команда **Открыть** контекстного меню) показан на рисунок 3.58.

В данном режиме каждая запись таблицы базы данных представлена как строка, состоящая из столбцов – полей, над которыми показаны подпи-

си полей или, при их отсутствии, имена полей. В нижней части таблицы присутствует пустая запись с символом * в левой колонке – это несуществующая запись, которая добавляется в таблицу, как только в ней появляется какая-либо информация. На нижней рамке окна присутствуют кнопки для перемещения по таблице, номер текущей записи и информация об общем количестве записей в таблице.

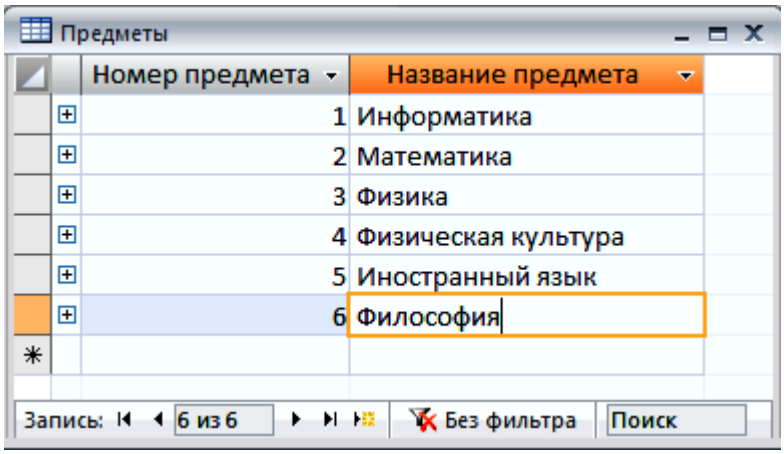


Рисунок 3.58. Работа с таблицей базы данных

В версии 2007 появилась строка быстрого поиска и сведения об установленном фильтре отбора данных. Колонка слева с символом ⊞ присутствует, если у таблицы есть связь от одной записи данной таблицы к многим записям связанной с ней таблицы. В данном случае при щелчке мышью на плюсе откроется подтаблица оценок для выбранного предмета.

Настройка списка отображаемых в таблице полей может быть выполнена в окне **Отображение столбцов** (см. рисунок 3.59), которое вызывается из вкладки **Главная**, группа **Записи**, раздел **Дополнительно**.

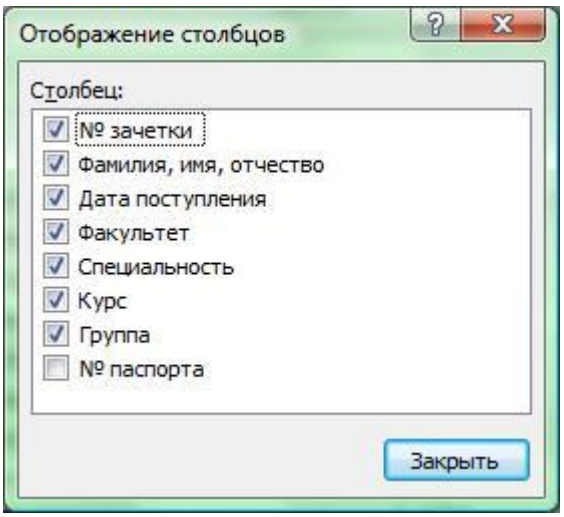


Рисунок 3.59. Настройка отображения полей таблицы

При работе с таблицей можно задать сортировку записей по одному из полей и фильтр для показа данных, соответствующих заданному условию (см. вкладка ленты **Главная**, группа **Сортировка и фильтр**).

После окончания добавления, редактирования или удаления данных следует дать команду **Сохранить** или при закрытии окна таблицы выбрать в появившемся окне команду сохранять

или нет изменения.

3.3.4.4 Разработка форм

Экранные формы позволяют организовать наглядную и удобную работу с базой данных, состоящей из нескольких связанных таблиц. В этом случае на одной форме можно организовать работу с главной и подчиненными таблицами, выбор данных из таблиц-справочников с использованием раскрывающихся списков, режимы поиска и отбора информации, печати необходимых отчетов на принтере и пр. В крупных автоматизированных системах управления предприятиями основной режим работы пользователей с информацией базы данных – экранные формы, которых может быть несколько тысяч. При этом на одной форме используются только некоторые из полей одной или нескольких таблиц, необходимые одному из пользователей системы. Каждый пользователь информационной системы может иметь доступ к ограниченному набору форм и полей базы данных этой системы, таким образом в системе может быть организовано множество разнообразных рабочих мест со своими полномочиями по вводу, редактированию и просмотру данных.

В системе **Access** 2007, как и в предыдущих версиях этой системы, имеется **Мастер** разработки экранных форм (см. рисунок 3.60), который позволяет легко создавать формы нескольких видов (простые для работы с данными одной таблицы, более сложные для работы с несколькими таблицами с использованием подчиненных форм). Полученные формы далее, как правило, приходится дополнять и модифицировать в **Конструкторе** экранных форм для реализации всех необходимых условий работы с данными.

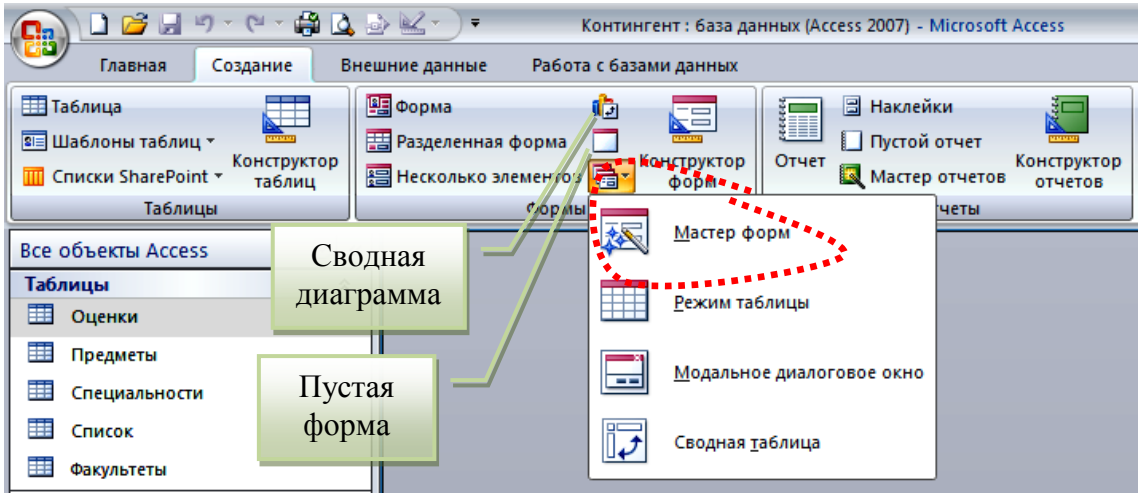


Рисунок 3.60. Мастер форм на вкладке **Создание** ленты **Access 2007**

Новый режим работы с формами – **Макет**, в котором одновременно с просмотром данных базы можно модифицировать ее внешний вид и добавлять поля данных.

Кроме **Мастера форм** для создания формы можно использовать другие варианты (см. рисунок 3.60): **Несколько элементов** (одновременно несколько записей на форме в виде таблицы), **Разделенная форма** (верхняя часть – одна запись, нижняя – таблица записей) и **Форма**. В последнем случае, если выбранная таблица имеет дочернюю (связь «один ко многим»), последняя будет показана на сгенерированной форме, как подчиненная форма. Однако, для подобных форм б**о**льшие возможности предоставляет использование **Мастера** разработки форм.

Для разработки экранной формы воспользуемся режимом **Мастер форм** (см. рисунок 3.60). В открывшемся окне **Создание форм** (рисунок 3.61) следует ответить на ряд вопросов **Мастера**:

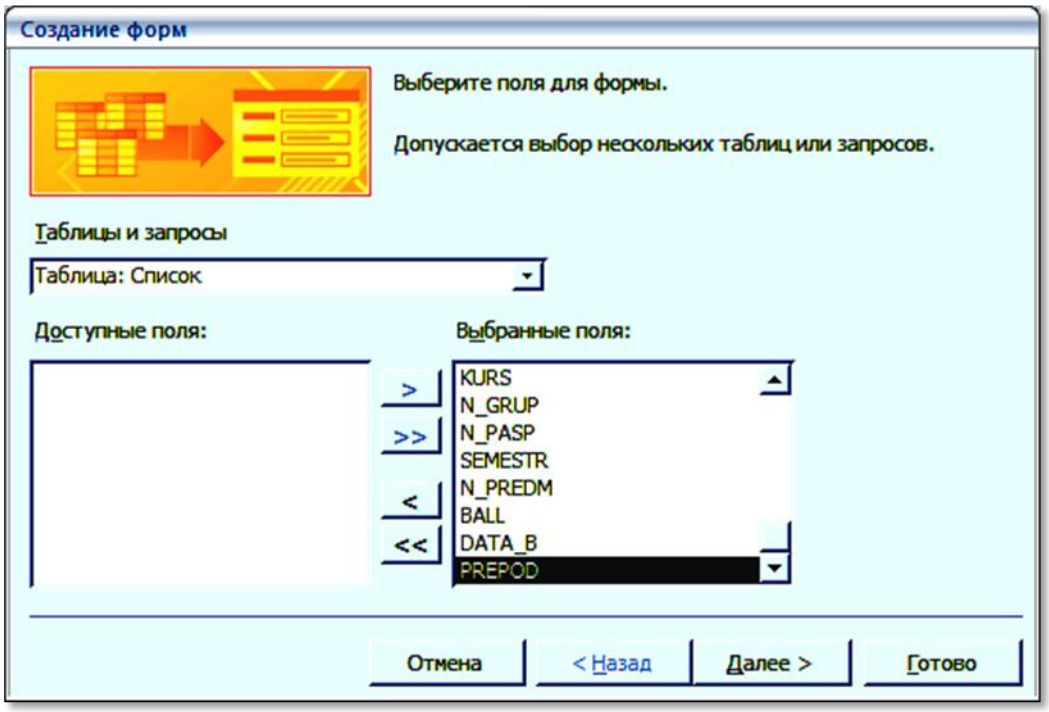


Рисунок 3.61. Первый шаг **Мастера** форм

1. *Выберите поля для формы* – выбираем все поля таблицы **Список** и все поля таблицы **ОЦЕНКИ**, кроме поля NZ (для последней таблицы поле **NZ** будет заполняться автоматически; добавьте его, если хотите убедиться, что на экране мы видим оценки только одного студента).
2. *Выберите вид представления данных* – выбираем **Подчиненные формы** – т. е. расположение данных главной таблицы и связанной с ней на одной форме.

3. *Выберите внешний вид подчиненной формы* – выбираем **ленточный**.
4. *Выберите требуемый стиль* – выбираем стиль **Access 2007**.
5. *Задайте имена форм* – задаем для главной формы название **Студенты**, для подчиненной формы **Оценки студента** и на том же экране ниже выбираем пункт **Изменить макет формы**, после чего нажимаем на кнопку **Готово**.

Полученная экранная форма будет открыта в конструкторе форм, в группе **Формы** области переходов базы данных (настройка области – тип объекта, все объекты) появится две новых формы – **Студенты** и **Оценки студента.**Для более детального отображения в конструкторе подчиненной формы лучше закрыть его окно и снова открыть в Конструкторе форму **Студенты**, в этом случае она будет иметь вид, приведенный на рисунок 3.62.

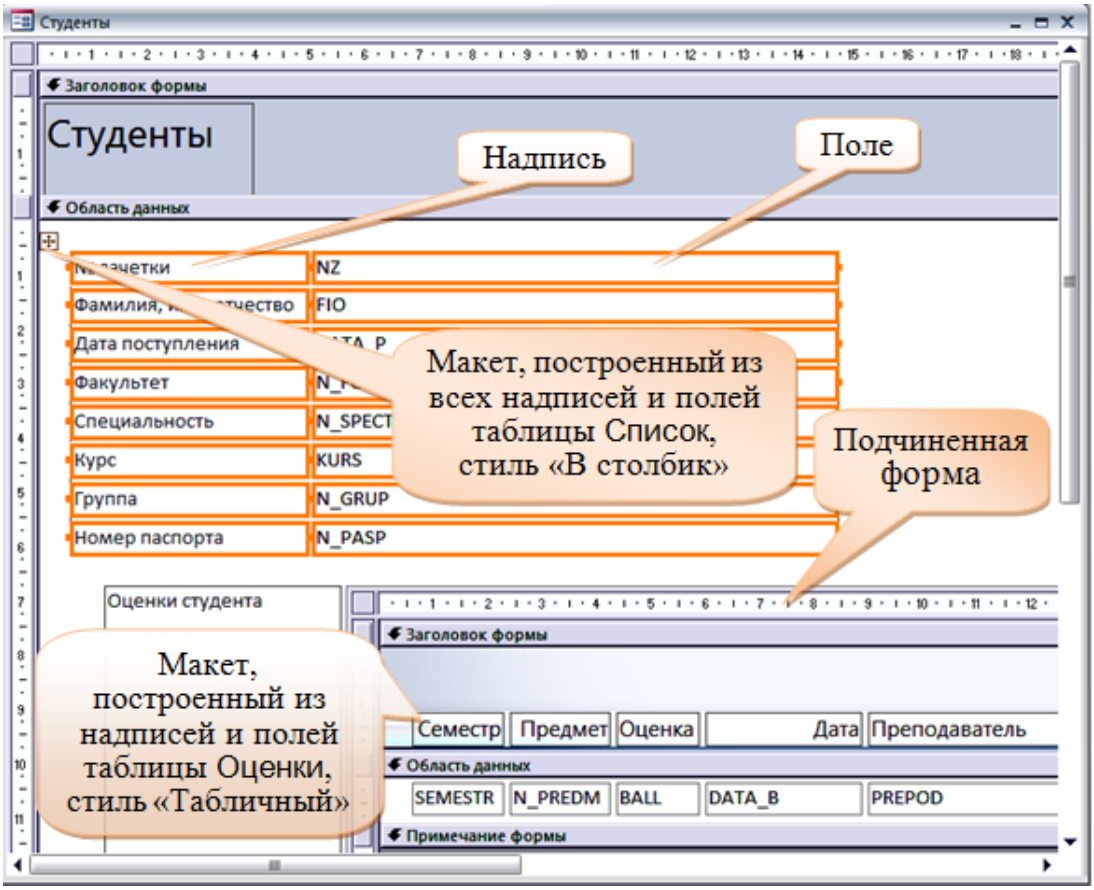


Рисунок 3.62. Форма, созданная с использованием **Мастера**

На форме элементы управления (надписи и поля) таблиц Список и Оценки объединены в 2 макета. Макет элемента управления – новый режим оформления полей и их подписей в стиле **Табличный** или **В столбик** (см.

вкладку **Упорядочить** контекстных групп вкладок **Инструменты конструктора форм** и **Работа с макетами форм**).

В результате ширина надписей и полей для макета **В столбик** выровнена «по самому широкому», такой вариант нельзя признать оптимальным. Размер формы достаточно большой – ширина 25,453 см, высота заголовка ~ 1 см, области данных – 13,429 см. Эти размеры будет иметь печатная форма, автоматически формирующаяся при выборе команды **Печать** (меню кнопки **Office** или на панели быстрого доступа).

Для формы и всех ее элементов можно открыть **Окно свойств**, щелкнув правой кнопкой мыши на любом объекте и выбрав в контекстном меню слово **Свойства** (рисунок 3.63).

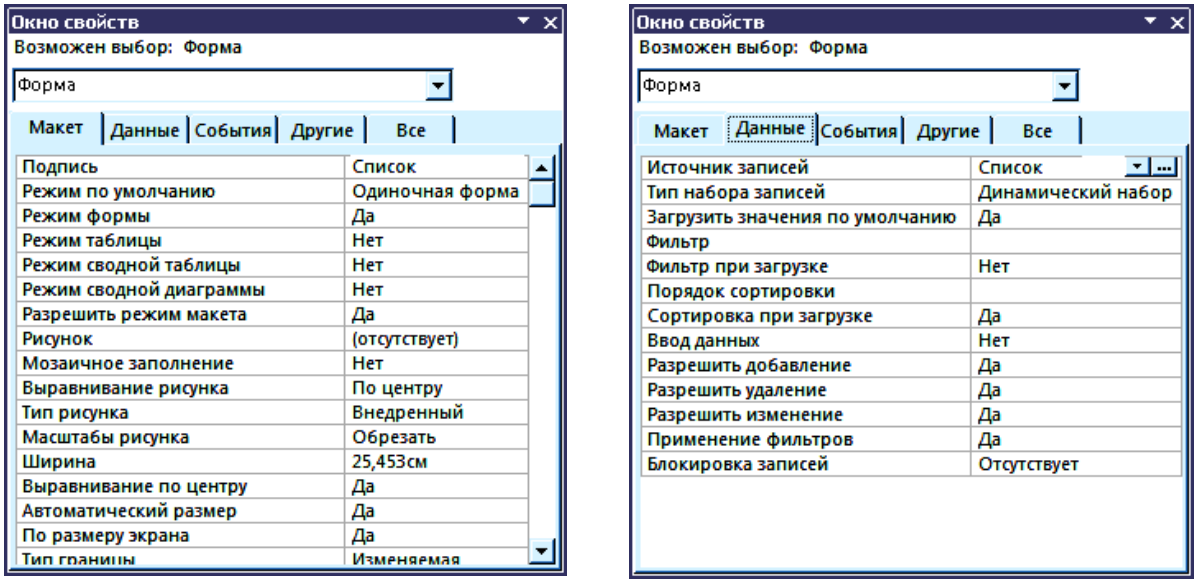


Рисунок 3.63. Окно свойств объекта **Форма**, вкладки **Макет** и **Данные**

Все свойства в окне разбиты на группы:

- ***Макет*** – расположение, шрифт, цвет и прочее, связанное с внешним видом объекта;
- ***Данные*** – важнейшее свойство в этом разделе – ***Источник записей или Данные*** (для поля) – для объектов, связанных с редактированием каких-либо данных (связанные объекты – поля, списки…);
- ***События*** – макросы, выражения и программы, выполняющиеся для объекта при наступлении определенных событий (**Загрузка, Открытие, До обновления, После обновления** и пр.);
- ***Другие*** – прочие свойства.

На экранной форме присутствуют элементы управления (объекты) следующих типов:

- **Надпись** – текст на форме, обычно не изменяющийся. Главные свойства этого объекта присутствуют на вкладке ***Макет*** окна свойств (рисунок 3.64).
- **Поле –** объект для редактирования данных, связанный с полем таблицы базы данных или с переменной. Главное свойство этого объекта – ***Данные*** на странице **Данные** окна свойств (рисунок 3.65), которое показывает, с какими данными связан этот объект формы. Контекстное меню объектов типа **Поле** имеет достаточно большое количество команд, как показано на рисунке 3.66.
- **Подчиненная форма –** вложенная форма для дочерней таблицы данных, связанной с главной таблицей, на которой могут присутствовать такие же элементы, как и на основной форме.

Кроме того, на форме могут присутствовать и другие объекты, которые можно добавлять, используя группу кнопок Э**лементы управления** контекстной вкладки ***Конструктор*** (рисунок 3.67):

- **Группа переключателей, Переключатель, Кнопка, Выключатель** – кнопки и группы кнопок разного вида, связанные с выполнением определенных процедур. При выборе объекта **Кнопка** запускается **Мастер**, который предложит для выбора набор стандартных кнопок перехода по записям таблицы, обработки записей (восстановить, дублировать, печатать и пр.), работы с формой, с отчетами (печатать, просмотреть, отправить в файл или по почте), работы с приложениями и разное. **Кнопки** можно создать с помощью мастера, при этом программный код, связанный с их действием, уже будет присутствовать (рисунок 3.68).
- **Флажок** – поле, связанное обычно с полем таблицы логического типа, в котором стоит галочка или нет.
- **Поле со списком** – сложный элемент, предоставляющий возможность показывать данные справочных таблиц, списков или массивов и заносить выбранные значения в поле другой таблицы.
- **Список** – список данных для выбора одного из значений.
- **Рисунок** – вставка рисунка в форму.
- **Свободная рамка объекта** – любой объект **Windows**-приложений, редактирование которого будет возможно вызовом соответствующего приложения.
- **Присоединенная рамка объекта** – для работы с полями таблиц типа поле объекта **OLE.**

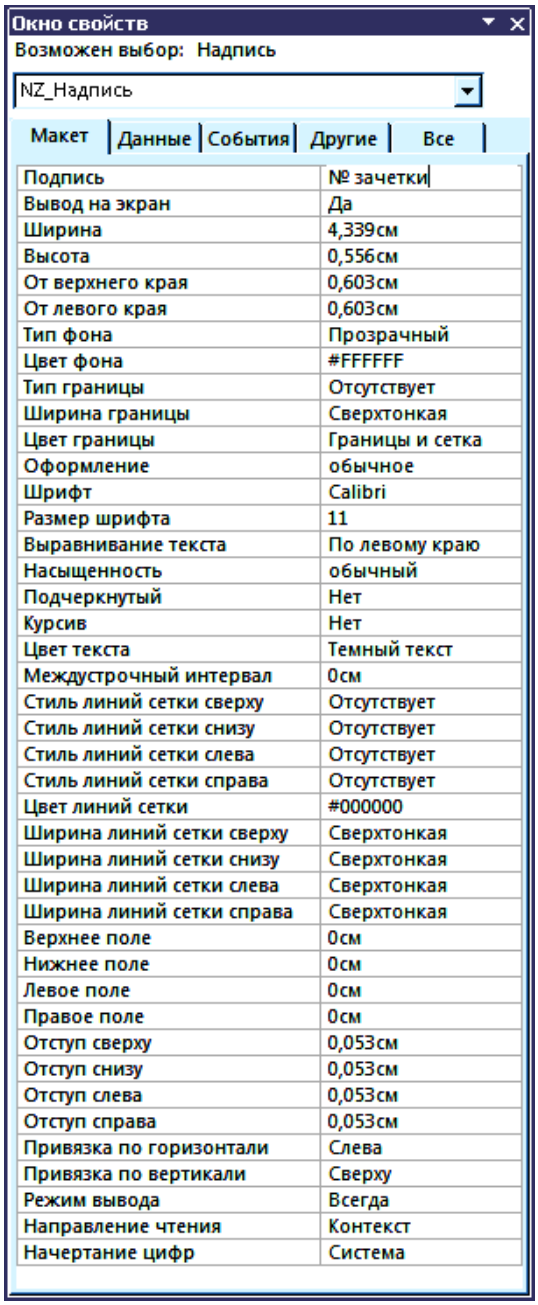


Рисунок 3.64. Группа свойств *Макет* объекта типа **Надпись**

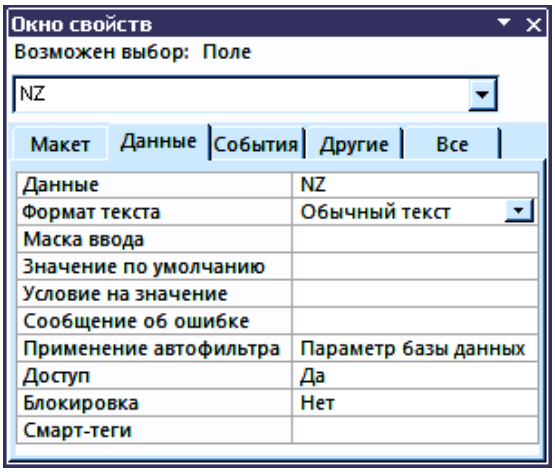


Рисунок 3.65. Окно свойств объекта типа **Поле,** группа ***Данные***

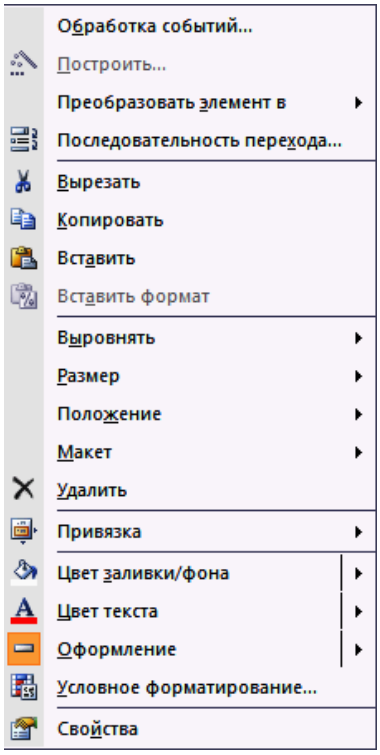


Рисунок 3.66. Контекстное меню объекта типа **Поле**

- **Набор вкладок** – многостраничная форма.
- **Линия, Прямоугольник** – элементы оформления.
- **Вложение –** объект, появившийся в версии **Access 2007** для работы с полями типа *Вложение*.

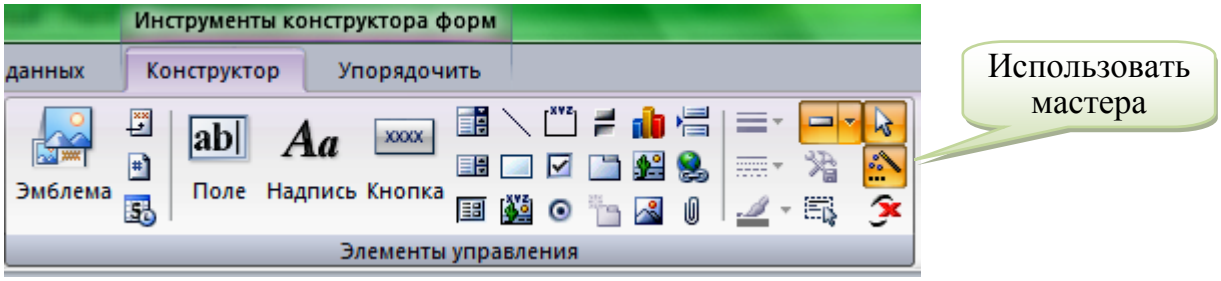


Рисунок 3.67. Элементы управления вкладки **Конструктор**

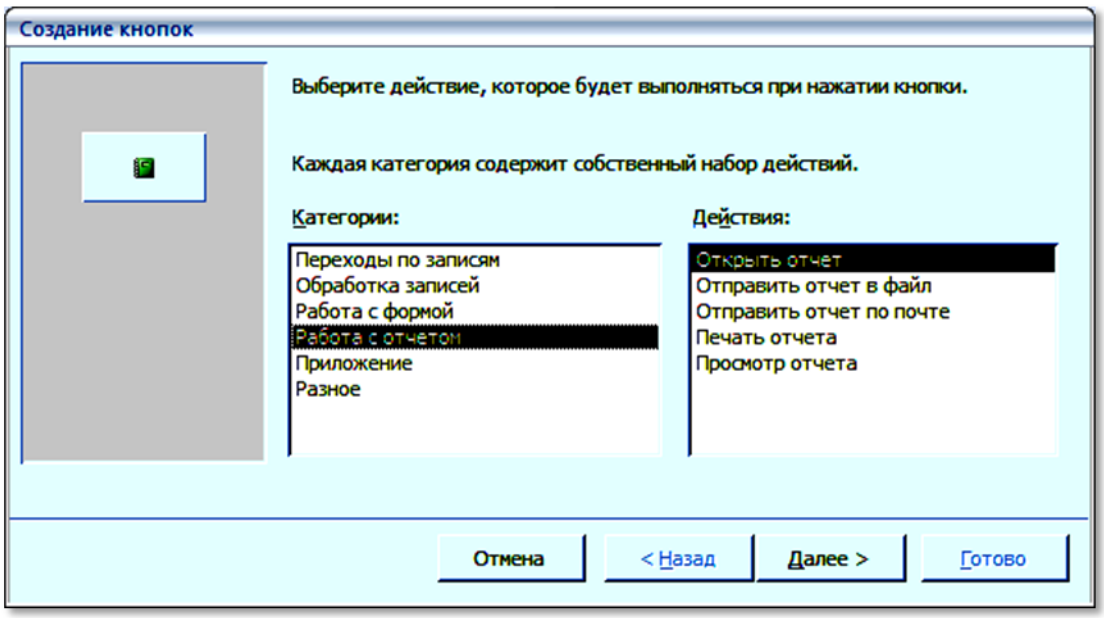


Рисунок 3.68. Окно Мастера создания кнопок

Для использования экранной формы нужно запустить ее в работу с данными, для этого на вкладке **Главная** в группе ***Представления*** выберем Вид – **Режим формы**. Вид формы при работе с базой данных приведен на рисунок 3.69.

В данном режиме можно редактировать существующие записи, добавлять новые в список студентов и в список оценок для каждого студента. Для удаления записей можно использовать контекстные меню для вертикальной полосы слева в главной форме и для метки слева подчиненной формы. Поле быстрого поиска на нижней рамке окна позволяет найти и выделить фрагменты значений полей для главной формы.

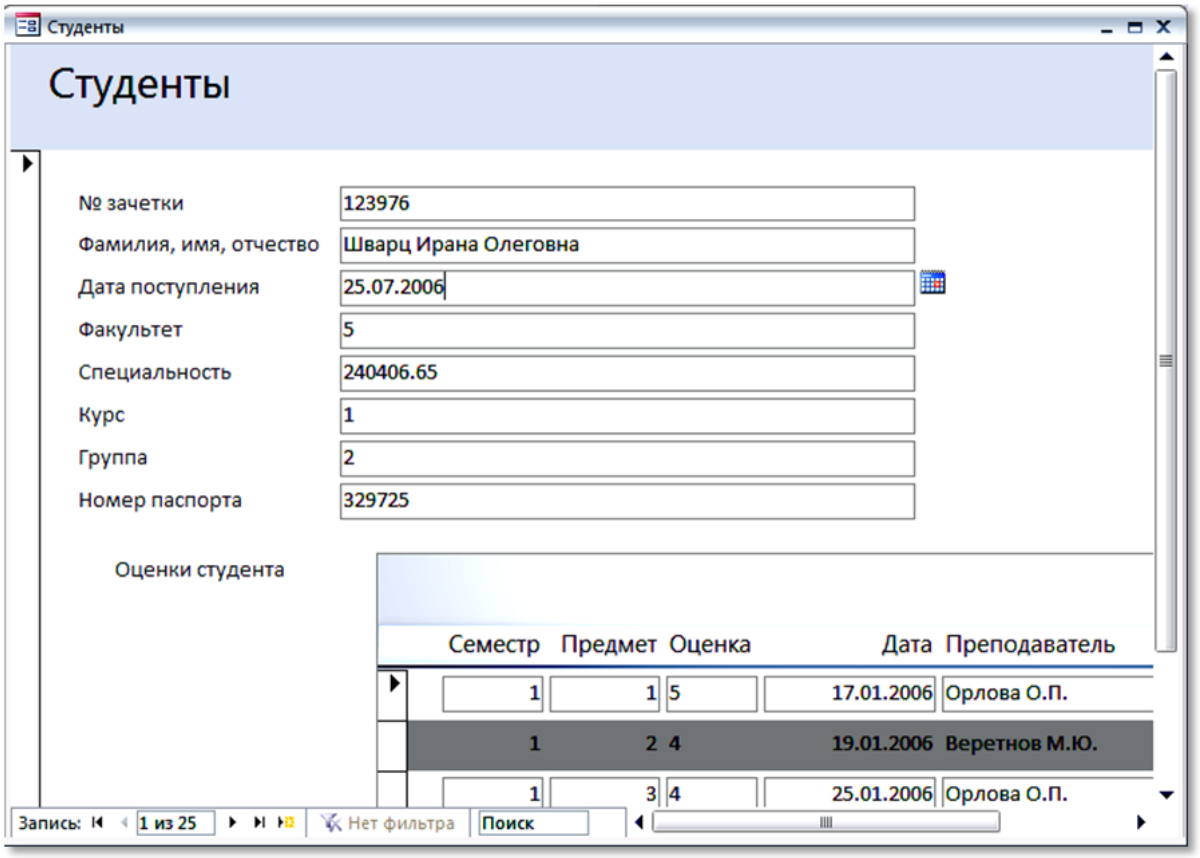


Рисунок 3.69. Работа с базой данных с использованием экранной формы, созданной с использованием **Мастера**

При работе с формой можно задать сортировку записей по одному из полей и фильтр для показа только тех данных, которые соответствуют заданному условию (см. вкладка **Главная,** группа ***Сортировка и фильтр***).

Как было сказано выше, для данной формы можно сделать значительные усовершенствования, повышающие удобства работы с базой данных.

Прежде всего, в **Конструкторе** выполним отмену объединения в единый макет полей таблиц **Список** и **Оценки** (кнопка **Удалить макет** в группе ***Макет*** контекстной вкладки **Упорядочить**). После этого можно изменить размер и расположение полей. Изменим также чередование цветов в записях таблиц (свойство **Цвет фона чередования Области данных**).

Далее добавляем на форму объекты **Поле со списком** для выбора из справочных таблиц факультета, специальности (с занесением соответствующих номеров в главную таблицу) и предмета (с занесением его номера из справочника в таблицу оценок). Эти основные свойства описываются с

помощью **Мастера**, который запускается автоматически при добавлении этого типа объекта к форме, если с группе кнопок Элементы управления включена кнопка *Использовать мастера*. Для добавления нового объекта необходимо выбрать его тип в группе ***Элементы управления*** вкладки **Конструктор**, затем показать его расположение и размер на форме.

На первом шаге **Мастера** выбираем пункт ***Объект "поле со списком" будет использовать значения из таблицы или запроса*** (см. рисунок 3.70).

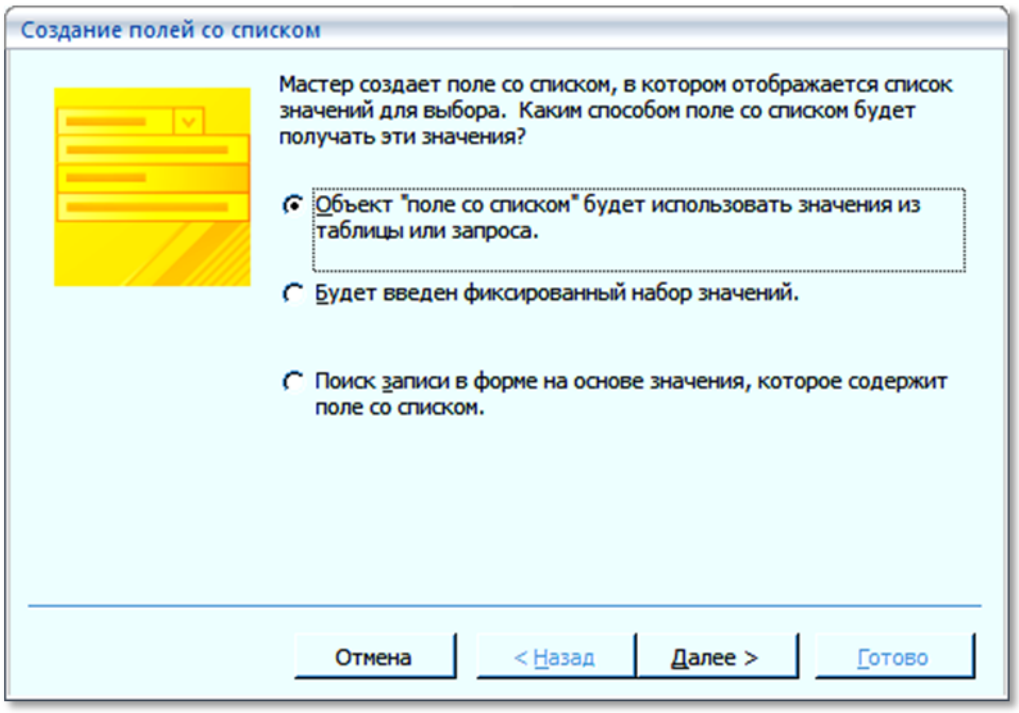


Рисунок 3.70. Первый шаг **Мастера** создания **Поля со списком**

На втором – выбираем из списка нужную нам справочную таблицу, на третьем – выбираем все поля (номер и название), на четвертом шаге – оставляем галочку у флажка ***скрыть ключевой столбец*** и задаем ширину поля для названия (см. рисунок 3.71).

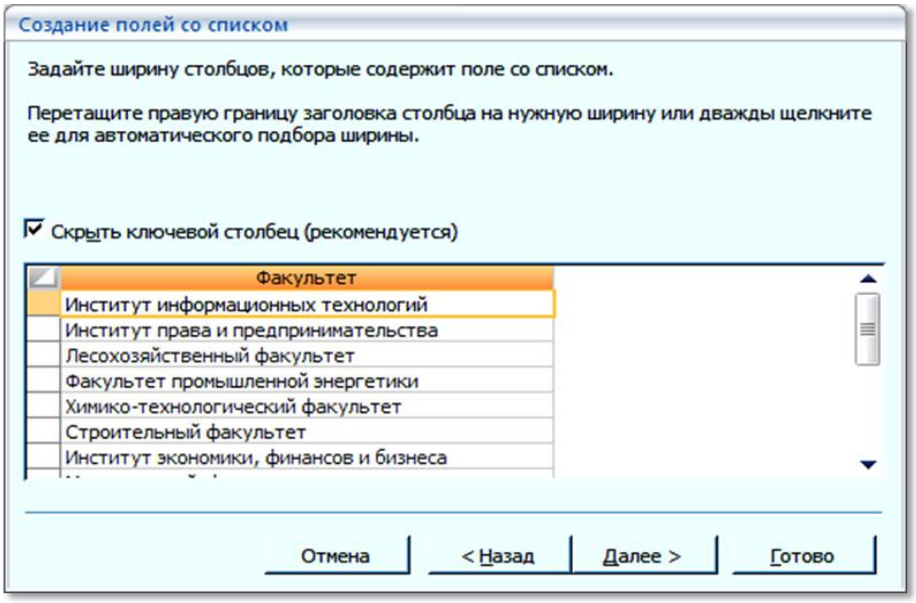


Рисунок 3.71. Четвертый шаг **Мастера** создания **Поля со списком**

Пятый шаг – задаем условие ***сохранить в поле*** и выбираем из списка поле главной таблицы, в котором будет сохраняться значение ключевого поля справочной таблицы (см. рисунок 3.72).

Если Вы не укажите поле таблицы, с которым связано поле со списком, все Ваши предыдущие шаги бессмысленны для данной задачи!!!

Рисунок 3.72. Последний шаг **Мастера** создания **Поля со списком**

Далее нажимаем **Готово**. **Надпись** для **Поля со списком** на форме можно удалить.

Наиболее важные свойства **Поля со списком** для работы со справочником факультетов, описанные с помощью **Мастера**, показаны на рисунке 3.73.

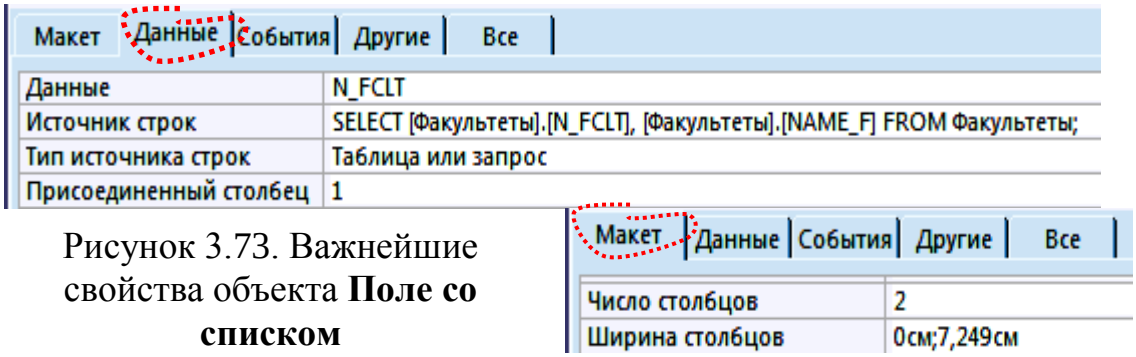


Рисунок 3.73. Важнейшие свойства объекта **Поле со списком**

После модификации форма будет иметь в **Конструкторе** вид, приведенный на рисунок 3.74.

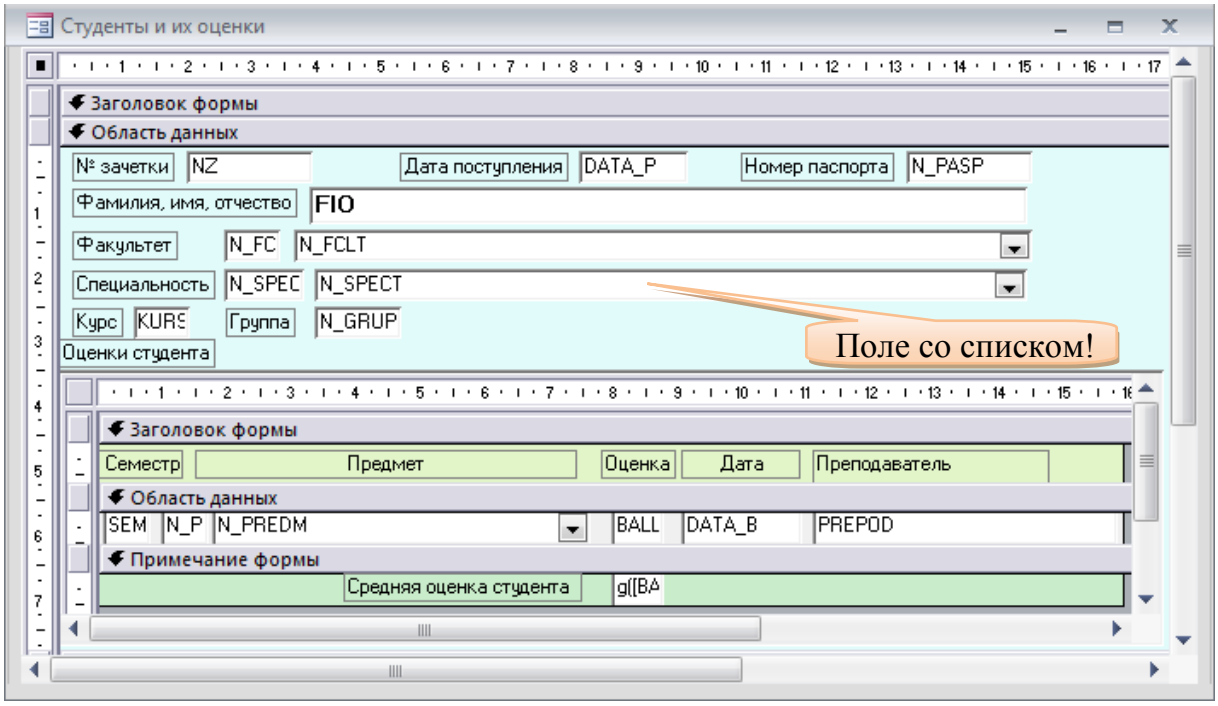


Рисунок 3.74. Экранная форма, модифицированная в **Конструкторе**

Цветовые решения формы легко изменить с использованием режима **Автоформат** в режиме формы **Макет**, для каждого элемента формы также можно использовать цветовые решения тем системы **Access**. Для подчиненной таблицы оценок изменено свойство Разделительные линии – Да. Кроме того, в зоне **Примечание формы** таблицы оценок добавлено вы-

числяемое поле (копированием поля BALL и его изменением) его свойство которого **Данные** = *AVG(BALL),* а также **Надпись** к нему (копированием другой надписи в таблице, чтобы сохранить стиль шрифта).

В режиме работы с базой данных в усовершенствованной форме можно видеть названия факультетов, специальностей, предметов и выбирать их из справочных таблиц с помощью раскрывающихся списков, а также просматривать средние оценки студентов (рисунок 3.75).

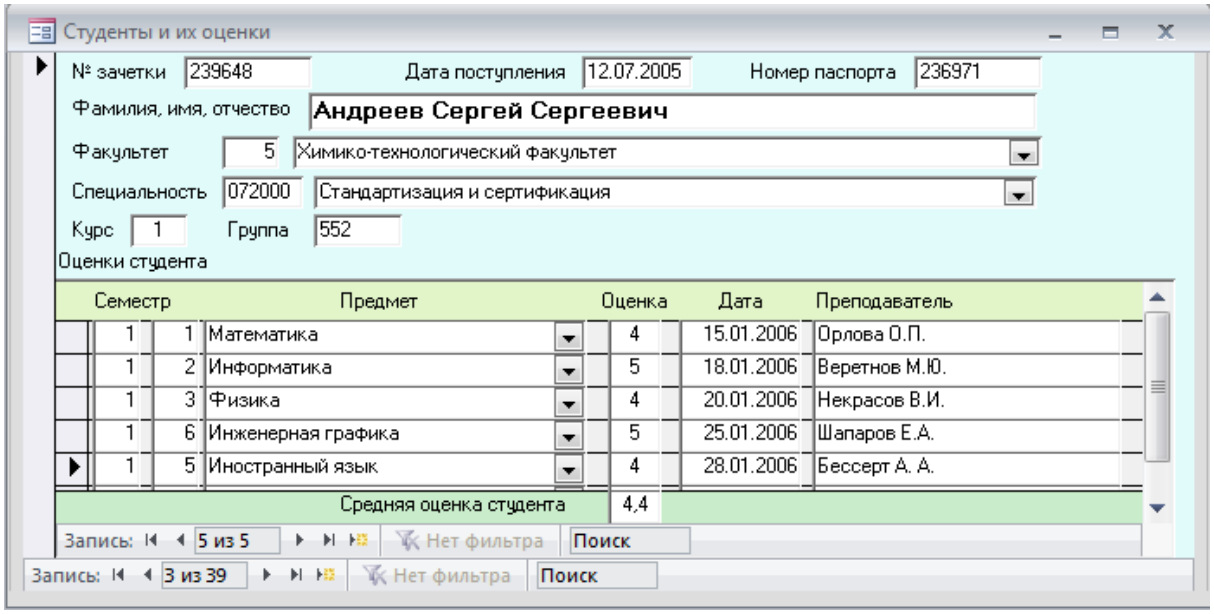


Рисунок 3.75. Усовершенствованная экранная форма с объектами *Поле со списком* и расчетом средней оценки студента

3.3.4.5 Разработка отчетов

Для разработки отчетов, отражающих информацию базы данных в печатных документах, в системе **Access** можно **Мастер отчетов** и **Конструктор отчетов**. Отчеты могут быть созданы на основе всей информации, присутствующей в таблицах базы, но чаще для отчетов необходимо отобрать нужную информацию из базы с использованием SQL запроса и на основе его создать отчет. Важным свойством отчетов является возможность группировки данных и получения итоговых данных для групп и всего отчета.

Печатные документы отчетности предприятий часто должны иметь точное соответствие с унифицированными формами, утвержденными Госкомстататом РФ.

Поставим задачу разработать отчет, в котором показаны оценки всех студентов с группировкой данных по факультетам, курсам, группам. Для

этого на вкладке **Создание** в группе **Отчеты** нажимаем кнопку **Мастер отчетов**.

На первом шаге **Мастера** выбираем поля главной таблицы базы (**Список**), которые мы хотим показать в отчете и все поля дочерней таблицы оценок (**Оценки**) (рисунок 3.76).

На втором шаге *Выберите вид представления данных* – выбираем первый вариант, когда выделена таблица **Список**.

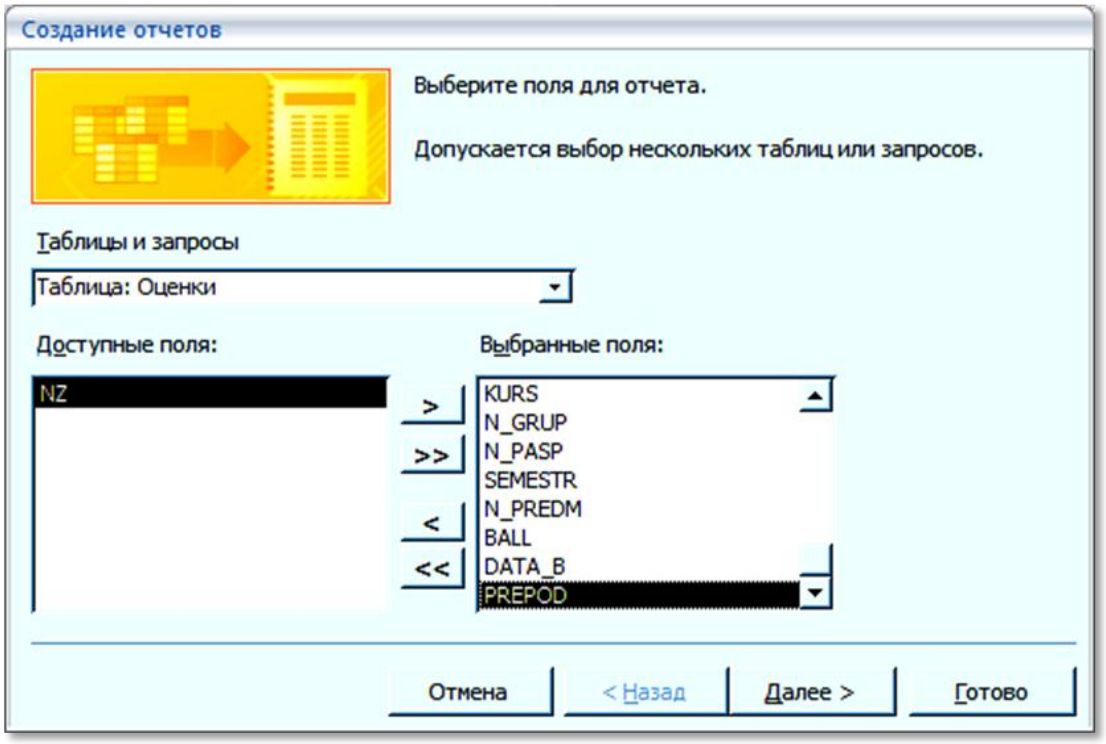


Рисунок 3.76. Выбор полей для отчета в **Мастере отчетов**

На третьем шаге добавляем группировку данных по факультетам, курсам и группам (см. рисунок 3.77). Более трех уровней группировки мастер задать не позволяет, однако это можно сделать при необходимости в **Конструкторе отчетов**. Сортировку на следующем шаге не задаем.

На 5-м шаге выбираем вид макета для отчета – *структура*, на следующем – стиль отчета – ***Access** 2007*, далее задаем название отчета **Студенты и их оценки** и нажимаем кнопку **Готово**. Полученный отчет в режиме **Конструктора** представлен на рисунок 3.78.

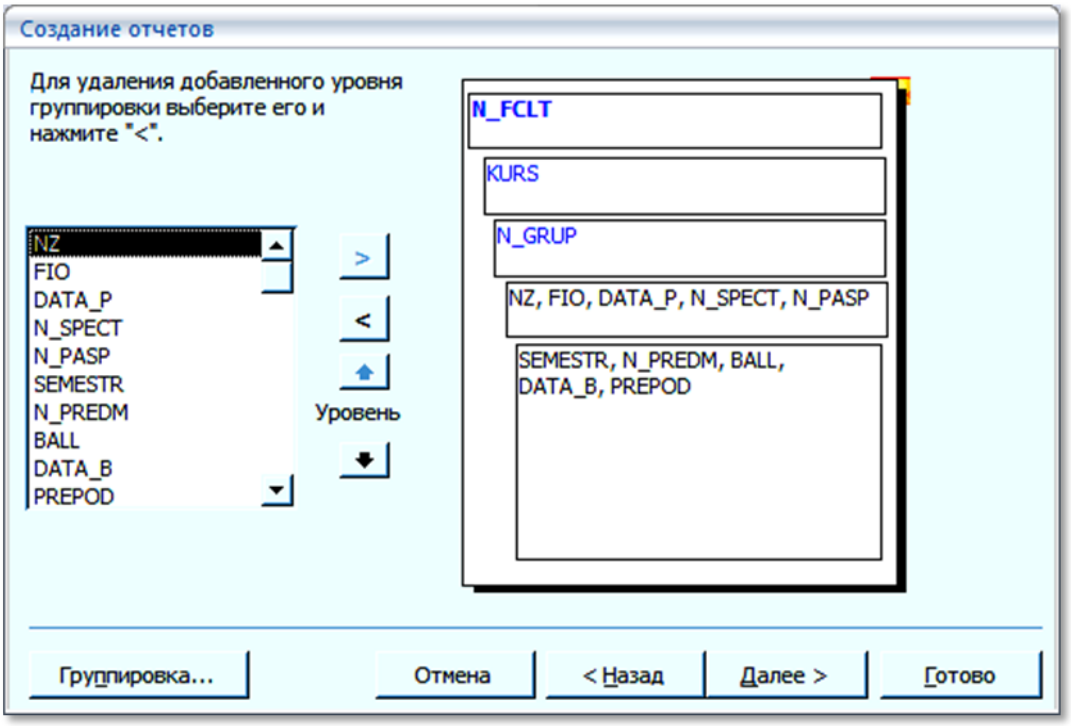


Рисунок 3.77. Добавление уровней группировки для отчета

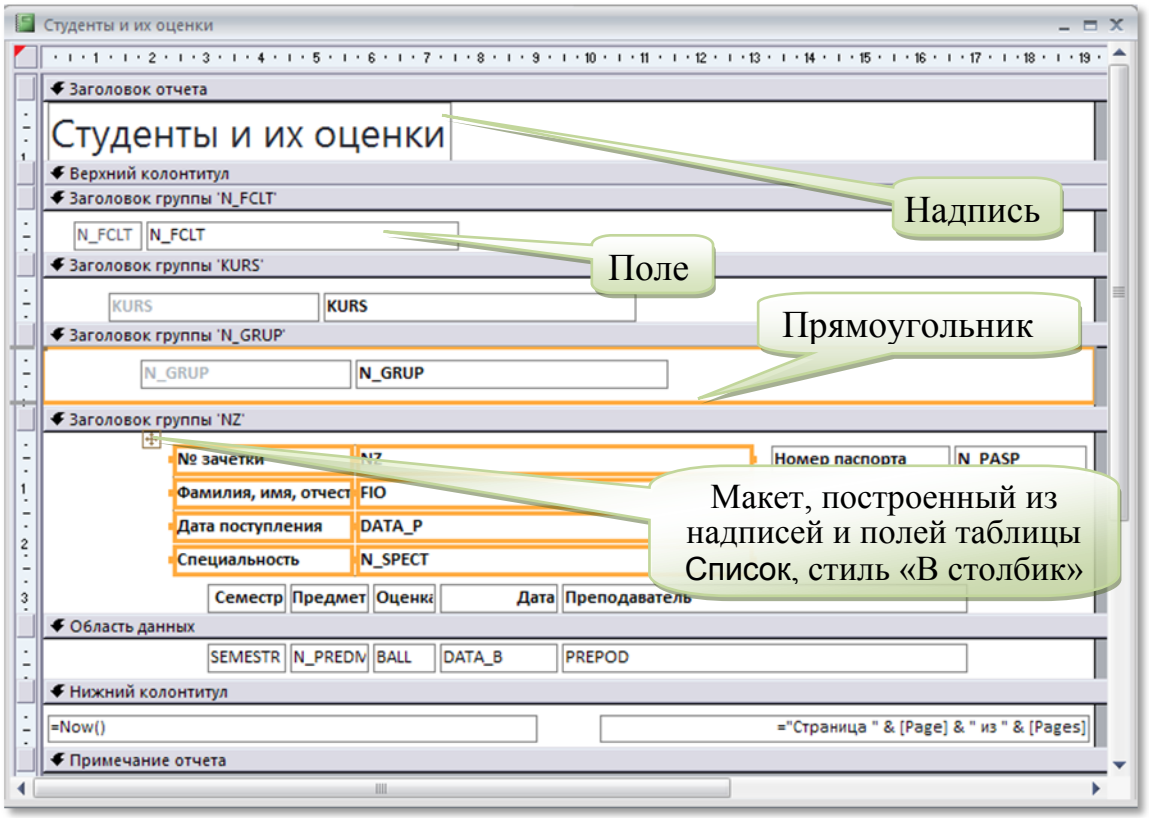


Рисунок 3.78. Отчет, созданный с использованием **Мастера отчетов**

В полученном отчете присутствуют объекты 3-х видов – **Поле**, которое в отчете может показывать данные поля таблицы базы, значение переменной или выражения, **Надпись** – любой текст в отчете и **Прямоугольник** – элемент оформления.

В **Конструкторе** отчет разбит на отдельные зоны, информация которых может присутствовать в отчете один раз (***Заголовок отчета*** и ***Примечание отчета***), в начале каждой страницы (***Верхний колонтитул***) или в конце каждой страницы (***Нижний колонтитул***), в начале каждой группы (***Заголовок группы***, групп может быть много) и в конце каждой группы (***Примечание группы***) и для каждой записи главной таблицы отчета (***Область данных***). Зоны ***Примечания…*** более правильно было бы назвать ***Итоги…***, т. к. здесь можно разместить поля общих итогов (сумма, среднее и пр.) для группы или всего отчета.

Полученный отчет можно просмотреть на экране, отправить на принтер (например, с использованием соответствующих кнопок на стандартной панели инструментов) или в файл rtf -формата.

Начало отчета в режиме просмотра и при печати на принтере приведено на рисунок 3.79.

Студенты и их оценки

N_FCLT 3

KURS 1

N_GRUP 4

№ зачетки 597126 Номер паспорта 1

Фамилия, имя, отче Моисеенко Анастасия Владимировна

Дата поступления 16.08.2006

Специальность 220501.65

Семестр	Предмет	Оценк	Дата	Преподаватель
1	4	4	17.01.2006	Шапаров Е.А.
1	1	4	20.01.2006	Орлова О.П.
1	3	5	23.01.2006	Рай А.И.
1	5	5	15.01.2006	Чернышова О.В.
1	2	3	25.01.2006	Веретнов М.Ю.

№ зачетки 597364 Номер паспорта 1

Фамилия, имя, отче Плотникова Ольга Валерьевна

Рисунок 3.79. Отчет, созданный Мастером, при печати его на принтере

Созданный **Мастером** отчет весьма несовершенен, его ширина превышает размер формата А4. В отчет необходимо добавить названия факультетов, специальностей и предметов и изменить их надписи.

Для добавления новых и модификации существующих объектов следует использовать вкладку **Конструктор** контекстной группы вкладок **Инструменты конструктора отчетов**, для настройки расположения объектов – вкладку **Упорядочить**, для настройки параметров страницы (полей и размера бумаги) – вкладка **Параметры страницы** (см. рисунок 3.80).

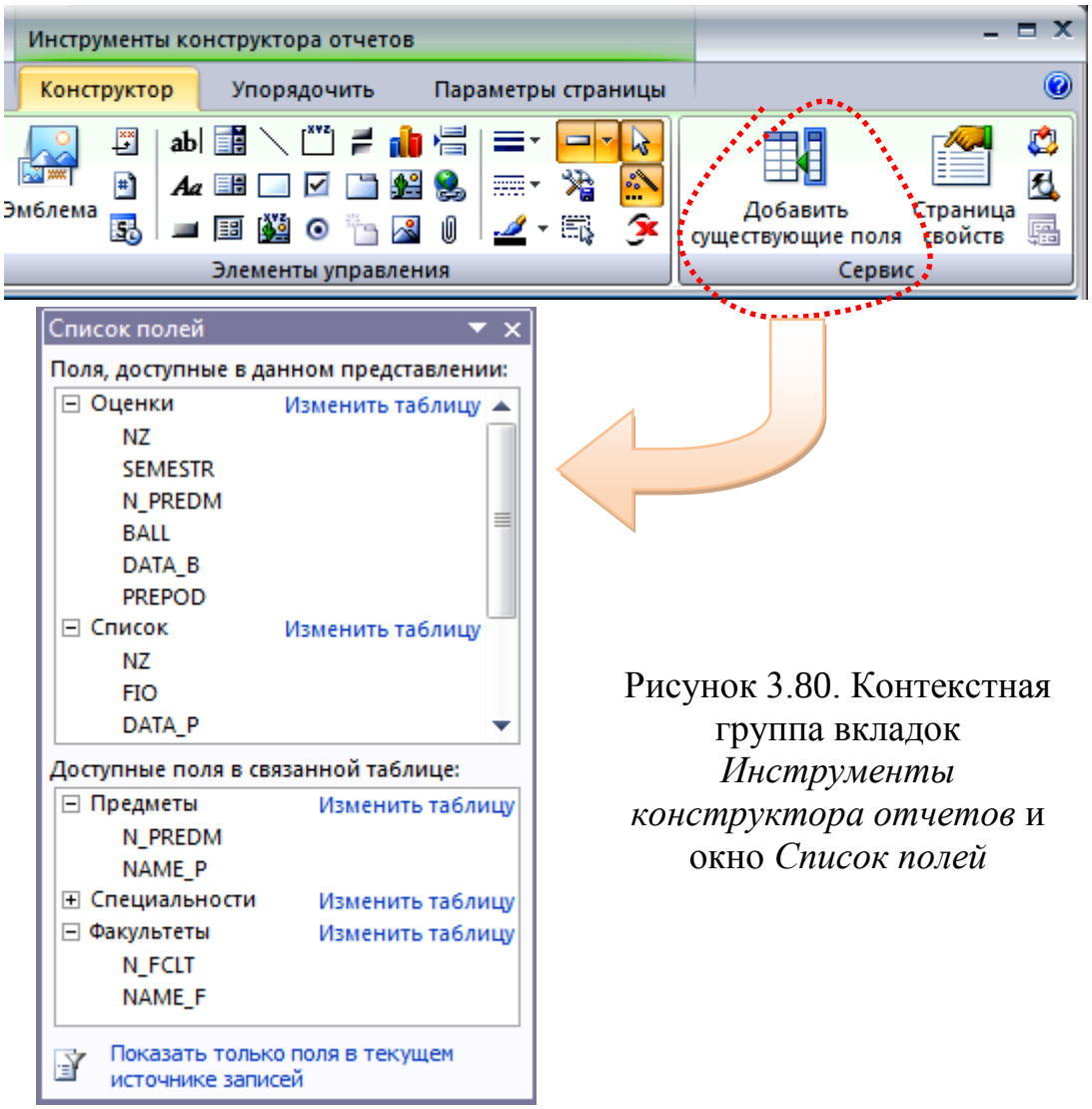


Рисунок 3.80. Контекстная группа вкладок *Инструменты конструктора отчетов* и окно *Список полей*

Для использования в отчете данных справочных таблиц следует выбрать команду ***Добавить существующие поля*** на вкладке **Конструктор** (см. рисунок 3.80), на нижней рамке окна **Список по**лей нажать на ссылку ***Показать все таблицы***, после чего перетащить поля **NAME_F**, **NAME_S**

и **NAME_P** в соответствующие зоны отчета. В результате изменится ***Источник записей*** **Отчета**, что можно увидеть в **Окне свойств** (рисунок 3.81). При нажатии на кнопку … в строке ***Источник записей*** открывается окно построителя запросов, которое после добавления нужных полей имеет вид, показанный на рисунок 3.82.

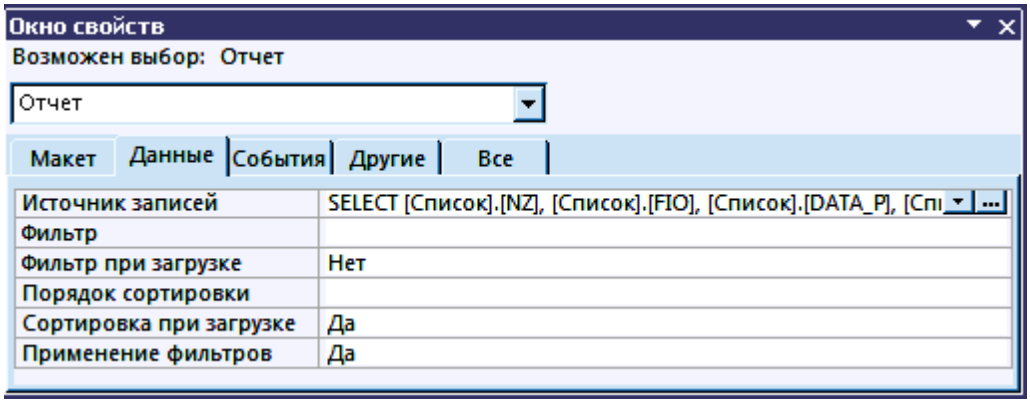


Рисунок 3.81. Окно свойств Отчета, вкладка Данные

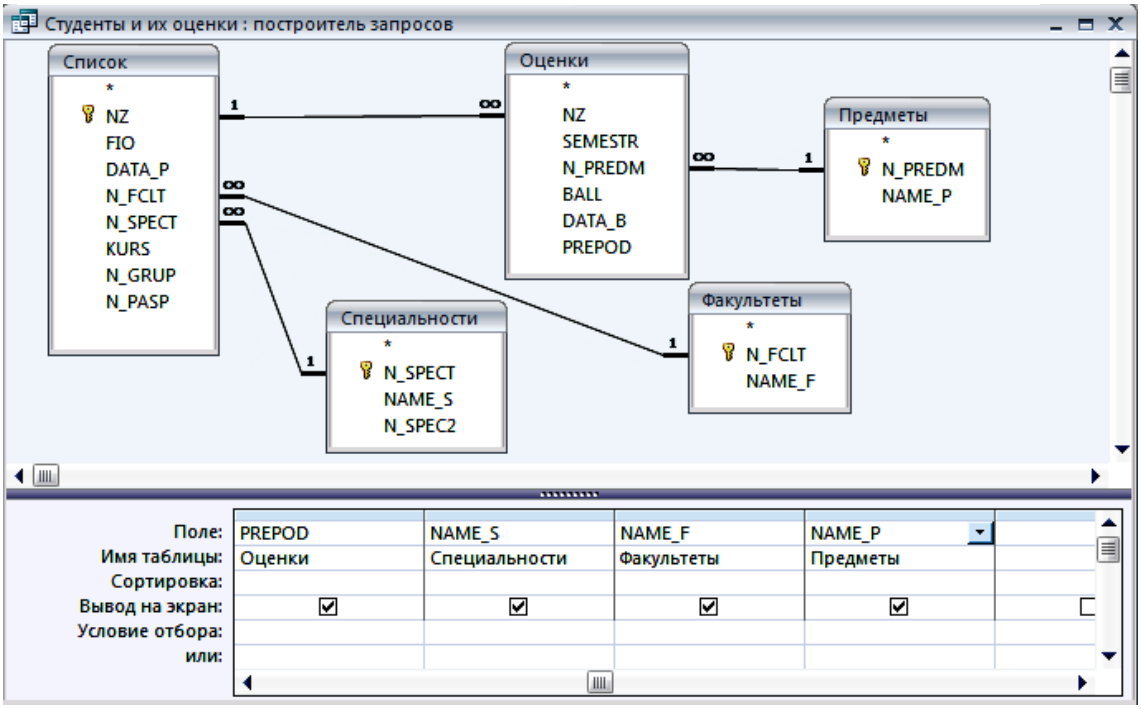


Рисунок 3.82. Окно Построителя запросов *Источника записей* Отчета

Кроме того, необходимо задать наличие **Примечаний** для всех групп, в зону примечаний поместить вычисляемые поля, которые будут

показывать средние оценки для студента, группы, факультета и для всего отчета. Для этого на вкладке **Конструктор** группы *Инструменты конструктора форм* следует в группе команд *Группировка и итоги* выбрать кнопку *Группировка и сортировка,* после чего в нижней части окна **Access** откроется панель Группировка, сортировка и итоги (см. рисунок 3.83).

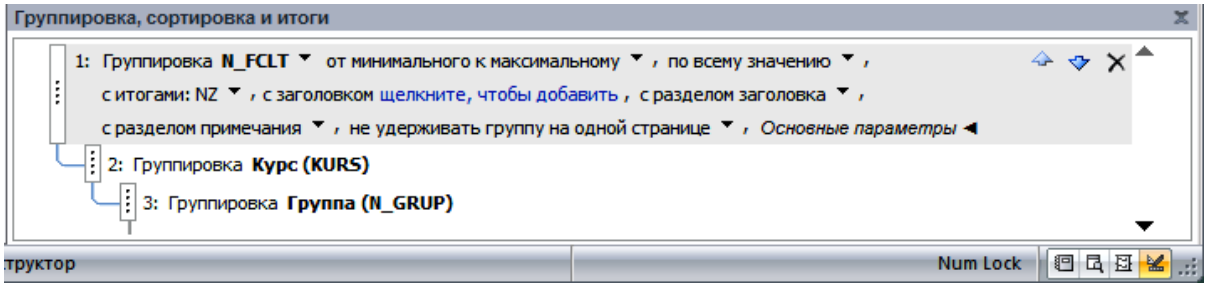


Рисунок 3.83. Панель Группировка, сортировка и итоги Отчета

Для каждого уровня группировки нажмем кнопку ***Больше*** ►, в появившихся параметрах выберем *с разделом примечания* и *с итогами*. Далее в **Конструкторе** Отчета скопируем в появившуюся зону примечаний для группы NZ поле BALL, в разделе *Данные* его Окна свойств изменим выражение на =Avg(BALL) для вычисления среднего значения поля, зададим Формат поля *фиксированный* и Число десятичных знаков = 2, затем скопируем измененное поле в зоны примечаний других групп.

При создании вычисляемых полей в пункте **Данные** Окна свойств можно нажать на кнопку с многоточием, после чего откроется окно **Построителя выражений**, где можно выбрать для построения выражений имеющиеся в системе функции и другие компоненты.

Разместим в отчете также надписи для итоговых значений, разлинуем таблицу (при этом для линий нужно задать свойство *Тип границы – Сплошная* и ее цвет).

Вид отчета в **Конструкторе** после его модификации показан на рисунке 3.84.

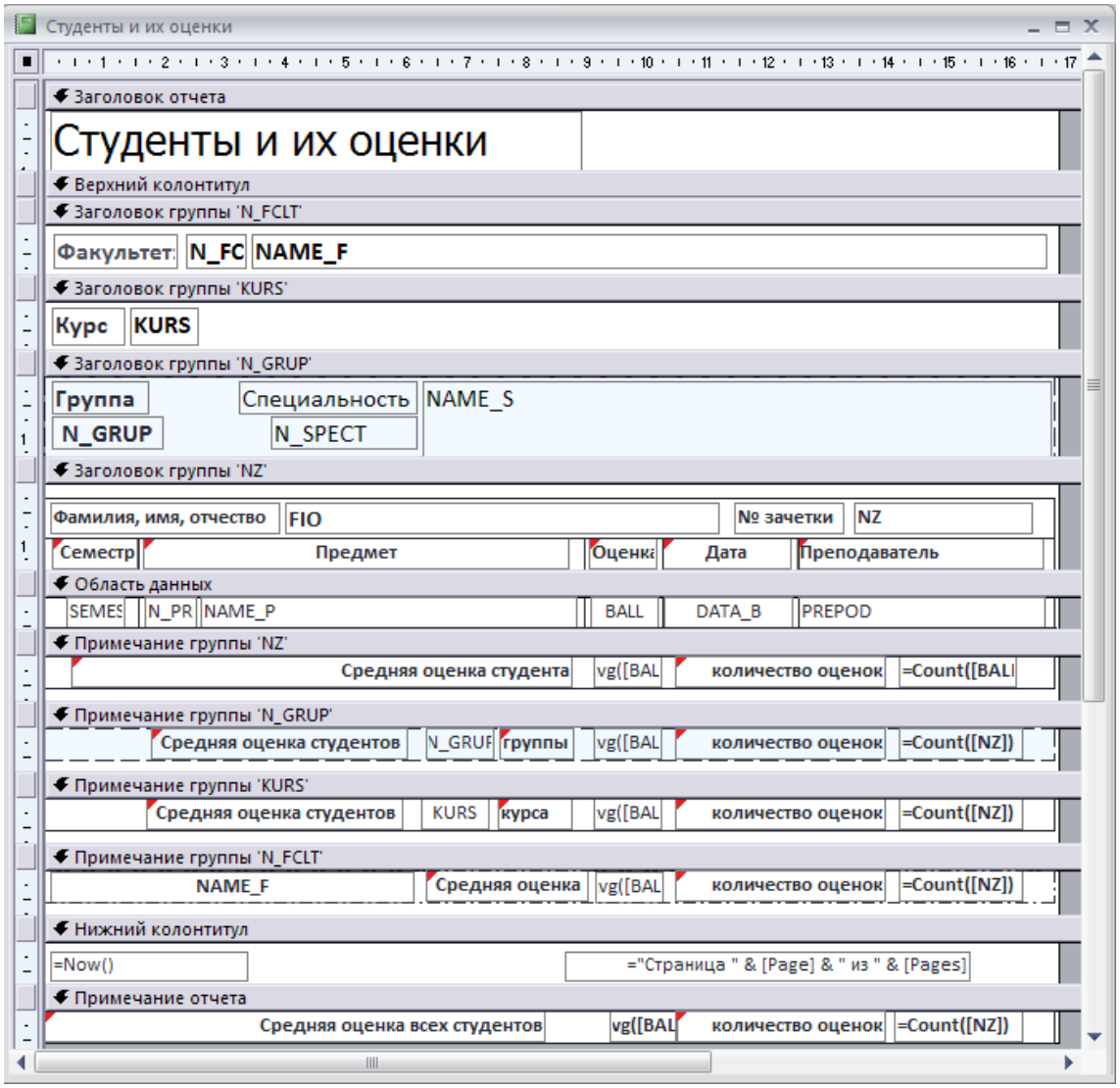


Рисунок 3.84. Отчет после модификации его в Конструкторе

Вид отчета при его печати на бумаге показан на рисунке 3.85.

Студенты и их оценки

Факультет 5 Химико-технологический факультет

Курс 1

Группа	Специальность	Охрана окружающей среды и рациональное использование природных ресурсов
571	280201.65	

Фамилия, имя, отчеств	**Антонова Ирина Сергеевна**		**№ зачетки**	**459871**
Семестр	**Предмет**	**Оценк**	**Дата**	**Преподаватель**
1	3 Физика	5	20.01.2006	Орлова О.П.
1	2 Математика	5	16.01.2006	Веретнов М.Ю.
1	1 Информатика	4	10.01.2006	Орлова О.П.
	Средняя оценка студента	4,67	**количество оценок**	**3**

Фамилия, имя, отчеств	**Практишин Алексей Петрович**		**№ зачетки**	**569783**
Семестр	**Предмет**	**Оценк**	**Дата**	**Преподаватель**
1	2 Математика	5	18.01.2006	Веретнов М.Ю.
1	4 Физическая культура	4	25.01.2006	Шапаров Е.А.
1	3 Физика	5	13.01.2006	Рай А.И.
1	1 Информатика	5	20.01.2006	Орлова О.П.
	Средняя оценка студента	4,75	**количество оценок**	**4**

И Т. Д. ...

Фамилия, имя, отчеств	**Зиновьева Елена Сергеевна**		**№ зачетки**	**555978**
Семестр	**Предмет**	**Оценк**	**Дата**	**Преподаватель**
1	1 Информатика	4	16.01.2006	Орлова О.П.
1	4 Физическая культура	5	18.01.2006	Шапаров Е.А.
1	3 Физика	4	15.01.2006	Рай А.И.
1	5 Иностранный язык	4	25.01.2006	Чернышова О.В.
1	2 Математика	5	12.01.2006	Веретнов М.Ю.
	Средняя оценка студента	4,40	**количество оценок**	**5**

Средняя оценка студентов	573	**группы**	4,43	**количество оценок**	**40**

Средняя оценка студентов	1	**курса**	4,40	**количество оценок**	**73**

Курс 2

Группа	Специальность	Метрология и метрологическое обеспечение
573	200501.65	

И Т. Д. ...

Фамилия, имя, отчеств	**Романов Антон Андреевич**		**№ зачетки**	**597366**
Семестр	**Предмет**	**Оценк**	**Дата**	**Преподаватель**
1	4 Физическая культура	5	16.01.2006	Шапаров Е.А.
1	4 Физическая культура	5	25.01.2006	Шапаров Е.А.
1	3 Физика	5	19.01.2006	Рай А.И.
1	1 Информатика	3	20.01.2006	Орлова О.П.
1	2 Математика	5	22.01.2006	Веретнов М.Ю.
	Средняя оценка студента	4,60	**количество оценок**	**5**

Средняя оценка студентов	674	**группы**	4,40	**количество оценок**	**10**

Средняя оценка студентов	1	**курса**	4,40	**количество оценок**	**15**

Строительный факультет	**Средняя оценка**	4,40	**количество оценок**	**15**

Средняя оценка всех студентов	**4,41**	**количество оценок**	**93**

Рисунок 3.85. Фрагмент распечатки отчета

3.3.4.6 Использование запросов

Ранее отмечалось, что при работе с таблицами, формами и отчетами можно использовать фильтр отбора данных, который задает условия отображения данных в этих объектах (вкладка **Главная** ленты, группа **Сортировка и фильтр**). Там же присутствует кнопка **Расширенный фильтр**, которая открывает окно **Конструктора запросов** для текущей таблицы.

Кроме того для разработки запросов и их сохранения в базе данных в системе **Access** имеются специальные режимы на ленте **Создание** в группе **Другие – Мастер запросов** и **Конструктор запросов**.

Запросы при работе с базами данных в различных СУБД имеют близкий синтаксис, определяемый правилами языка SQL (Structured Query Language) и фактически являются текстовой командой на этом языке. Система **Access** использует команды языка SQL не только для отбора данных (команда **`SELECT...`**), но и для изменения множества данных одной командой **`UPDATE...`**, добавления множества записей одной командой **`INSERT...`** и удаления группы записей одной командой **`DELETE`** …, а также для выполнения вычислений путем создания в запросах вычисляемых полей.

Типы запросов, которые могут быть заданы в системе **Access**:

1. **Выборка** – запрос для отбора данные по заданным сложным условиям из одной или нескольких таблиц баз данных, с показом результатов выполнения запроса в виде таблицы, либо с использованием его для форм и отчетов данных – команда SQL **`SELECT...`**; в **Окне свойств** запроса может быть настроен параметр *Тип набора записей:* динамический набор – после редактирования данных в таблице запроса данные таблиц базы данных обновляются! (с некоторыми ограничениями) и статический набор – только просмотр результатов отбора.
2. **Перекрестный** – формирование двухмерной итоговой таблицы, с группировкой по двум выражениям, одно из которых становится заголовком строки, другое – заголовком столбца.
3. **Создание таблицы** – задается имя новой таблицы, куда заносятся результату выполнения запроса, с сохранением ее в текущей или другой базе данных.
4. Запросы на изменение данных:
   4.1. **обновление** данных – команда занесения общих изменений в группу записей одной или нескольких таблиц (команда SQL **`UPDATE...`**);

4.2. **добавление** данных – команда добавления группы записей из одной или нескольких таблиц в конец одной или нескольких таблиц (команда SQL **`INSERT...`**);

4.3. **удаление** данных – команда удаления группы записей из одной или нескольких таблиц (команда SQL **`DELETE...`**).

Принцип формирования запросов наиболее легко освоить при использовании **Мастера запросов**. Предположим, нам нужно отобрать тех студентов, которые по предмету высшая математика имеют только отличные оценки по результатам второго семестра.

Для создания запроса на вкладке **Создание** нажимаем кнопку **Мастер запросов**, после чего открывается окно с названием ***Новый запрос***, в котором предлагается выбрать тип запроса из списка:

- ***Простой запрос*** (создание запроса на выборку из определенных полей)
- ***Перекрестный запрос*** (создание запроса, выводящего данные в компактном формате, подобном формату электронной таблицы)
- ***Повторяющиеся записи*** (создание запроса на поиск повторяющихся записей в простой таблице или запросе)
- ***Записи без подчиненных*** (создание запроса на поиск записей, которым не соответствует ни одна запись в подчиненной таблице).

Выберем первый тип – простой запрос, после чего откроется окно **Мастера запросов, на первом шаге** которого требуется выбрать таблицы и поля для запроса (рисунок 3.86).

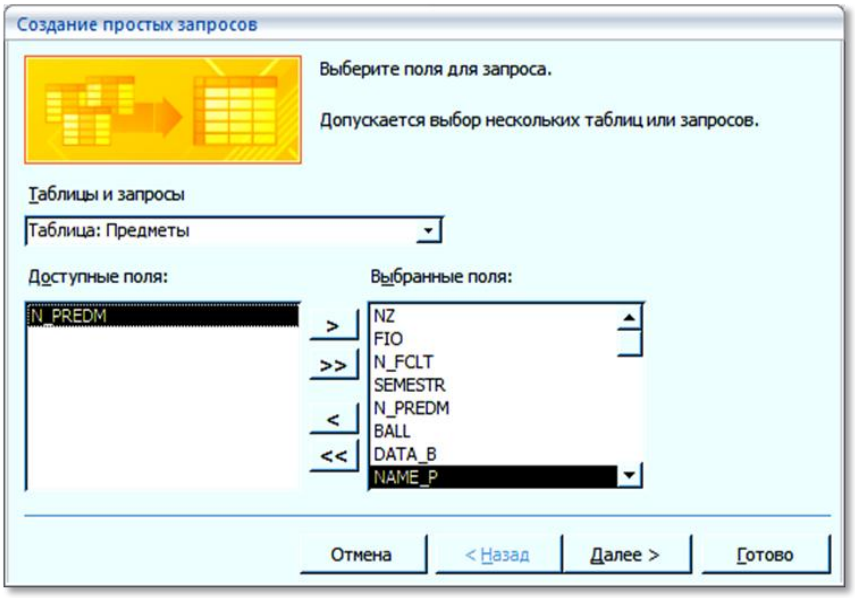


Рисунок 3.86. Выбор полей в **Мастере запросов**

Выбор полей может быть выполнен из нескольких таблиц базы. Для нашего примера выбираем из таблицы **Список** все поля, кроме **DATA_P** и **N_PASP**, из таблицы **Оценки** 4 поля: **SEMESTR, N_PREDM, BALL, DATA_B** из таблицы **Предметы** поле **NAME_P**.

На шаге 2 необходимо выбрать подробный или итоговый отчет, выбираем подробный.

На последнем **шаге 3** задаем название запроса **Математика** и выберем вариант **Изменить макет запроса**, после чего нажимаем кнопку **Готово**. Запрос открывается в **Конструкторе запросов**, его вид показан на рисунок 3.87.

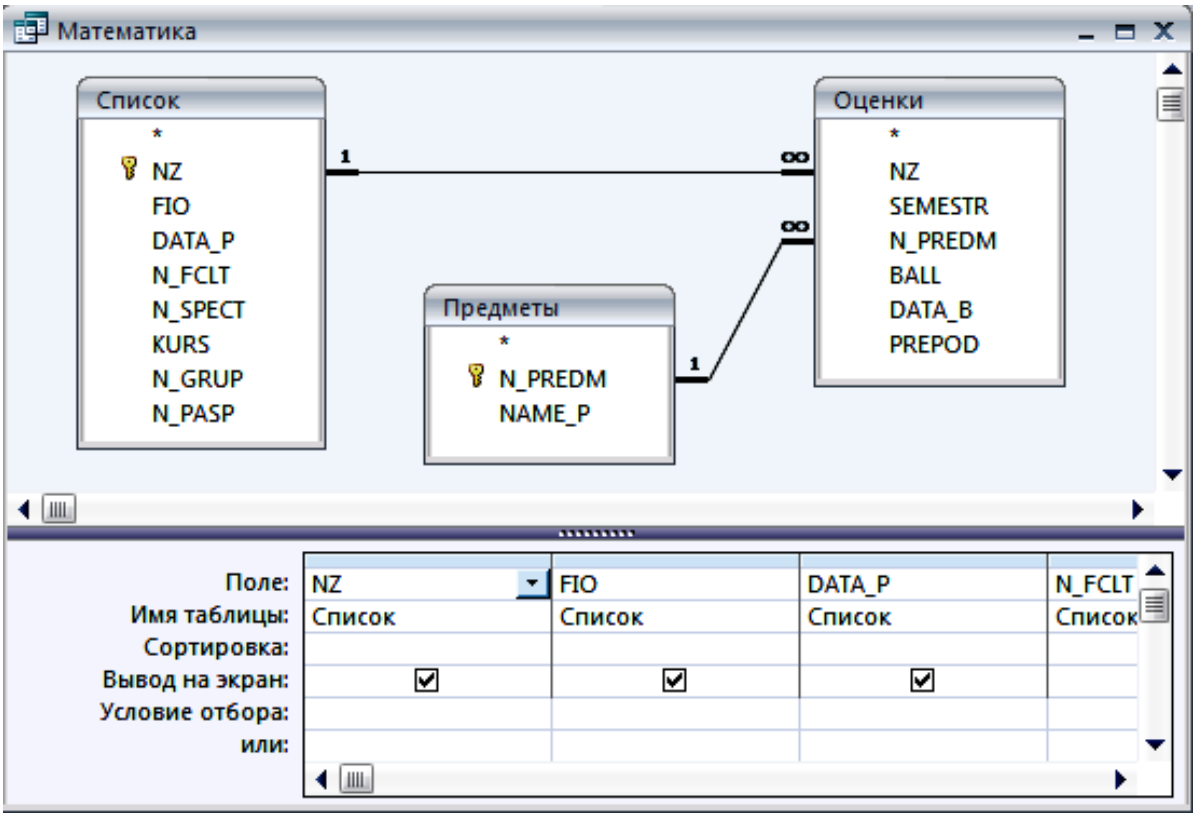


Рисунок 3.87. Конструктором запросов

При работе с Конструктором запросов используется контекстная вкладка **Работа с запросами** – **Конструктор** (см. рисунок 3.88), в которой, в частности, есть кнопка **Итоги**, после нажатия на которую в **Конструкторе запросов** появляется строка **Групповая операция** для задания условий группировки данных и создания **итоговых вычисляемых полей.**

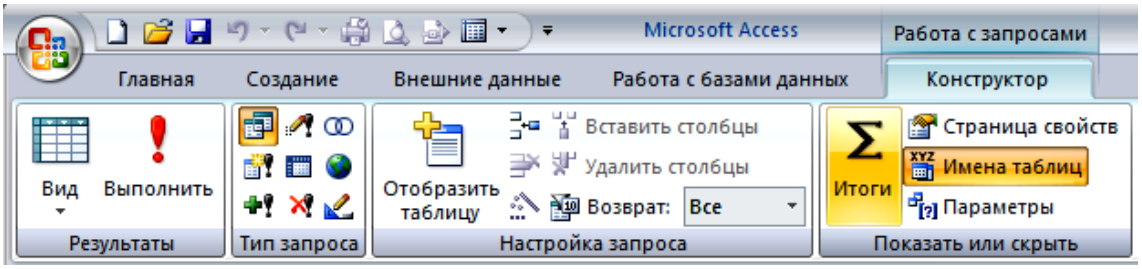


Рис. 3.88. Контекстная вкладка ленты Работа с запросами

В верхней половине конструктора запросов показаны таблицы, используемые для отбора данных и связи между ними, в нижней части – таблица для выбора полей, задания сортировки и условий отбора данных. Двойной щелчок на линии связи между таблицами открывает окно **Параметры объединения** (см. рисунок 3.89), в котором можно задать один из трех вариантов объединения данных таблиц.

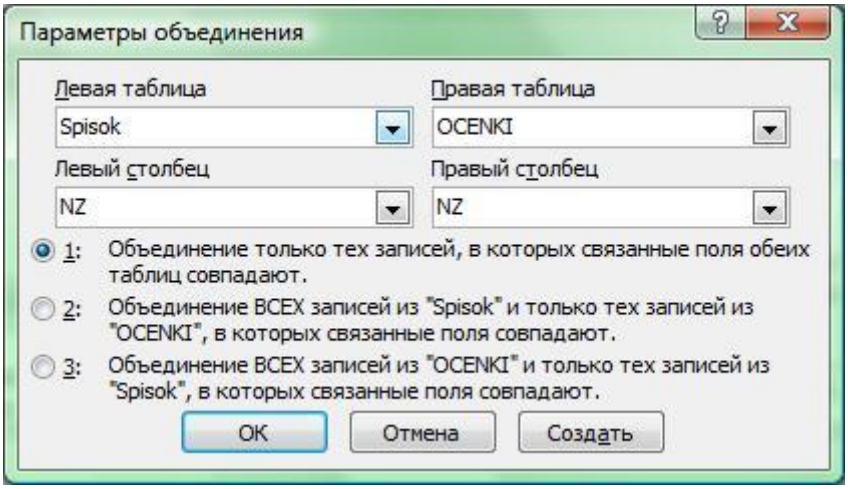


Рис. 3.89. Параметры объединения таблиц в запросе, в синтаксисе SQL: 1– INNER JOIN; 2 – LEFT JOIN; 3 – RIGHT JOIN

Модифицируем запрос для задания условия отбора данных и упорядочения студентов по их фамилии. Для этого в колонке поля **FIO** зададим сортировку по возрастанию, для поля **SEMESTR** зададим условие отбора 1 (первый семестр), для поля **BALL** зададим условие отбора **5** и для поля **NAME_P** зададим условие отбора **Математика**. Если в условии отбора написать текст в квадратных скобках, при выполнении запроса появится окно для ввода этого параметра. Например, если для поля **NAME_P** в условии написать **[Задайте предмет]**, можно будет использовать один и тот же запрос для отбора данных по разным предметам. Можно также убрать галочки у тех полей, которые Вы не хотите показывать на экране.

Сохраним запрос и посмотрим его текст в режиме **SQL** (Structured Query Language). Текст запроса будет выглядеть следующим образом:

```
SELECT Список.NZ, Список.FIO, Список.DATA_P,
     Список.N_FCLT, Список.N_SPECT, Список.KURS,
     Список.N_GRUP, Список.N_PASP, Оценки.SEMESTR,
     Оценки.N_PREDM, Оценки.BALL, Оценки.DATA_B,
     Оценки.PREPOD, Предметы.NAME_P
FROM Список INNER JOIN (Предметы INNER JOIN Оценки ON
     Предметы.N_PREDM = Оценки.N_PREDM) ON Список.NZ =
     Оценки.NZ
  WHERE (((Оценки.SEMESTR)=2) AND
       ((Предметы.NAME_P)="Математика"))
       ORDER BY Список.FIO;
```

Текст запроса состоит из нескольких частей:

1) после ключевого слова **SELECT** следует список полей, которые будут представлены в результате выполнения запроса (например, **Список.NZ** – имя таблицы и её поле, разделенные точкой);

2) после слова **FROM** следует список таблиц, из которых будут отбираться данные, и условия объединения таблиц;

3) после слова **WHERE** следуют условия отбора данных;

4) после слова **FROM** указаны условия упорядочения данных в запросе.

Закроем окно конструктора и выполним запрос командой **Открыть** или двойным щелчком мышью. Результат отбора данных будет показан на экране в виде таблицы.

Результаты выполнения запроса или данные таблиц можно представить в виде диаграмм и графиков. Создадим запрос для таблиц **Список** и **Оценки**, в котором покажем в графическом виде средний балл по студенческим группам по предмету **Математика** (№ предмета = 1). Для группировки данных, как отмечалось выше, на вкладке ленты **Работа с запросами – Конструктор** нажмем на кнопку **Итоги.**

Получим следующий текст SQL-запроса:

```
SELECT Список.N_GRUP, Avg(Оценки.BALL) AS [Avg-BALL]
FROM Список INNER JOIN Оценки ON Список.NZ = Оценки.NZ
GROUP BY Список.N_GRUP, Оценки.N_PREDM
HAVING (((Оценки.N_PREDM)=1));
```

Для представления данных запроса в виде графика выбираем **Вид ► Сводная диаграмма**, после чего открывается окно **Построителя** диаграмм. Для оформления диаграмм используется контекстная вкладка лен-

ты Работа со сводными диаграммами ► Конструктор. На рис 3.90 показана диаграмма для приведенного выше запроса.

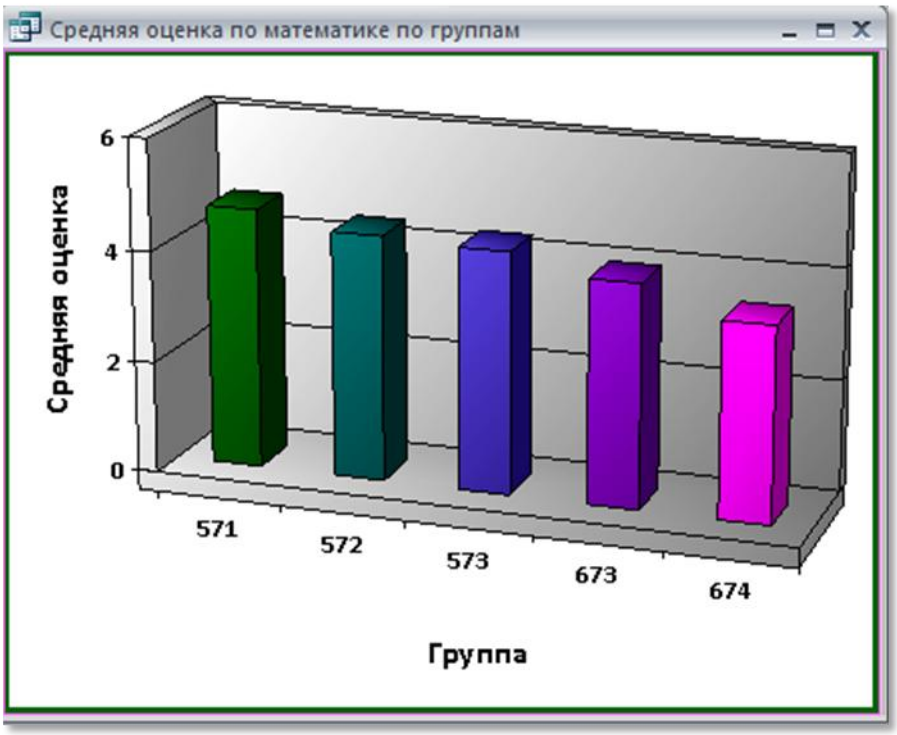


Рис. 3.90. Результаты выполнения запроса с группировкой данных, представленные в виде диаграммы

На рис. 3.91 приведена трехмерная диаграмма для запроса следующего вида:

```
SELECT DISTINCTROW Факультеты.NAME_F,
     Предметы.NAME_P, Avg(Оценки.BALL) AS [Avg-BALL]
FROM Предметы INNER JOIN ((Список INNER JOIN Оценки
     ON Список.NZ = Оценки.NZ) INNER JOIN Факультеты
     ON Список.N_FCLT = Факультеты.N_FCLT) ON
     Предметы.N_PREDM = Оценки.N_PREDM
GROUP BY Факультеты.NAME_F, Предметы.NAME_P;
```

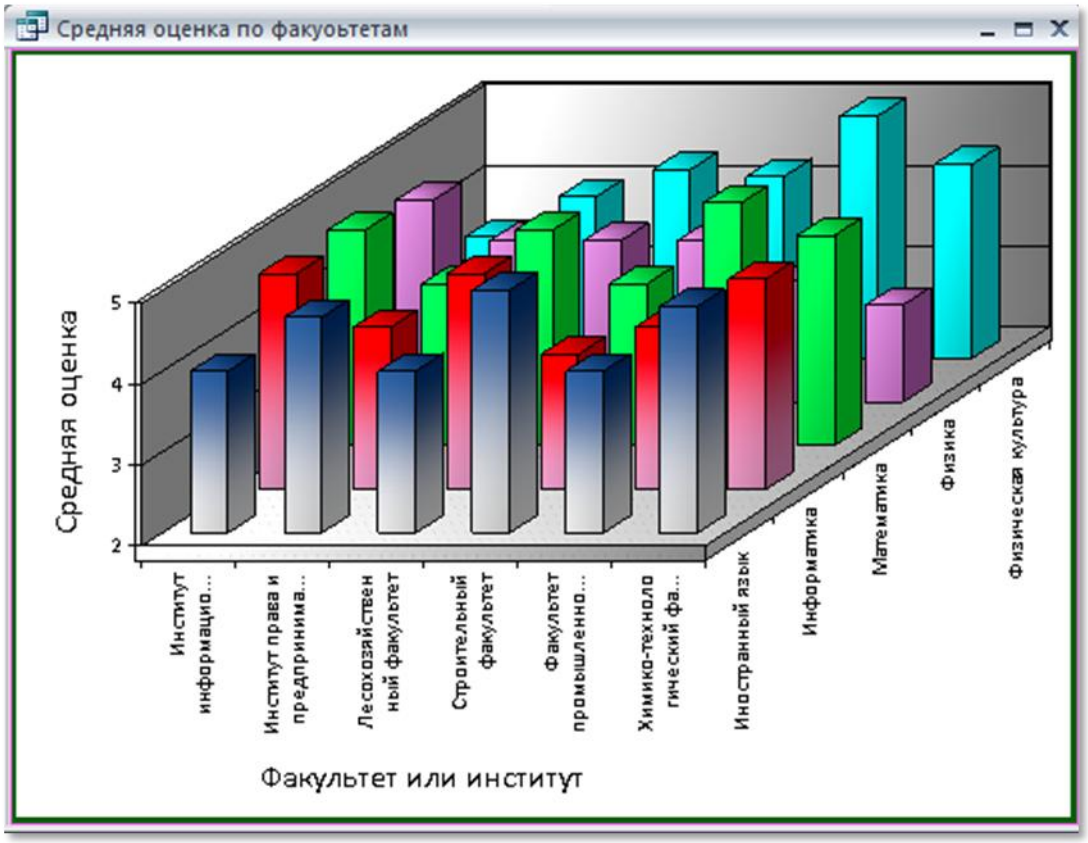


Рис. 3.91. Результаты выполнения запроса с группировкой данных, представленные в виде трехмерной диаграммы

Контрольные вопросы к главе 3

1) Какие виды программного обеспечения существуют?
2) Что такое операционная система? Каковы ее функции?
3) Приведите краткую характеристику первой операционной системы для ПК.
4) Какие два семейства ОС Windows для ПК существовали?
5) Какие файловые системы использует Windows?
6) Какой максимальный размер файла может существовать в FAT32 и в NTFS?
7) В чем отличие интерфейса пользователя в ОС Windows Vista и Windows 7 от предшествующих версий?

8) В чем разница Проводника ОС Windows Vista (или 7) и предшествующих версий?
9) Что такое библиотеки Windows 7?
10) Что такое гаджеты?
11) В чем особенности браузеров Internet Explorer 7 и 8?
12) Что нового в стандартных приложениях Windows 7?
13) Что такое учетные записи и профили пользователей?
14) Как можно организовать общий доступ к информации на ПК в Windows 7?
15) Что такое компьютерный вирус, какие их виды бывают?
16) Какие антивирусные программы имеют наиболее высокий рейтинг, каков принцип их работы?
17) Какие компоненты входят в состав Microsoft Office System?
18) Что такое стиль в системе Microsoft Office Word, какие основные параметры он определяет?
19) Какие основные свойства имеет таблица и ее ячейки в Microsoft Office Word?
20) Какие возможности имеют режимы работы с диаграммами и формулами в Microsoft Office Word?
21) Какие особенности имеет работа с разными видами информации в системе Microsoft Office Excel?
22) Что такое относительная и абсолютная адресация в формулах Microsoft Office Excel?
23) Что такое информационная система, какие этапы имеет ее разработка?
24) Что такое база данных?
25) Что такое триггеры базы данных?
26) Какие обязательные атрибуты должны быть определены для каждого поля таблицы реляционной базы данных?
27) Какие типы могут иметь поля таблицы Microsoft Office Access?
28) Что такое форма в системе Access, какие элементы (объекты) могут на ней присутствовать?
29) Что такое отчет в системе Access, какие элементы (объекты) могут на ней присутствовать, что такое зоны отчета?
30) Что такое запрос при работе с базой данных, какие основные разделы имеет его синтаксис на языке SQL?

Глава 4. Работа с информацией в Интернете

Интернет – глобальная компьютерная сеть, объединяющая компьютеры тысяч региональных сетей и миллионов пользователей всего мира.

С функциональной точки зрения, Интернет – новая коммуникационная платформа, соединяющая людей и организации в самых разных сферах жизни и деятельности.

Интернет (Internet – Interconnected networks, связанные сети) обеспечивает пользователю ПК доступ к информации, размещенной на Интернет-серверах. Интернет – инструмент, позволяющий людям всего мира свободно получать любую необходимую им информацию и свободно общаться друг с другом. По данным на конец 2007 года, в России насчитывалось 35 миллионов активных пользователей Интернета.

Благодаря сети стал доступен (бесплатно или за умеренную плату) огромный объем разнообразной информации. В Интернете представлены практически все стороны жизни мирового сообщества: новости и история, политика и религия, экономика и финансы, производство и торговля, наука и образование, культура и искусство, развлечения и отдых. Пользователи могут получить доступ к книгам электронных библиотек, к Интернет-версиям журналов, к деловой информации корпораций и организаций, к Интернет-магазинам. Для Интернет-общения в реальном времени работают IRC-серверы и программное обеспечение на обычных Интернет-страницах. Поисковые системы за доли секунды находят информацию в Интернете по любому запросу. Интернет-почта за несколько секунд передается на любой компьютер, подключенный к Интернету, даже если он находятся на другом конце света.

4.1 История Интернета

Зарождение Интернета связано с программой Агентства передовых исследовательских проектов в области обороны (DARPA) при Министерстве обороны США. Первая исследовательская программа DARPA, посвященная системе глобальной коммуникации, была начата 4 октября 1962 года. Возглавлявший программу Дж. Ликлайдер (J. C. R. Licklider) опубликовал работу «Galactic Network», в которой предсказывал возможность существования в будущем глобальной компьютерной связи между людь-

ми, имеющими мгновенный доступ к программам и базам данных из любой точки земного шара.

Большое значение для разработки принципов организации компьютерных сетей имела теория коммутации пакетов передачи данных, которую Леонард Клейнрок (Leonard Kleinrock) впервые опубликовал в июле 1964 года. При коммутации пакетов передаваемая информация разбивается на части, к каждой части присоединяется заголовок, который содержит полную адресную информацию. При коммутации же каналов во время передачи информации пара компьютеров соединяется «один к одному».

В 1966 году DARPA приступило к разработке проекта компьютерной сети **ARPANET**. Проект имел сугубо военные цели, отвечающие задачам управления военными и гражданскими объектами в период войны. Принципиально важна была децентрализация, позволяющая сети функционировать даже при уничтожении нескольких узлов сети.

В декабре 1969 года была сдана в эксплуатацию экспериментальная сеть, объединявшая четыре узла со скоростью передачи данных 56 Кбит/с. В 1970-х годах сеть в основном использовалась для пересылки электронной почты, тогда же появились первые списки почтовой рассылки, новостные группы и доски объявлений. Впоследствии к сети **ARPANET** были подключены тысячи серверов и отдельных сетей (главным образом университетских и государственных), что привело к созданию так называемой «**ARPA Internet**» – прародительницы современной сети **Интернет** (см. рисунок 4.1).

В 1983 году сеть **ARPANET** перешла с протокола NCP на TCP/IP (протокол управления передачей/Инернет-протокол), как основу архитектуры. **ARPANET** сыграл важную роль в развитии Интернета. Однако, к тому времени стали возникать проблемы при расширении сетей, наиболее очевидными из которых являлись перегрузки на линиях связи. Помехой развития был также набор правил для пользователей сети, обязательный для всех (так называемая Acceptable Usage Policy – AUP). Согласно этому своду правил, запрещалось использование сети **ARPANET** в коммерческих целях.

В 1985 году Национальный научный фонд (National Science Foundation – **NSF**) приступил к созданию новой сети **NSFNET. NSF** финансировал создание сети, объединившей пять суперкомпьютерных центров и предложил любым региональным и университетским компьютерным центрам, находившимся в пределах физической досягаемости от этой сети подсоединиться к ней. Это был зародыш сети Интернет в нынешнем ее виде. Целый ряд университетов подсоединились к сети **NSF**, чтобы получить

доступ к суперкомпьютерам. Кроме исследовательских задач сеть использовалась для передачи электронной почти, файлов и новостей.

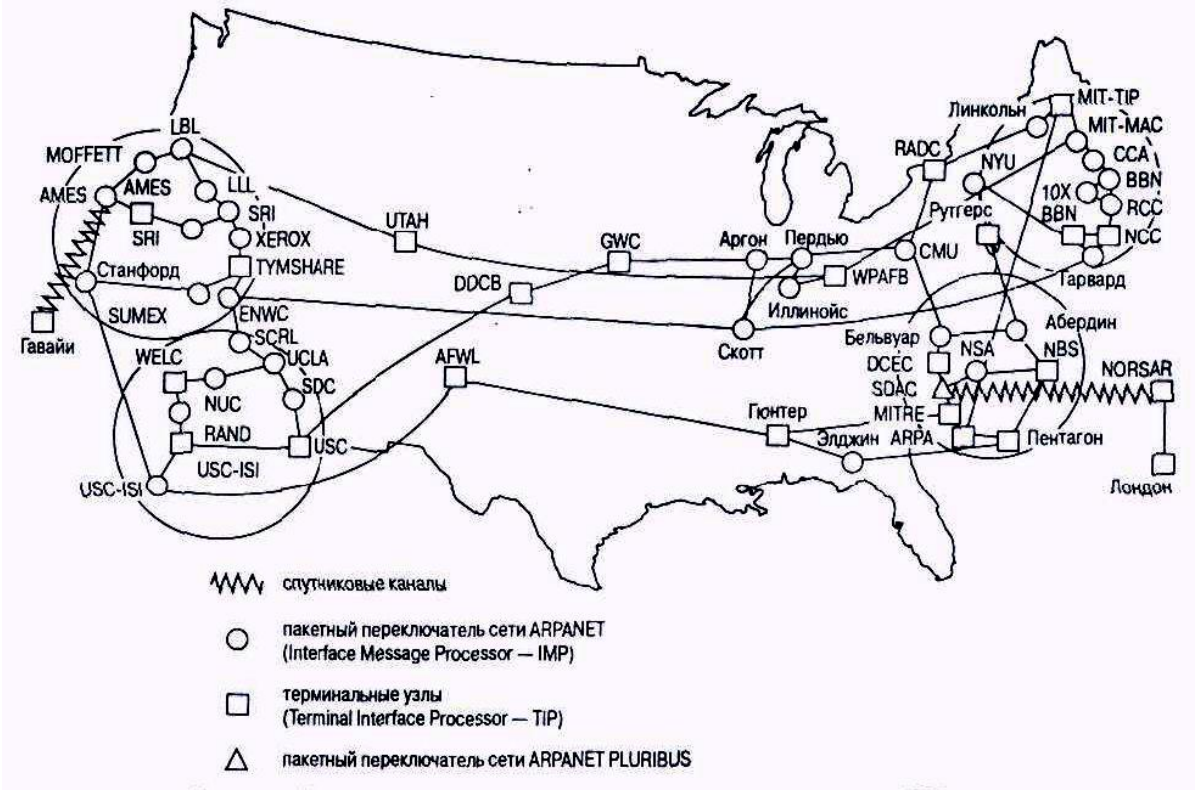


Рисунок 4.1. Структура сети ARPANET (1976 г.)

Большое значение для развития Интернета имела разработка международного стандарта локальных сетей Ethernet. Локальные сети произвели революцию, позволив существенно увеличить производительность труда работников управленческого аппарата, разработать и внедрить автоматизированные системы управления предприятиями. Но они были явно недостаточными для крупных корпораций, имеющих отделения в разных городах и даже странах. Поэтому вполне естественным стал интерес их интерес к подключению своих локальных сетей к общенациональной сети **NSFNET**.

Всемирная паутина – World Wide Web (**WWW**, или просто Web) – появилась после разработки в 1990 году проекта гипертекстовой разметки документов. Автор разработки – Тим Бернер-Ли – программист Европейского центра ядерных исследований (CERN), расположенного в Женеве. В документе 1990 года рассматривались понятия, имеющие принципиальное значение для гипертекстовых документов: общий протокол обмена документами (протокол **HTTP**); общий документальный протокол для постав-

щиков информации и потребителей (язык разметки гипертекстов); поддержка поиска по индексу и возможность просмотра этих документов при помощи браузеров текста и графики.

Слово **Web** также используется для обозначения многих компонентов WWW: Web-браузер, Web-страница, Web-почта, Web-чат, Web-конференция и пр.

В 1993 году **NSF** предложил радикальным образом изменить архитектуру сети:

- создать разветвленную сеть точек доступа к сети (**NAP**), посредством которых провайдеры (поставщики Интернет-услуг) могли бы осуществлять обмен трафиком;
- реализовать проект арбитража маршрутизации в сети (Routing Arbiter project – RA), что облегчило бы согласование правил работы в сети и управление адресами между несколькими провайдерами, подключенными к **NAP**;
- утвердить единого провайдера для службы обеспечения высокоскоростной магистральной сети (very high-speed Backbone Network Service – **vBNS**);
- обеспечить транзит трафика по существующим сетям в целях поддержки межрегиональных соединений путем подключения к провайдерам, которые подключены к **NAP** или напрямую к **NAP**. Любой избранный для этой цели провайдер должен иметь соединения по крайней мере с тремя **NAP.**

В 1994 году **NSF** объявил о строительстве четырех **NAP** в Сан-Франциско, Нью-Йорке, Чикаго и Вашингтоне.

Согласно терминологии **NSF,** точка доступа к сети **NAP** – это высокоскоростной коммутатор или сеть коммутаторов, к которой подключается определенное число маршрутизаторов для обмена трафиком. Любая **NAP** должна работать со скоростью не ниже 100 Мбит/с и иметь возможность повышения пропускной способности по запросу или в зависимости от нагрузки.

30 апреля 1995 года опорная сеть **NSF** была закрыта, а архитектура **NAP** превратилась в Интернет.

4.2 Структура Интернета

Современная сеть Интернет представляет собой объединение операторов Интернет-услуг (провайдеров), у которых в нескольких регионах имеются узлы, называемые точками присутствия (Points Of Presence –

POP). Все подключения клиентов к Интернет-провайдерам осуществляются через серверы доступа или маршрутизаторы, расположенные на **POP** провайдера. Строгая иерархическая структура в Интернете отсутствует.

Решением проблем функционирования Интернета, разработкой новых технологий и стандартов сети, распределением доменных имен выполняет достаточно большой круг общественных и научных организаций.

Ведущей среди них является специальная негосударственная некоммерческая организация, расположенная в США, – **Internet Society** (**ISOC**). Членами **ISOC** могут быть как частные лица, так и организации, но правом голоса на выборах в управляющие органы **ISOC** обладают только частные лица.

В рамках **ISOC** действуют несколько специальных групп:

1) **Совет по архитектуре Интернета** – **IAB** (Internet Architecture Board) – занимается перспективами развития семейства протоколов TCP/IP: рассматривает различные идеи усовершенствования, принципы построения новых систем и пр.;
2) **Рабочая инженерная группа по Интернету** – **IETF** (Internet Engineering Task Force) – занимается оперативным разрешением текущих проблем эксплуатации и развития, а также разработкой новых стандартов и протоколов;
3) **Центр сетевой информации – InterNIC** занимается регулированием IP-адресов, доменных имен, справочных служб и хранением документов по стандартам Интернета.

В 1998 г. функции **InterNIC** были переданы специально созданной организации **ICANN** (Internet Corporation for Assigned Names and Numbers), которая занимается вопросами стратегического регулирования назначения IP-адресов и доменных имен. Международной организацией оперативного управления распределением IP-адресов и доменных имен является **IANA** – Internet Assigned Numbers Authority (Уполномоченная организация по распределению нумерации в сети Интернет).

IANA определила Региональных регистраторов сети Интернет (Regional Internet Registry – **RIR**), обеспечивающих распределение пространства IP-адресов и назначения номеров автономным системам в сети Интернет по всему земному шару:

- Американский реестр адресов сети Интернет (American Registry for the Internet Numbers – **ARIN**);
- Европейский сетевой координационный центр (Reseaux IP Europeens Network Coordination Center – **RIPE NCC**);
- Азиатско-Тихоокеанский сетевой информационный центр (**APNIC**);

- Региональный центр Латинской Америки и Карибских островов (Regional Latin-American and Caribbean IP Address Registry – **LACNIC**).

Общее администрирование и техническое сопровождение российского национального домена (**.RU**) первоначально (в 1993 году) было поручено нейтральной организации – Российскому научно-исследовательскому Институту развития общественных сетей (**РосНИИРОС**).

7 апреля 1994 года **IANA** занесла в международную базу данных национальных доменов верхнего уровня российский национальный домен RU.

В 2001 году был учрежден российский Координационный центр национального домена сети Интернет. В 2002 году Координационному центру переданы полномочия по выработке правил регистрации доменных имен в домене ***RU***, аккредитации регистраторов и разработке перспективных проектов, связанных с развитием российского национального домена. В целях сохранения стабильности функционирования домена ***RU*** на **РосНИИРОС** возложены функции Технического центра домена ***RU***.

Список аккредитованных регистраторов домена ***RU*** можно найти на сайте Координационного центра (*www.cctld.ru/ru*), среди них:

- Центр регистрации доменов RU-center – *www.nic.ru;*
- Регистратор доменных имен РЕГ.РУ – *www.reg.ru;*
- Регистратор доменов – *www.mastername.ru*;
- НЕТ - *www.zastolbi.ru* и прочие.

Структура российского сегмента Интернета начала формироваться в 1991 – 92 годах (EUnet/Relcom, Sovam Teleport, Гласнет, FREEnet). В конце 90-х Ростелеком, почувствовав коммерческий интерес к Интернет-рынку, создал первую магистральную IP сеть РФ. До 2003 г. Ростелеком контролировал магистрального оператора Интернет-доступа РТКомм.ру. В 2000-2001 году на рынок магистрального провайдинга вышел ТрансТелеКом (**ТТК**) с молниеносно построенной более современной, более широкополосной и более разветвленной сетью. Магистральных операторов стало два (см. рисунок 4.2).

Операторы Интернет-доступа по функционально-географическому признаку разделяются на следующие типы:

- магистральные операторы (Internet Backbone Provider – **IBP**) – операторы межрегиональных IP-сетей, обеспечивающие перенос Интернет-трафика между регионами страны и за рубеж;

- конечные провайдеры (Интернет-сервис-провайдеры – **ISP**), обеспечивающие предоставление Интернет-услуг конечным пользователям:
  - региональные **ISP –** действующие на региональном уровне;
  - локальные **ISP** – действующие на локальном уровне.

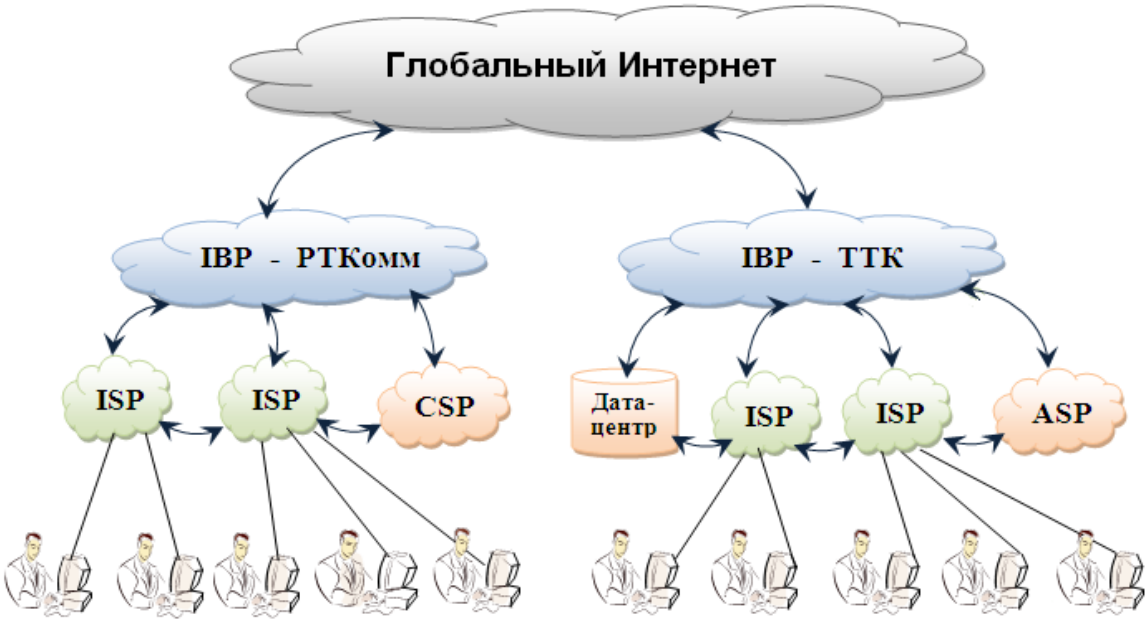


Рисунок 4.2. Структура интернет-провайдинга в России

Обычному пользователю для получения доступа в сеть необходимо воспользоваться услугами локального Интернет-провайдера.

Кроме операторов Интернет-доступа существуют также операторы контент-ресурсов – компании, деятельность которых направлена на информационное наполнение сети Интернет и предоставление конечным пользователям услуг информационного характера.

Различают следующие типы операторов контент-ресурсов:

- операторы дата-центров, предоставляющие возможность своим пользователям разместить их информацию на серверах и сделать ее доступной в сети **И**нтернет;
- провайдеры приложений (Application Service Provider – **ASP**) – компании, предоставляющие пользователям программные приложения, использующие Интернет-технологии и доступные из сети Интернет;
- контент-провайдеры (Content Service Provider – **CSP**) – компании, предоставляющие пользователям Интернета ту или иную информацию или службу (контент).

Классификация Интернет-операторов показана на рисунке 4.3.

Рисунок 4.3. Классификация операторов Интернет-услуг

По данным фонда «Общественное мнение» (*www.fom.ru*) в России осенью 2009 года насчитывалось 42 млн. пользователей Интернета (36% взрослого населения), наибольший процент пользователей в Москве (60%) и Северо-западном регионе (48%), более 27% пользователей в других регионах (см. рисунок 4.4).

Одной из проблем российского Интернета на начальной стадии его развития было практическое отсутствие межсетевого обмена IP-трафиком внутри страны. Межсетевые связи реализовывались в глобальном Интернете. Вследствие этого пользователи, имеющие подключение через одну национальную сеть, добирались до российского сервера, стоящего в другой сети (нередко в том же городе), проходя через океаны и геостационарные космические аппараты и перегружая не слишком широкие внешние магистральные каналы.

В середине 1996 года было подписано историческое соглашение по созданию точки взаимного обмена IP-трафиком (Internet eXchange) в Москве (M9-IX на автоматической междугородней телефонной станции АМТС-9). Участниками соглашения стали АО "Релком" (сеть EUnet/Relcom), компания «Демос» (сеть Demos/Internet), МГУ (MSUnet),

НИИЯФ МГУ (сеть RUHEP/Radio-MSU), корпорация «УНИКОР» (сеть FREEnet), ассоциация RELARN (сеть RELARN-IP), АО «РОСПРИНТ».

Рисунок 4.4. Доля пользователей Интернетом

Впоследствии к соглашению присоединились Институт космических исследований (сеть RSSI), компании «Совам Телепорт» (сеть Sovam), «Глас-Интернет» (сеть GlasNet), Институт «Открытое общество» (сеть IIP), Вузтелекомцентр (сеть RUNNet), ОАО «Российская телекоммуникационная сеть» (сеть Rosnet). Обмен IP-трафиком между участниками осуществляется на основе отдельных парных договоров. Аналогичная по своим задачам точка была создана в Санкт-Петербурге с участием сетей EUnet/Relcom, RUNNet, RELARN-IP и региональных сетей RUSnet N/W и "РОКСОН С3". Более подробную информацию о работе этих точек взаимного обмена можно получить на сервере Российского НИИ развития общественных сетей (*www.ripn.net*).

Организация обмена между операторами Интернет-доступа Интернет-трафиком своих клиентов (пиринг) разгрузила внешние каналы связи, но роль последних весьма велика, т. к. большую часть информационных ресурсов пользователи пока потребляют из зарубежных сетей. В конце 1995 года суммарная пропускная способность каналов связи российских сетей с зарубежными провайдерами составляла чуть более 2 Мбит/с, а через год превысила 15 Мбит/с и ежемесячно продолжает возрастать как за счет ввода новых каналов, так и за счет повышения производительности существующих.

Крупнейшими магистральными операторами России остаются РТКомм.ру и ТрансТелеКом (ТТК).

ОАО «РТКомм.ру» имеет 45-процентную долю на рынке операторского доступа к Интернету. Сеть РТКомм, ориентированная на передачу IP-трафика, была построена в течение 2002 г. на основе цифровых магистральных каналов связи. В настоящее время сеть компании насчитывает более 160 магистральных и региональных узлов (из них 128 поддерживают технологию MPLS) и дата-центры в Москве, Новосибирске и Ростове-на-Дону. Таким образом, компания обеспечила присутствие собственной сети в большинстве регионов России. В течение 2007 года планируется расширение сети.

РТКомм – это самая разветвленная в России IP-сеть с основными магистральными каналами с цифровой скоростью STM-1 (155 Мбит/с) и выходом на зарубежные сети по кольцу STM-16 (2,5 Гбит/с) по маршруту Москва – Лондон – Франкфурт – Санкт-Петербург – Москва. Общая пропускная способность международной составляющей сети – 12 Гбит. Введены в эксплуатацию узлы в Риге, Лондоне, Франкфурте и Амстердаме (см. рисунок 4.5).

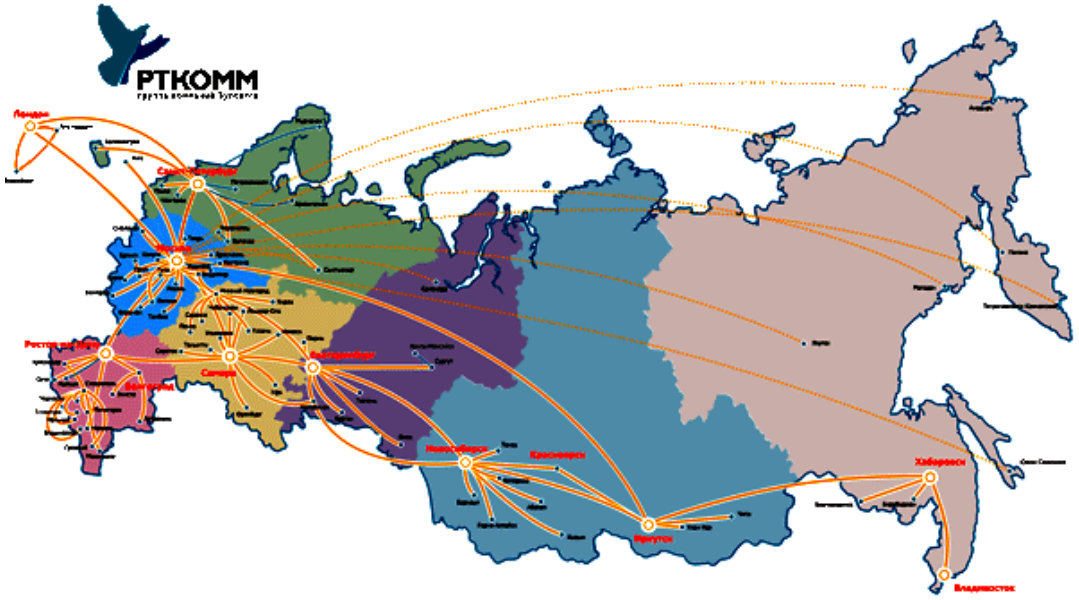


Рисунок 4.5. Сеть ОАО «РТКомм.ру»

ЗАО «ТрансТелеКом» первоначально занималась строительством и эксплуатацией высокоскоростной телекоммуникационной сети вдоль российских железных дорог. Сейчас услугами ТТК охвачена территория, на которой проживает 90% населения России, ТТК имеет 50 000 км магистрально-цифровой сети связи. Сеть проходит через 11 часовых поясов, имеет более 950 точек доступа в 71 из 89 регионов России и выходит на гра-

ницы с Финляндией, Польшей, Монголией, Китаем, странами СНГ и Балтии (см. рисунок 4.6). По утверждению главы компании Сергея Липатова, ТТК занимает 44% рынка выделенных каналов и 45% рынка транзита Интернет-трафика.

Рисунок 4.6. Сеть ЗАО «ТрансТелеКом»

Другие сети национального масштаба, представляющие российскую часть Интернета:

- **Роснет** (*www.rosnet.ru*) – владелец сети АО «Российская телекоммуникационная сеть», имеет высокоскоростное IP-подключение к магистральной части американского Интернета;
- **EUnet/Relcom** (*www.relcom.ru*) – базовые узлы в Москве и Санкт-Петербурге, имеет более 100 узлов в России и СНГ и два выхода в европейскую сеть Eunet;
- **Demos/Internet** (*www.demos.su*) – ядром сети являются несколько узлов в Москве, имеет магистральный выход в сеть MCI (США);
- **Sovam Teleport** (*www.sovam.com*) – входит в группу Global Telesystems Group (GTS), имеет отделения во многих городах, услуги Интернета предоставляются в рамках проекта «Россия-Он-Лайн» (*www.online.ru*);
- **Гласнет** (*www.glasnet.ru*) – имеет центральный узел в Москве, канал в сеть CRL (США);
- **Global One Russia** (*www.global-one.ru*) – представляет в России крупнейшую мировую телекоммуникационную сеть Global One, образован-

ную компаниями Deutsche Telekom, France Telecom и Sprint. В России существует около 200 узлов доступа к сети Global One;

- **RUNNet** (*www.runnet.ru*) – федеральная университетская сеть России, магистральная часть которой включает узлы более чем в 20 городах России, спутниковые и наземные цифровые каналы и выход в глобальный Интернет через скандинавскую сеть NORDUnet;
- **RUHEP/Radio-MSU** (*www.radio-msu.net*) – имеет узлы в ряде городов России и СНГ, основана на системе спутниковых каналов, в которой ключевую роль играют узлы в Москве и Гамбурге;
- **RSSI** (*www.rssi.ru*) – Russian Space Science Internet (Российская космическая научная сеть Интернет) – объединяет научно-исследовательские центры и институты, медицинские учреждения, учебные заведения и имеет выход в глобальный Интернет через американскую сеть NASA Internet;
- **FREEnet** (*www.free.net*) – старейшая российская научно-образовательная сеть, имеющая более пятнадцати региональных отделений, через которые работает большое число вузов и научных организаций РАН;
- **RELARN-IP** (*www.ripn.net/relarn/*) – сеть, поддерживаемая Ассоциацией научных и учебных организаций – пользователей компьютерных сетей передачи данных (ассоциация RELARN);
- **MSUnet** (*www.msu.ru*) – сеть Московского университета, имеет выход в европейскую сеть Ebone и предоставляет свои услуги многим организациям, а также региональным сетям.

Технологии обеспечения доступа в сеть Интернет можно разделить на три категории, в зависимости от того, какой канал или среда передачи данных используется. К ним относятся:

- телефонный провод;
- сетевой кабель (технологии Ethernet);
- оптико-волоконный кабель;
- беспроводные системы (сотовая, радиорелейная или спутниковая связь) .

Все компьютеры, участвующие в передаче данных в сети (в настоящее время и в локальной при использовании в качестве сетевой операционной системы Windows), используют протокол **TCP/IP** (читается «ти-си-пи/ай-пи»).

На самом деле протокол **TCP/IP** – это два разных протокола, определяющих различные аспекты передачи данных в сети:

– **TCP** (Transmission Control Protocol) – протокол управления передачей данных, использующий автоматическую повторную передачу пакетов, содержащих ошибки; этот протокол отвечает за разбиение передаваемой информации на пакеты и правильное восстановление информации из пакетов получателя;
– **IP** (Internet Protocol) – протокол межсетевого взаимодействия, отвечающий за адресацию и позволяющий пакету на пути к конечному пункту назначения проходить по нескольким сетям.

Схема передачи информации по протоколу **TCP/IP** такова: протокол **TCP** разбивает информацию на пакеты и нумерует все пакеты; далее с помощью протокола **IP** все пакеты передаются получателю, где с помощью протокола **TCP** проверяется, все ли пакеты получены; после получения всех пакетов протокол **TCP** располагает их в нужном порядке и собирает в единое целое.

Общий список используемых в Интернете протоколов достаточно большой:

1) *на прикладном уровне используются протоколы*:

- *DNS* · *HTTPS* · *POP3* · *Telnet*
- *FTP* · *IMAP* · *SMTP* · *XMPP*
- *HTTP* · *LDAP* · *SSH* · *SNMP*

2) *на сеансовом уровне/уровне представления используются:*

- *SSL* · *TLS*

3) *на транспортном уровне*:

- *TCP* · *UDP*

4) *на сетевом уровне*:

- *BGP* · *IGMP* · *OSPF* · *EIGRP*
- *ICMP* · *IP* · *RIP* · *IS-IS*

5) *на канальном уровне*:

- *Ethernet* · *SLIP* · *Frame relay* · *PPP*
- *HDLC*

4.3 Адресация в Интернете

Каждый компьютер, подключенный к Интернету имеет два равноценных уникальных адреса: цифровой **IP**-адрес и символический доменный адрес **DNS** (Domain Name System, доменная система имен). В настоящее время существует и используется две версии цифровой адресации: **Ipv4** и **Ipv6** (поддерживается в Windows Vista).

IP-адрес представляет собой уникальную четырехоктетную (32-битовую) величину, выраженную в десятичных числах, разделенных точками, в форме W.X.Y.Z, где точки используются для разделения октетов (например, 10.0.0.1). Поле адреса размером 32 бита состоит из двух частей: адрес сети или связи – сетевая часть адреса, и адрес компьютера в сети (хоста) – идентифицирующий хост в сетевом сегменте.

Для **Ipv4,** кроме обычного распределения адресов по классам, используются также методы использования масок переменной длины (**VLSM** – Variable-Length Subnet Mask) и бесклассовая междоменная маршрутизация (**CIDR** – Classless Interdomain Routing).

Традиционное разграничение сетей по количеству хостов в них осуществляется на основе так называемых классов IP-адресов. Сегодня существует 5 классов IP-адресов (три из которых используются для уникальной адресации сетей и хостов): A, B, C, D и E. Все они представлены в таблице 4.1.

Таблица 4.1. Классы IP-адресов и их функции

Класс	Диапазон адресов	Старшие биты	Биты для обозначения подсети	Биты для обозначения хоста	Функция
A	0.0.0.0 – 127.255.255.255	0	7	24	Уникальные адреса
B	128.0.0.0 – 191.255.255.255	10	14	16	Уникальные адреса
C	192.0.0.0 – 223.255.255.255	110	21	8	Уникальные адреса
D	224.0.0.0 – 239.255.255.255	1110			Многоадресное обращение
E	240.0.0.0 – 255.255.255.255	11110			Зарезервировано

Только адреса классов A, B и C могут использоваться как уникальные. Адреса класса D применяются для обращения к набору узлов, а адреса класса E зарезервированы для исследовательских целей и в настоящее время не используются. Несколько адресов во всех классах зарезервированы для специальных целей. Некоторые из них представлены в табл. 4.2.

Вся информация по получению и распределению адресов находится на Web-сервере ARIN (www.arin.net).

Таблица 4.2. IP-адреса, зарезервированные для специальных целей

Диапазон адресов	Назначение
0.0.0.0	Неизвестная сеть (обычно представляет сеть по умолчанию)
10.0.0.0 – 10.255.255.255	Зарезервировано для частных сетей
127.0.0.1 – 127.255.255.255	Зарезервировано для локальных адресов типа «петля»
172.16.0.0 – 172.31.255.255	Зарезервировано для частных сетей
192.168.0.0 – 192.168.255.255	Зарезервировано для частных сетей
192.168.255.255 – 255.255.255.255	Широковещательный запрос

Географически адресное пространство распределено между четырьмя основными регионами (см. таблицу 4.3):

1. Европой, 2.Северной Америкой и Северной Африкой,
2. Тихоокеанским регионом, 3. Южной и Центральной Америкой.

Таблица 4.3. Распределение адресного пространства по регионам

Адресное пространство	Кому выдано	Когда выдано
61.0.0.0 – 61.255.255.255	Тихоокеанский регион	апрель 1997
62.0.0.0 – 62.255.255.255	Европа	апрель 1997
193.0.0.0 – 195.255.255.255	Европа	май 1993
199.0.0.0 – 199.255.255.255	Северная Америка	май 1993
200.0.0.0 – 200.255.255.255	Центральная и Южная Америка	май 1993
201.0.0.0 – 201.255.255.255	Центральная и Южная Африка	май 1993
202.0.0.0 – 203.255.255.255	Тихоокеанский регион	май 1993
204.0.0.0 – 205.255.255.255	Северная Америка	май 1993
206.0.0.0 – 206.255.255.255	Северная Америка	март 1994
207.0.0.0 – 207.255.255.255	Северная Америка	апрель 1995
208.0.0.0 – 208.255.255.255	Северная Америка	ноябрь 1995
209.0.0.0 – 209.255.255.255	Северная Америка	апрель 1996
210.0.0.0 – 210.255.255.255	Тихоокеанский регион	июнь 1996
211.0.0.0 – 211.255.255.255	Тихоокеанский регион	июнь 1996
212.0.0.0 – 212.255.255.255	Европа	октябрь 1997
213.0.0.0 – 213.255.255.255	Европа	март 1999
216.0.0.0 – 217.255.255.255	Северная Америка	апрель 1998

С помощью специального механизма (маскирования) любая сеть может быть представлена набором более мелких сетей (подсетей).

Адреса класса A, B и C имеют стандартные маски, которые задаются с учетом максимального количества сетей и узлов в них для каждого класса. Для стандартных классов сетей маски имеют следующие значения:

- 255.0.0.0 – для сети класса A (адрес сети – первый октет, второй, третий и четвертый – для адресации подсетей и компьютеров в подсетях);
- 255.255.0.0 – для сети класса B (адрес сети – первый и второй октет);
- 255.255.255.0 – для сети класса C (адрес сети – первый, второй и третий октет, максимум компьютеров для одной подсети – 254).

Для IP-адресов класса B первые два октета являются номером сети. Оставшаяся часть IP-адреса может использоваться как угодно. Например, администратор сети может задать, что третий октет будет определять номер подсети, а четвертый октет – номер узла (компьютера) в ней. Конфигурация подсетей описывается в файлах настройки Интернет-сервера организации. Это описание является локальным для организации и не видно вне ее. Все машины вне организации видят одну большую IP-сеть. Изменения, происходящие в IP-сети организации, не видны вне ее. Администратор может добавить новую подсеть, новые узды и т. д.

Если в IP-адресах класса B третий октет использовать для задания номера подсети, сеть может иметь 256 подсетей, в каждой из которых может быть до 254 узлов. Маска подсети в такой системе равна 255.255.255.0. Но, если в сети нужно иметь больше подсетей, а в каждой подсети не будет при этом более 60 узлов, то можно использовать маску 255.255.255.192. Это позволяет иметь 1024 подсети и до 62 узлов в каждой.

Бесклассовая междоменная маршрутизация (Classless InterDomain Routing, **CIDR**) была новым шагом на пути эволюции IP-адресов классов A, B и C. Бесклассовая адресация основывается на переменной длине маски подсети (Variable Length Subnet Mask – **VLSM**), При работе с **CIDR** в IP-адрес сети через косую черту вносится число, показывающее количество бит соответствующих маске подсети, связанной с данным сетевым адресом. Например, сеть с адресом 198.24.16.0 и префиксом /21, которая записывается как 198.24.16.0/21. Здесь префикс /21 указывает на то, что в маске используются старшие 21 битов. Таким образом, это соответствует IP-сети 198.24.16.0 с маской 255.255. 248.0.

Пример расшифровки маски сети в бесклассовой нотации:

биты маски подсети	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
октеты маски подсети	255				255				248				0																		

Протокол IPv6. 14 июля 1999 года Управление Интернет по выделенным номерам (**IANA**) объявило о введении версии **IPv6** во всемирном масштабе. 4 февраля 2008 г. **ICANN** начала добавлять в шесть из трина-

дцати корневых DNS-серверов записи, содержащие адреса в формате протокола **IPv6.**

Протокол **IPv6** выделяет на адрес 128 бит вместо 32 для IPv4. Новая версия IP-протокола **IPv6** даст Интернету 1 миллиард адресов в квадрате, которых должно хватить на много лет.

Изменения **IPv6** по отношению к IPv4 можно поделить на следующие группы:

- ***Расширение адресации.***
  В IPv6 длина адреса расширена до 128 бит, что позволяет обеспечить больше уровней иерархии адресации, увеличить число адресуемых узлов, упростить автоконфигурацию. В адресное поле введено субполе «scope» (группа адресов). Определены новые типы адресов: ***уникастный*** (Unicast) – одиночный адрес; ***эникастный*** (Anycast) – пакет доставляется одному из интерфейсов, указанному в адресе (ближайший, в соответствии с мерой, определенной протоколом маршрутизации) и ***мультикастный*** (Multicast) **–** пакет, посланный по мультикастинг-адресу, доставляется всем адресатам, определенным этим адресом.
- ***Спецификация формата заголовков.***
  Некоторые поля заголовка IPv4 отбрасываются или делаются опционными, уменьшая издержки, связанные с обработкой заголовков пакетов с тем, чтобы уменьшить влияние расширения длины адресов в IPv6.
- ***Улучшенная поддержка расширений и опций.***
  Изменение кодирования опций IP-заголовков позволяет облегчить переадресацию пакетов, ослабляет ограничения на длину опций, и делает более доступным введение дополнительных опций в будущем.
- ***Возможность пометки потоков данных.***
  Введена возможность помечать пакеты, принадлежащие определенным транспортным потокам, для которых отправитель запросил определенную процедуру обработки, например, нестандартный тип TOS (вид услуг) или обработка данных в реальном масштабе времени.
- ***Идентификация и защита частных обменов.***
  В IPv6 введена спецификация идентификации сетевых объектов или субъектов, для обеспечения целостности данных и при желании защиты частной информации.

Формат и семантика адресов **IPv6** описаны в документе RFC-1884. Версия ICMP **IPv6** рассмотрена в RFC-1885.

Существует три стандартные формы для представления **IPv6** адресов в виде текстовых строк:

1. Основная форма имеет вид x:x:x:x:x:x:x:x, где 'x' шестнадцатеричные 16-битовые числа. Например:

   fedc:ba98:7654:3210:FEDC:BA98:7654:3210

   1080:0:0:0:8:800:200C:417A

   Начальные нули в каждом из конкретных полей не пишутся, но в каждом поле должна быть, по крайней мере, одна цифра (за исключением случая, описанного в пункте 2).

2. Из-за метода записи некоторых типов **IPv6** адресов, они часто содержат длинные последовательности нулевых бит. Для того чтобы сделать запись адресов, содержащих нулевые биты, более удобной, имеется специальный синтаксис для удаления лишних нулей. Использование записи "::" указывает на наличие групп из 16 нулевых бит. Комбинация "::" может появляться только при записи адреса. Последовательность "::" может также использоваться для удаления из записи начальных или завершающих нулей в адресе.
   Например, адреса:

1080:0:0:0:8:800:200c:417a	уникаст-адрес
ff01:0:0:0:0:0:0:43	мультикаст адрес
0:0:0:0:0:0:0:1	адрес обратной связи

   могут быть представлены при использовании "::" в виде:

1080::8:800:200c:417a	уникаст-адрес
ff01::43	мультикаст адрес
::1	адрес обратной связи

3. Альтернативной формой записи, которая более удобна при работе с **IPv4** и **IPv6**, является x:x:x:x:x:x:d.d.d.d, где 'x' шестнадцатеричные 16-битовые коды адреса, а 'd' десятичные 8-битовые, составляющие младшую часть адреса (стандартное IPv4 представление).

   Например:
   0:0:0:0:0:0:13.1.68.3
   0:0:0:0:0:FFFF:129.144.52.38
   или в сжатом виде:
   ::13.1.68.3
   ::FFFF:129.144.52.38

Доменный адрес DNS, в отличие от цифрового, является текстовым и легче запоминается человеком. Пример доменного адреса: maestro.it3207.agtu.ru. Здесь maestro – имя реального компьютера в домене it3207 (третьего уровня), принадлежащем домену более высокого второго уровня agtu, ru – обозначение группы русскоязычных доменов первого уровня.

Доменов первого (высшего) уровня немного – около 250. Большая часть из них – так называемые географические домены. Например,.ru (Russia, Россия),.jp (Japan, Япония),.us (United States, США),.uk (United Kingdom, Великобритания),.ca (Canada, Канада),.au (Australia, Австралия) и т. д.. Негеографические домены верхнего уровня –.com (для коммерческих компаний),.net (для сетевых ресурсов),.edu (образовательные учреждения),.mil (военные организации),.org (некоммерческие организации),.gov (правительственные ведомства),.int (интернациональные корпорации) и др.

В процессе передачи данных доменный адрес преобразуются в **IP**-адрес. В сети существуют специальные DNS-серверы, отвечающие за адресацию некоторой части сети и сохраняющие информацию о соответствии **IP**-адресов и **DNS**-имен.

Для приобретения в собственность имени домена второго уровня (например www.agtu.ru), необходимо обратиться к одному из регистраторов доменных имен. Аккредитацией регистраторов в домене.ru занимается Координационный центр национального домена сети Интернет, как указывалось раньше (см. раздел 4.2). За зарегистрированные имена необходимо ежегодную плату (например, в Центре регистрации доменов RU-Center около 600 руб. в домене.ru, 450 руб. в доменах NET, COM, ORG, BIZ, INFO).

Для поиска нужной информации в сети используется адрес ресурса (**URL**-адрес, англ. Uniform Resource Locator), содержащий имя протокола по которому нужно обращаться к требуемой информации, адрес сервера и имя файла на этом сервере.

В общем виде формат URL выглядит так:

method://host.domain[:port]/path/filename

где:

method - одно из значений, перечисленных ниже:

- file – файл на локальной системе;
- http – файл на World Wide Web сервере;
- gopher – файл на Gopher сервере;
- wais – файл на WAIS (Wide Area Information Server) сервере;
- news – группа новостей телеконференции Usenet;

- telnet – выход на ресурсы сети Telnet;
- ftp – файл на FTP-сервере.

host.domain – доменное имя в сети Интернет.

port – номер UDP-порта (User Datagram Protocol – протокол пользовательских датаграмм).

Пример: ***http://www.agtu.ru/archive/index.html***.

Префикс http:// указывает, что далее следует адрес Web-страницы, **www.agtu.ru** – зарегистрированное доменное имя и, одновременно, имя Интернет-сервера, /archive описывает каталог с именем archiv на сервере, а index.html – имя файла.

Часто имена файлов на Интернет-серверах выглядят непривычно. Это связано с тем, что большинство серверов работают под управлением операционной системы UNIX, в которой имена файлов чаще всего не имеют расширения файла, идентифицирующего его тип. Основной кодировкой кириллицы в текстовых файлах на UNIX-серверах является KOI8-R.

В Интернете в настоящее время основным протоколом работы с информацией является гипертекстовый протокол **http** (hyper text transfer protocol) – основной режим работы современных Интернет-обозревателей (браузеров). В состав операционной системы Windows входит **Internet Explorer**, другими популярными Интернет-обозревателями являются также **Netscape Navigator**, **Opera** и пр.

Основными протоколами работы с электронная почтой (e-mail) являются **SMTP и POP3**. Отправка сообщений электронной почты осуществляется по протоколу SMTP (Simple Mail Transfer Protocol, 25-й UDP-порт), а прием по протоколу POP3 (Post Office Protocol, 110-й UDP-порт). Для работы с этими протоколами служат программы Outlook Express в ОС Windows, Microsoft Office Outlook, и пр.

При использовании электронной почты, работающей по протоколам POP3/SMTP, каждому абоненту присваивается уникальный почтовый адрес, формат которого имеет вид: <имя пользователя> @ < имя почтового сервера>, например a.ivanov@agtu.ru.

Организация информации в Интернете в таком виде, когда файлы связаны между собой информационными связями в виде гипертекстовых ссылок, называется всемирной паутиной и обозначается **www**. Под «паутиной» подразумеваются не линии связи, по которым идет сигнал, а связи информационные.

Набор файлов на одном Интернет-сервере, связанных между собой гипертекстовыми ссылками, называется **сайтом**. Например, сайт в Интернете может содержать информацию о фирме. Он имеет свой «входной»

файл – домашнюю страницу (home page), которая содержит в виде меню и названий разделов гипертекстовые ссылки на остальные части сайта. Обычно структура ссылок имеют иерархическую или более сложную структуру.

4.4 Способы подключения к Интернету конечных пользователей

В настоящее время существует достаточное большое количество способов простому пользователю подключиться к Интернету и пользоваться всеми его большими возможностями.

Старый способ подключения – с использованием аналогового модема для коммутируемых линий (dial-up подключение). При этом подключении заключается договор с провайдером, в котором указаны номер телефона, по которому должен звонить модем, имя пользователя, его пароль для подключения и пр. данные для настройки подключения, на счет договора вносится плата за трафик (объем информации, полученной и переданной при работе с Интернетом). Другой вид взаимоотношений с провайдером – работа в кредит. При этом имя пользователя и пароль у всех один, идентификация пользователей идет по номеру телефона. Счет за Интернет выставляется каждый месяц вместе с оплатой обычной телефонной связи. Скорость обычного коммутируемого соединения не может превышать 56 Кбит/с.

Высокую скорость подключения к Интернету обеспечивает технология xDSL. В основе технологии ADSL (Asymmetric Digital Subscriber Line – асимметричная цифровая абонентская линия) лежит принцип разделения частот для передачи голосового сигнала и для передачи данных. Для подключения ADSL скорость передачи данных к пользователю составляет от 64 Кбит/с до 8 Мбит/с, от пользователя – от 64 Кбит/с до 1 Мбит/с в зависимости от тарифного плана. Для жителей Архангельска на конец 2009 г. скорость исходящего трафика для всех ADSL-тарифов (от абонента в сеть Интернет) составляла 512 Кбит/с при ограничении скорости входящего трафика до 550 Кбит/с при стоимости безлимитного подключения (без ограничения трафика) 350 руб./мес. (оборудование – ADSL-модем предоставлялся бесплатно), при скорости до 1100 Кбит/с – 680 руб./мес., при скорости до 2220 Кбит/с – 1100 руб./мес., при скорости до 3000 Кбит/с – 2000 руб./мес.

Подключение к компьютеру модема-маршрутизатора выполняется через сетевой или USB-порт. Провод из телефонной розетки подключается

к сплиттеру (маленькая коробочка с 3-мя разъемами), к нему же подключается телефон и ADSL-модем. При этом возможно одновременное использование телефона для разговоров и Интернета.

Третья технология, обеспечивающая приемлемые скорости передачи данных – асимметричный доступ в Интернет через спутник.

Существуют две основные схемы работы через спутник – симметричная и асимметричная.

В первом случае клиент осуществляет и передачу запроса на спутник, и прием данных со спутника. Такое решение является достаточно дорогим, как по части клиентского оборудования, так и по стоимости обслуживания, и применяется в основном в тех случаях, когда его использование является либо единственно возможным, либо более дешевым, чем использование проводных или радиоканалов (например, в труднодоступных или удаленных районах с неразвитой инфраструктурой связи).

Во втором случае со спутника осуществляется лишь прием информации, в то время как передача запросов осуществляется по наземным каналам связи через Интернет-провайдера (см. рисунок 4.7).

Рисунок 4.7. Схема асимметричного спутникового доступа в Интернет

Стоимость такого решения более низкая – не требуется дорогостоящего спутникового оборудования передачи данных, для приема можно использовать стандартную «тарелку» совместно с DVB картой, для передачи исходящих запросов – обычный модем. DVB карта (сокращение от Digital Video Broadcast – передача цифрового видео) представляет собой ISA или PCI карту, которая производит обработку цифрового спутникового сигнала, поступающего с приемной антенны.

Жители европейской части России могут воспользоваться услугами двух наиболее известных провайдеров – Europe OnLine (EOL) и NTV

Internet (NTVi). Не так давно к этим двум компаниям добавилась новая – StarSpeeder (*http://www.starspeeder.by/*).

Для примера приведем данные о функционировании системы StarSpeeder.

Система запущена в коммерческую эксплуатацию 1 июня 2001 года. Трансляция сигнала осуществляется с помощью спутника «Europe*Star 1». StarSpeeder является системой нового поколения, использующей для своей работы стандартные Интернет-протоколы и технологию виртуальных частных сетей VPN (Virtual Private Network), в результате чего работают все Интернет-службы – POP3, IRC, ICQ и т.п. В этом заключается основное отличие данной системы от EOL и NTVi, работа которых основана на принципе прокси-сервера.

Алгоритм работы системы следующий.

Запрос от пользователя на требуемый ресурс уходит по наземным линиями связи в Интернет при помощи технологии VPN. По каналам Интернет через VPN-соединение запрос приходит к ресурсу Интернет.

Приходящий ответ отправляется на наземную станцию спутниковой связи. Станция спутниковой связи преобразует ответ клиенту в формат DVB и отправляет его на спутник "Europe*star 1".

Спутник "Europe*star 1" транслирует ответ клиенту в эфир. Ответ приходит вместе с общим DVB-потоком на приемную антенну клиента. Спутниковый сигнал принимается картой DVB-декодера и из потока извлекаются IP-пакеты клиента. Клиент получает ответ от нужного ресурса.

Для того, чтобы воспользоваться услугами системы, необходим стандартный набор оборудования, включающий в себя спутниковую антенну диаметром от 70 см (рекомендуемый диаметр – 90 см), DVB плату, а также комплект программного обеспечения.

Пользователю помимо Web-серфинга (HTTP) и FTP доступны также все обычные службы сети без каких-либо ограничений. Ограничение скорости передачи данных в системе StarSpeeder в настоящее время установлено на уровне примерно 400 Кбит/с на абонента.

Ежемесячная абонентская плата составляет около 500 руб.

4.5. Поиск информации в Интернете

Для поиска информации в Интернете созданы мощные средства: поисковые серверы (поисковики), каталоги (рубрикаторы), списки рейтингов, тематические списки ссылок, онлайновые энциклопедии и справочники и пр.

Поисковая система – комплекс программных и аппаратных средств для автоматического просмотра ресурсов Интернет, индексации их содержания и предоставления услуг по поиску информации Интернет-пользователям. Поисковые системы могут отличаться по эффективности поиска, по языку поиска (русский, английский и др.) и по некоторым другим возможностям. Например, одни поисковые системы находят информацию только в виде Web-страниц, другие могут просматривать и группы новостей, и файловые серверы.

Наиболее известны следующие международные системы для поиска информации в информационных ресурсах:

Google (*http://www.google.com/* или *http://www.google.ru/*);
Microsoft Live Search (*http://www.live.com/*);
Alta Vista (http://www.altavista.com/);
Yahoo! (http://www.yahoo.com/);
Infoseek (http://www.infoseek.com/);
Hot Bot (http://www.hotbot.com/)/.
Российские поисковые системы:
Рамблер (http://www.rambler.ru/);
Яндекс (http://www.yandex.ru/).
Апорт (http://www.aport.ru/).

Поисковые системы могут быть 2-х типов: универсальные и специализированные. Наиболее популярные современные поисковые системы сочетают в себе оба типа.

В универсальных системах используется обычный принцип поиска в неструктурированных документах – по заданной строке поиска.

На домашней странице поисковой системы обычно расположено поле для ввода строки поиска, может также присутствовать меню и прочие элементы. Домашняя страница системы **Google** может настраиваться зарегистрированным в системе пользователем. Большинство поисковых серверов предоставляют пользователям также услуги Интернет-почты (почтовые ящики), возможность создания личных Web-страниц, новости, гороскопы, курсы валют, прогноз погоды и прочие сервисы.

Строка поиска может состоять из одного слова или группы слов. Если слова разделены пробелами, ищутся документы, в которых присутствует хотя бы одно слово из перечисленных. Для поиска словосочетания его следует заключить в кавычки. В некоторых системах можно осуществлять поиск по части слова, оставшаяся часть слова заменяется знаком «*», как в шаблоне имени файла. Знак «+» или «&» между словами требуют обязательного присутствия всех слов в документе.

Часто существует также кнопка перехода к расширенному поиску. Главное отличие расширенного поиска – использование в запросе логических операторов и круглых скобок. Для построения сложного запроса используются логические операторы AND (И), OR (ИЛИ), NOT (НЕТ) и NEAR (около; не далее чем в 10 символах). Логические операторы ставятся между словами или словосочетаниями. Здесь могут использоваться даты документов размер документов и другие критерии. Интерфейсы расширенного поиска у разных поисковых систем существенно отличаются.

Поисковые системы обычно состоят из трех компонентов:

1. поисковый робот (агент, паук или кроулер), который перемещается по сети и собирает информацию;
2. база данных, которая содержит всю информацию, собираемую роботом;
3. поисковый механизм, который используется как интерфейс для взаимодействия пользователей с базой данных.

Поисковые роботы – это специальные программы, которые занимаются поиском страниц в сети, сохраняют гипертекстовые ссылки на эти страницы и автоматически индексируют присутствующую на них информацию для построения базы данных поисковика. При построении индекса исходные данные преобразуются так, чтобы объем базы был минимальным, а поиск осуществлялся очень быстро и давал максимум полезной информации.

При запросе к поисковой системе она отыскивает в своей базе данных информацию, соответствующую запросу, и выводит список ссылок. В этом списке представлены ссылки на различные Web-страницы, причем ссылки располагаются по степени убывания встреченных на данных страницах слов, совпадающих с ключевыми словами или по дате источника. При просмотре списка можно выбрать и просмотреть заинтересовавшие Вас страницы.

Признанным лидером среди поисковых систем является **Google**, который обеспечивает наиболее скоростной поиск с наибольшим количеством ссылок по многим тематикам. В распоряжении **Google** имеется более 15 000 серверов, разбросанных по дата-центрам всего мира, которые выполняют поиск информации. На рисунке 4.8 показано поле поиска системы **Google** с автоматически раскрывающимся контекстным списком ссылок после набора первых букв строки поиска.

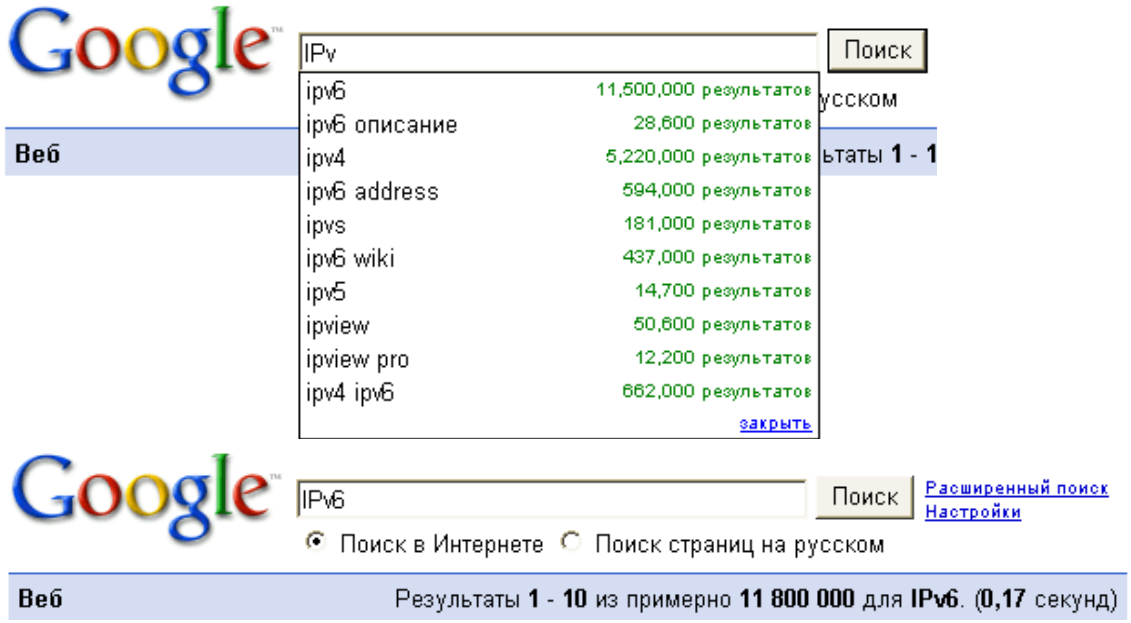


Рисунок 4.8. Поиск в системе Google

Приведенный на рисунке 4.7 пример поиска для строки **IPv** в **Google** занимает по времени 0,17 сек и находит примерно 11 800 000 ссылок, **Microsoft Live Search** находит 8 390 000 ссылок, аналогичный поиск в **Яндексе** находит только около 400 000 ссылок, в **Рамблере** – около 290 000 ссылок.

Каталог в Интернете – данные, структурированные по темам в виде иерархических структур. Ссылки в такие каталоги заносятся не автоматически, а с помощью администраторов. Как правило, хорошие каталоги Интернет обеспечивают разнообразный дополнительный сервис: поиск по строке поиска в своей базе данных, списки последних поступлений, списки наиболее интересных из них, выдачу случайной ссылки, автоматическое оповещение по электронной почте о свежих поступлениях. Один из наиболее популярных каталогов в России – ***http://list.mail.ru/***.

Рейтинги сайтов содержат списки наиболее посещаемых мест в Интернете. Одним из наиболее популярных рейтингов является Rambler's Top 100. (*http://top100.rambler.ru/top100/*). Популярность ресурса оценивается по ряду параметров, в том числе по количеству уникальных посетителей в единицу времени и хитам – количеству заходов на сайт за в единицу времени.

Онлайновые энциклопедии и справочники представляют собой электронные версии многих известных соответствующих бумажных документов. Одной из крупнейших онлайновых энциклопедий является ресурс «Яндекс. Энциклопедии» (*http://encycl.yandex.ru/*), который содержит 14

энциклопедий, в том числе Большую Советскую Энциклопедию, Энциклопедию Брокгауза и Эфрона и пр. К крупным относится и Энциклопедия Кирилла и Мефодия, которую можно найти по адресу www.km.ru.

Помимо традиционного словаря, ресурс Webopedia (*www.pcWebopedia.com*) имеет массу специализированных сервисов, например: "Кто есть кто в компьютерных технологиях", "Сравнительная таблица микропроцессоров", "История развития компьютерных технологий" и др.

Интенсивно развивается Википедия – свободная энциклопедия в Интернете (*http://ru.wikipedia.org*), в которой могут размещать свои статьи по любым вопросам, редактировать и дополнять существующие материалы любые желающие.

Для студентов и школьников полезным ресурсом является ***Интернет-портал Минобразования РФ*** (*http://www.edu.ru*, см. рисунок 4.9) и его **«Единое окно доступа к образовательным ресурсам»** *http://window.edu.ru.*

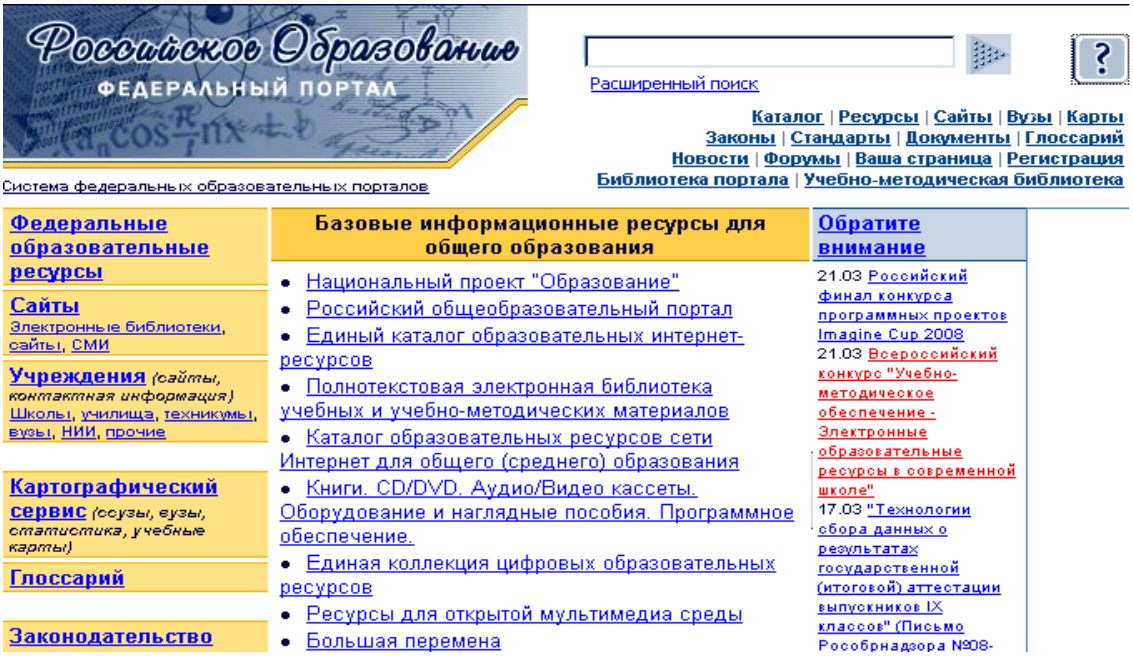


Рисунок 4.9. Федеральный портал Российское образование

4.6 Электронная почта в Интернете

Электронная почта (E-mail) – один из самых популярных сервисов Интернета. Первое почтовое сообщение было отправлено в 1971 году, когда Рей Томлинсон создал программу, отправляющую текстовые сообще-

ния с одного компьютера сети Arpanet на другой. И всего через год 3/4 передаваемых по сети сообщений были сообщениями электронной почты. В 1975 году появилась первая электронная рассылка, т. е. периодическая отправка на определенные адреса электронных писем. Вскоре в рассылках стали публиковаться письма подписчиков, на которые другие читатели отвечали также через рассылку. Так родилась UseNet – система электронных конференций.

В настоящее время для приема и хранения писем используются специальные почтовые серверы или Web-серверы, для работы пользователей с электронной почтой используются специальные программы – почтовые клиенты. Основные функции почтовых клиентов – прием писем, обеспечение их просмотра, распределение писем по папкам (Входящие, Отправленные, Удаленные и пр.), автоматизация создания новых писем и поддержка адресной книги. При подготовке электронного письма пользователь готовит текст сообщения и заполняет ряд стандартных полей: «To» (Кому), «CC» (Копия), «BCC – Blind Carbon Copy» (невидимая копия). Разница между полями CC и BCC заключается в том, что все те адреса, которые вы укажете в поле CC, будут видны всем участникам переписки, а те, что вы укажете в поле BCC, не будут видны всем адресатам (каждый будет видеть только свой адрес).

Пользователи электронной почты регистрируются на сервере, каждому из них дается адрес (например, *a.ivanov@agtu.ru*, где a.ivanov – имя пользователя и его почтового ящика, agtu.ru – имя домена, в котором находится почтовая служба).

Для работы специальных почтовых серверов и клиентов используются специальные протоколы, основные из них **POP3, IMAP, SMTP**.

Наиболее распространенным в настоящее время является протокол **POP3** (Post Office Protocol **–** почтовый протокол, версия 3, до этого были протоколы POP2, POP, но они не предоставляли полноценную защиту пароля доступа к ящику). Протокол **POP3** предусматривает обращение клиента с предложением забрать пришедшие письма, сообщение серверу имени ящика и пароля, загрузку писем на машину пользователя и удаление их из ящика.

Согласно правилам протокола **POP3**, все сообщения хранятся в одном файле, так что удаление с сервера пришедших писем может производиться только после полной их загрузки почтовым клиентом.

Для предоставления возможности выборочной работы с почтой создан протокол **IMAP** (Internet Mail Access Protocol – протокол доступа к сообщениям электронной почты в Интернете). Пользователь может просмот-

реть информацию о полученных сообщениях, об отправителях, узнать размеры, темы и выборочно загрузить необходимые письма. На сервере письма сгруппированы по папкам, пользователь может создавать новые папки для писем. Поддержкой протокола **IMAP**, хотя и не в полном объеме, обладает большинство современных почтовых клиентов.

В принципе для пересылки сообщений достаточно наличия в сети только POP-серверов. Почтовый отправитель переправит письмо серверу получателя. Однако если почтовый сервер адресата по каким-то причинам окажется неработоспособным, то письмо отправлено не будет и придется ждать, когда тот снова начнет функционировать. Для решения этой проблемы создан **SMTP**-протокол (Simple Mail Transfer Protocol – простой протокол пересылки почты), по которому работают все **SMTP**-серверы исходящей почты, занимающиеся накоплением отправляемых писем и обеспечением их доставки адресатам. После помещения пользователем на **SMTP**-сервер писем, тот начинает запрашивать почтовые серверы, которым эти сообщения адресованы, на предмет готовности принять эти письма. Если сервер адресата не готов, то **SMTP**-сервер пробует соединиться с ним через некоторое время, а если он недоступен на длительный срок – возвращает письмо отправителю (на обратный адрес, указанный в письме). Кроме того, при загрузке сообщений на **SMTP**-сервер проводится анализ адресов получателей и выясняется, существуют ли такие адреса в принципе. Если нет, сервер возвращает письмо отправителю с информацией о том, что получатель не найден.

Вместе с усовершенствованием механизмов передачи электронных посланий менялась и структура самого письма. Первоначально письма представляли собой простые текстовые файлы. Однако впоследствии был разработан так называемый стандарт **MIME** – Multipurpose Internet Mail Extensions (многоцелевые расширения почтового стандарта Интернет). В соответствии с этим стандартом электронное письмо может состоять из нескольких частей, содержащих свой пакет данных в отдельном формате. Так с помощью **MIME** можно одновременно поместить в письмо текст, Web-страницу, графический, звуковой либо исполняемый файл или архив. Каждая часть будет иметь свой заголовок, в котором обозначен формат этой части. Почтовый клиент способен анализировать заголовки и соответствующим образом воспроизводить фрагменты письма.

В состав операционной системы **Windows** входит почтовый клиент – **Outlook Express**, имеющий возможность работы с электронной почтой по протоколам **POP3**, **IMAP** и **HTML**. В состав **Microsoft Office System** входит приложение **Outlook**, имеющее несколько большие возможности по

сравнению с **Outlook Express**. Другие почтовые клиента: Eudora, Netcsape Mail, The Bat! и т. д.

При настройке почтовой программы необходимо указать имена почтовых серверов для входящей почты (например, pop3.agtu.ru) и исходящей почты (например, Smtp.agtu.ru).

В состав новейших почтовых систем включаются функции шифрования. Самыми распространенными системами шифрования являются **S/MIME**, **PEM** (privacy enhanced mail – почта с повышенной конфиденциальностью) и **PGP** (pretty good privacy – достаточно высокая степень конфиденциальности). **S/MIME** – стандартная система шифрования, которую поддерживают **Outlook Express**, **Netscape** и пр.

Существует также **Web-почта** с использованием обычного Web-обозревателя в качестве почтового клиента и Интернет-серверов для работы с почтой. В сети достаточно много широко известных специализированных почтовых сайтов и поисковиков с почтовыми сервисами, предоставляющих бесплатные почтовые услуги для личного использования всем желающим. Например, почтовая служба фирмы Microsoft – *www.hotmail.com* – предоставляет любому зарегистрировавшемуся пользователю почтовый ящик размером до 5 ГБ, бесплатное приложение электронной почты с функцией поиска компании Google – служба Gmail – ящик размером 6544 Мбайт. Русская почтовая служба в Интернете *www.mail.ru* после регистрации для хранения писем выделяет 10 Гб свободного пространства, если в процессе использования в почтовом ящике остается менее 100 Мб, то пользователь может увеличить его объем еще на 2 Гб. Если со временем и этого места не будет хватать, то возможно и дальнейшее поэтапное расширение. Размер одного письма на @mail.ru не может превышать 30 Мбайт. Бесплатные почтовые службы имеют также поисковики Яндекс, Рамблер, Yahoo! и др. Впрочем, существуют в Интернете и платные службы Web-почты.

Web-почта предоставляет все типичные функции электронной почты. На Web-страничке пользователю предлагается ввести имя и пароль, после чего он видит папки своих писем «Входящие», «Отправленные», «Черновики», «Удаленные» и др. и может читать выбранные письма и создавать и отправлять новые письма.

Основными плюсами Web-почты являются доступность с любого компьютера, имеющего выход в Интернет, хорошая борьба с рекламными рассылками (спамом), достаточно простое имя типа *user@mail.ru* (правда, многие имена уже заняты из-за миллионов зарегистрированных в службе

пользователей и часто бывает трудно придумать для себя подходящее имя).

К недостаткам бесплатной Web-почты следует отнести то, что на почтовых сайтах всегда присутствует много рекламы.

Web-почта может использовать для установления связи протоколы с шифрованием **SSL/TLS** (Secure Sockets Layer / Transport Layer Security), что обеспечивает конфиденциальность сообщений. С постоянным использованием этого протокола работает, например, сервер Hotmail.com.

4.7 RRS-каналы

RRS-каналы – новый вид представления списков новостей в Интернете. RSS включает в себя несколько стандартов, включая RSS 0.91 – **R**ich **S**ite **S**ummary (), RSS 0.9, 1.0 и 1.1 – **R**DF **S**ite **S**ummary и RSS 2.0 – **R**eally **S**imple **S**yndication (очень простое приобретение информации). Atom – еще одна XML-структура, используемая для распространения содержимого.

Основа версии RSS 2.0 – формат XML версии 1.0. Более старые версии RSS 0.9× и RSS 1.0 использовали форматы XML и RDF (см. *http://rdfabout.com*). Существует несовместимость форматов 2.0 и 0.9×, 1.0. Многие современные браузеры, почтовые клиенты и Интернет-пейджеры умеют работать с RSS-каналами, среди них Microsoft Internet Explorer (начиная с версии 7), Microsoft Office Outlook, Opera, Mozilla Firefox, Miranda, Safari, Maxthon и др. Кроме того, существуют специализированные приложения (RSS-агрегаторы), собирающие и обрабатывающие информацию RSS-каналов. Также очень популярны Web-агрегаторы, представляющие собой сайты по сбору и отображению RSS-каналов, такие как Яндекс.Лента, Google Reader, Новотека, Bloglines и пр.

RSS-канал отображается в браузере, почтовом клиенте или агрегаторе, как список заголовков. Заголовки обновляются автоматически. Выбрав любой заголовок, можно прочитать информацию по этой теме.

Преимуществом RSS является сосредоточение всей информации по выбранной теме канала из нескольких веб-источников в одном месте. Больше не нужно посещать различные веб-узлы в поисках информации. С помощью RSS можно получить краткое изложение содержимого веб-узлов и решать, какие именно статьи следует прочитать, щелкнув соответствующую ссылку.

RSS-каналы можно найти с помощью специализированных поисковых систем. В Интернете существуют веб-узлы, которые ведут большие списки доступных RSS-каналов, например, по адресу *http://subscribe.ru/catalog?rss*

размещен RSS-Каталог, содержащий на данный момент 20594 ссылки на RSS-каналы, сгруппированные по следующим тематикам (в скобках – количество ссылок в группе):

Отрасли экономики *(415)*

Промышленность, Высокие технологии, Банки, Недвижимость, Строительство, Мобильная связь, Сельское хозяйство и пищевая промышленность, Услуги

Экономика. Бизнес *(517)*

Торговля, Платежные системы, Поиск партнеров. Инвестиции, Школа бизнеса, Новости, Обзоры, Финансы, Электронная коммерция, Бухучет. Налоги, Реклама, Маркетинг и продажи, Страхование. Безопасность, Книги и ПО, Кондиционирование и вентиляция, Технологии для бизнеса

Компьютеры, ПО и Интернет *(6367)*

Железо, Программное обеспечение, Компьютерные игры, Интернет, Документация, книги, словари, Дизайн, Персональные дневники и сайты, Порталы, Сообщества и форумы

Культура и искусство *(604)*

Новости. Обзоры. Анонсы, История, Религия, Человек и общество, Психология, Литература, Ювелирное искусство, Живопись и Архитектура, Музыка, Театр, Фотография, Музеи, Танцы

Государство *(190)*

Политика, Законодательство, Армия и силовые структуры

Работа, образование *(246)*

Поиск работы, Обучение, Студенту, Иностранные языки

Отдых *(781)*

Веселости, Тесты, Знакомства, О любви и сексе, Эротические фото, Это интересно, Праздник, Кино, Азартные игры, Интеллектуальные игры, Туризм и путешествия, Хобби, Астрология, Таинственное

Дом *(287)*

Мода и Красота, Милым дамам, Дети, Флора и Фауна, Полезности, Строительство и ремонт, Интерьер, Вкусности, Природа

Новости. СМИ *(543)*

Пресса. Новости. Обзоры, Радио, Новости дня, Телевидение

Наука и техника *(530)*

Медицина и здоровье, Гуманитарные науки, Естественные науки, Точные науки, Новости науки, Моделирование, Техника, Автомобили

Спорт *(269)*

Спорт. Новости. Обзоры. Размышления, Футбол, Теннис, Баскетбол, Автогонки, Боевые искусства, Прогнозы и ставки, Экстремальный спорт, Прочее

Страны и Регионы *(435)*

Россия, Украина, Израиль, Европейские страны, Другие страны

Прогноз погоды *(7)*

Россия

Прочие *(6554)*

Еще не классифицированные

Удобным средством для работы с RSS-каналами является Microsoft Office Outlook. Данная система для начала работы с RSS-каналами предлагает достаточно большой список каналов, на которые можно подписаться, названия некоторых из них приведены ниже:

Избранные RSS-каналы

Новости	Развлечения и спорт
Новости MSNBC	*Развлекательные программы MSNBC*
Лучшие новостные ролики	*Музыкальные клипы*
Деловые новости MSNBC	*Новости на веб-узле FOXsports.com*
Журнал «Forbes»	*Спортивные сюжеты*

Веб-каналы Майкрософт

MSDN	
Загрузочные файлы MSDN	*Статьи MSDN по Office*

Сетевые журналы по Office

Продукты и компоненты	Избранное
Access	*Сетевой журнал Хозяйки Office*
Excel	*Сетевой журнал по Office Online*
OneNote	
Outlook	
Project	
Publisher	
SharePoint Designer	
Word	

4.8 Общение в Интернете в реальном времени

С момента появления компьютерных сетей и Интернета и по настоящее время множество людей использовали эти средства коммуникации для обмена новостями, своими мнениями и знаниями, спрашивали совета у специалистов и просто разговаривали на всевозможные темы.

Первоначально главным средством для этого была электронная почта и серверы списков рассылки. Серверы рассылки формируют группы списков по различным тематикам, к этим спискам адресатов может присоединиться любой желающий и он будет получать все сообщения членов этой группы и сам сможет отвечать в эту группу, его сообщения будут рассылаться по всем адресам в списке группы. В настоящее время такой способ общения известен как группы новостей (News) в программе Outlook Express, конференции или форумы на Интернет-страницах.

Другие способы общения – службы мгновенных сообщений или Интернет-пейджеры и чат (chat) двух видов – IRC (Internet Relay Chat) и Web-чат.

4.8.1 Службы мгновенных сообщений

Программы **IM** (Instant Messaging) – обмена мгновенными сообщениями – позволяют пользователям в Интернете устанавливать связь друг с другом и общаться в реальном времени или посылать сообщения неактивному пользователю, который получит их после подключения к системе. Самой известной **IM**-программой (Интернет-пейджером) является программа **ICQ** (аббревиатура ICQ созвучна английской фразе «I seek you» – я ищу тебя, произносится ай-сик-ю или, в РФ, «аська»). Доля **ICQ** среди аналогичных программ в РФ составляет более 70%, **Microsoft Manager** – менее 10%, **QIP** – 7%, **Skype** – 4%, прочие (**Jabber, Google Talk, Miranda** и др.) – 8%. Число зарегистрированных пользователей **ICQ** более 50 млн. человек.

Система **ICQ** была разработана четырьмя молодыми программистами из Тель-Авива (Израиль): Сефи Вигисер, Арик Варди, Яир Гольдфингер и Амнон Амир. Они основали свою компанию Mirabilis (лат. чудесный, удивительный) и в 1996 году системы начала свое распространение в Интернете. В 1997 года компания объявила о том, что количество пользователей программы достигло миллиона. В 1998 году компания Mirabilis сообщила о том, что сеть **ICQ** растет со скоростью 1 миллион пользователей за 23 дня. Количество одновременных подключений составляло ~ 500 тысяч. В 1998 года компания Mirabilis продала права на ICQ крупнейшему американскому провайдеру **AOL** (America Online) за 407 миллионов долларов. В результате компания Mirabilis была преобразована в **ICQ Inc.** и стала частью корпорации America Online.

Помимо основных функций обмена сообщениями и поиска зарегистрированных пользователей по задаваемым параметрам (фамилия, имя, прозвище (ник), город, пол, возраст и пр. – см. рисунок 4.10), **ICQ** может также выполнять большое количество других задач, таких как **ICQ**-чат, организация видеоконференций, интернет-телефония, отправка SMS-сообщений, передача файлов, проверка почты и т.д.

ICQ и другие Интернет-пейджеры – программное обеспечение, которое может быть бесплатное загружено с сайта производителя и бесплатно использоваться.

Для работы с **IM**-программой новому пользователю необходимо установить связь с одним из серверов этой системы и зарегистрироваться в системе, после чего он получает уникальный 9-ти значный номер в системе (**UIN**). **ICQ** имеет кроме основного сервера login.icq.com более 20 дополнительных серверов, размещенных по всему миру (например,

icq.mirabilis.com, icq1.mirabilis.com и пр.). В 2005 году Rambler Media и America Online (**AOL**, собственник системы **ICQ**) заключили соглашение по использованию русской версии Рамблер-ICQ, которая интегрирована со многими службами системы Рамблер. В 2007 году Рамблер-ICQ использовали около 2 млн. человек.

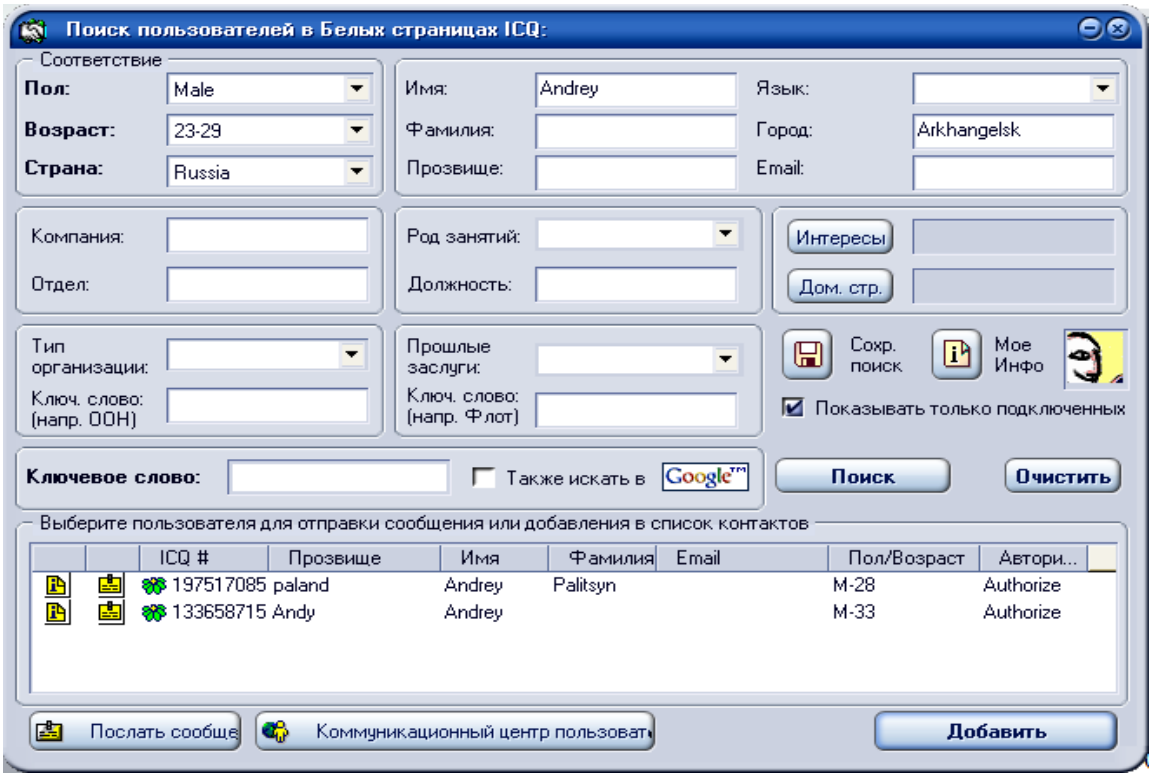


Рисунок 4.10. Поиск пользователей в базе ICQ

Существуют многофункциональные клиенты, поддерживающие в Windows несколько протоколов систем обмена мгновенными сообщениями:

- **Miranda IM** – протоколы ICQ, IRC, AIM, MSN, Jabber, Google Talk, Yahoo, Skype, Gadu-Gadu, Tlen, Netsend и пр.
- **SIM IM** – протоколы ICQ, AIM, MSN, Jabber, Yahoo и пр.
- **Trillian** – протоколы ICQ, MSN, Jabber, Yahoo и пр.
- **QIP** – поддерживает ICQ, Jabber и пр.
- **Pidgin** – протоколы ICQ, IRC, AIM, MSN, Jabber, Yahoo, Gadu-Gadu и пр.
- **Instantbird** – протоколы XMPP, Google Talk, ICQ, Yahoo!, AIM, MSN, Gadu-Gadu и пр.

4.8.2 Коллективное виртуальное общение

Если **IM**-сети в основном предназначены для индивидуального общения, то программы организации **чатов, Web-форумов** и **Web-конференций, гостевых книг, блогов** – для общения групп людей, имеющих какие-либо общие интересы.

Чат (chat) – специальной программное средство для организации виртуального общения одновременно большого числа людей в Интернете – существует в двух разных видах:

1. **IRC** – Internet Relay Chat – использует специальные серверы и клиенты для своей работы, подключенные к Интернету;
2. **Web-чат** – реализован, как программное обеспечение, работающее на обычных Web-страницах.

У этих двух видов много общего: для участия в разговорах нужно зарегистрироваться и задать себе ник (псевдоним), разговоры ведутся в общих тематических «комнатах» – окнах системы (или каналах в IRC), можно уйти в «приват» – поговорить тет-а-тет в отдельной «комнате» и пр.

В операционной системе Windows 95 среди ее стандартных программ существовало приложение Microsoft Chat (Comic Chat), фирма Microsoft имела собственные chat-серверы (chat.msn.com, chat.microsoft.com). Впоследствии эти серверы были закрыты и программа Microsoft Chat была убрана из Windows.

IRC-сеть состоит из серверов, соединенных друг с другом В мире существуют десятки IRC-сетей. Наиболее старые и известные – EFNet (существует с 1990 г.), DALnet (1994), IRCNet (1996) и RusNet (1997), в каждой из этих сетей десятки серверов и десятки каналов (комнат). Более полный список можно найти на *www.ircnet.ru*.

Самым известным IRC-клиентом (программой для работы с IRC-серверами через Интернет) является mIRC. После его установки и настройки подключения к Интернету можно выбрать сеть и канал сети среди множества их в списке этого клиента. Список может редактироваться и обновляться. Другие клиенты под Windows: Pirch, Virc, IceChat, HydraIR, Klient, XChat (последний изначально был создан под Linux).

Наиболее популярными программами – IRC-серверами являются: IRCPlus, Conference Room, WIRCsrv, IRCD.

Существенных преимуществ IRC перед Web-чатами в настоящее время не существует. Недостаток IRC для простых пользователей – необходимость устанавливать специальную программу, настраивать ее и разбираться с особенностями ее использования.

Web-чаты существуют на тысячах известных и широко посещаемых сайтах, например *chat.mail.ru, www.chat.ru* и пр. На этих сайтах существуют десятки тематических комнат для группового общения, каждый желающий может создать свою комнату. Так, на сайте *chat.mail.ru* более 300 комнат, вот некоторые из них (в скобках указано количество человек в данный момент в комнате): Все_на_свете (48); Любовь (26); Знакомства (122); Питер (38); Москва!!! (58); Работа (13); Владивосток (24); КАЗАНЬ!!! (24); МИНСК (19); КРАСНОДАР!!!!! (18); Нормальное_общение (16); Компьютерная_помощь (18); ИРКУТСК__НАШ!!!! (20); ОМСК (24); Футбол (8); *****КлУб_ПыШнЫх_КрАсАвИц***** (26); "Замок ""Сумеречный грифон"" (10)" и т. д. Следует отметить, что современные программные средства организации чатов на Web-страницах делают их не менее быстрыми в передаче и показе сообщений, чем IRC.

Сетевые новости (Netnews) или группы новостей (Newsgroups) – это глобальная система Интернет-конференций, которая позволяет организовать текстовые дискуссии в рамках тематических групп. Информация, которую можно прочитать в группе новостей, – это необязательно новости в привычном смысле слова. Это может быть любое сообщение, поданное для обсуждения в данную группу новостей, например, предложение работы или ее поиск, просьбы о консультациях по каким-либо вопросам и многое другое.

Интернет-конференции используют для передачи сообщений протокол NNTP. По строению этот протокол во многом сходен с протоколом приема и передачи электронной почты SMTP. Протокол NNTP, как и SMTP, является текстовым, то есть все команды и ответы на них являются обычными текстовыми строками. Важной особенностью протокола NNTP является его эффективность в случае сложных графов связей между серверами новостей. Чтобы одно и то же сообщение не передавалось многократно, отправляющий сервер сначала сообщает идентификатор нового сообщения, а само сообщение отправляет только после подтверждения принимающей стороны о том, что этого сообщения там еще нет.

Сообщение, посылаемое в Newsgroups, становится доступной для всех участников группы новостей на определенный срок. Существуют специальные клиентские программы для работы с группами новостей, например Newsreader, однако чаще для этого используется Outlook Express, который, помимо почтового клиента, содержит программу для чтения новостей.

Существует огромное количество новостных серверов, которые образуют целую инфраструктуру, обеспечивающую механизм передачи но-

востей по всему миру. Каждый узел сети, получивший сообщение, передает его всем узлам, с которыми он обменивается новостями.

Web-форум – специальное программное обеспечение для организации общения посетителей Web-сайта. Сообщения в форум передаются по электронной почте. Читать сообщения форума может любой желающий, чтобы отправить сообщение в форум, обычно нужно в нем зарегистрироваться, поэтому у каждого сообщения присутствует информация о его авторе и дате/времени отправления.

Форум предлагает набор разделов для обсуждения. Работа форума заключается в создании пользователями тем в разделах и последующим обсуждении внутри этих тем. Отклонение от начальной темы обсуждения часто запрещено правилами поведения в форуме. За соблюдением правил следит модераторы и администраторы – участники, наделенные возможностью редактировать, перемещать и удалять чужие сообщения в определенном разделе или теме, а также контролировать к ним доступ отдельных участников.

По методу формирования набора тем форумы бывают с динамическим списком тем и с постоянным списком тем. В форумах с динамическим списком тем простые участники могут создавать новую тему в рамках тематики форума.

Форум отличается от чата разделением обсуждаемых тем и возможностью общения не в реальном времени. Это располагает к более серьезным обсуждениям, поскольку предоставляет отвечающим больше времени на обдумывание ответа. Форумы часто используются для разного рода консультаций, в работе служб технической поддержки.

В настоящее время Web-форумы почти полностью вытеснили новостные группы на базе NNTP и являются одним из наиболее популярных способов обсуждения вопросов во Всемирной паутине. На данный момент форумы сосуществуют наравне с блогами. Эти две формы общения в Интернете практически не уступают друг другу по популярности.

Блоги – личные электронные дневники на Web-страницах.

В Интернете имеются сайты, главное назначение которых – размещение блогов всех желающих, например, *www.livejournal.com*. Пользователям Web-почты также часто предоставляется возможность вести свои дневники на страницах почтового сайта, например, очень большое количество блогов можно найти на @mail.ru. Особенность этого вида общения – любой желающий может читать блоги и писать свои комментарии к записям автора блога, каждая такая запись имеет дату и время создания. В разделе «Звездные блоги на Mail.Ru» присутствуют дневники сотен Россий-

ских знаменитостей. Иногда подобные блоги преследуют рекламные цели – сообщения о новых дисках, личных сайтах звезд и т. п.

Сайты некоторых известных корпораций, например, Microsoft, организуют на своих Интернет-серверах страницы с блогами своих сотрудников. Предполагается, что они будут там рассказывать всем любопытствующим, в основном, о своей работе.

На сайте корпорации Microsoft можно найти достаточно большие списки ее сотрудников-блоггеров, количество только русскоязычных блогов более 20, вот некоторые из них:

• **Александр Ложечкин** Technical Evangelist Manager, Москва
http://blogs.gotdotnet.ru/personal/allo/

• **Алексей Еременко** специалист по Microsoft Dynamics, Москва
http://blogs.msdn.com/aeremenk/default.aspx

• **Екатерина Лажинцева** Lead Subsidiary Program Manager, Москва
http://blogs.technet.com/katerina/default.aspx

• **Людмила Фокина** группа разработчиков SQL Server, Redmond
http://blogs.technet.com/lyudmila_fokina/default.aspx.

Пример текста одной из записей блога, автор – *Alexander Lozhechkin*:

19 ЯНВАРЯ 2008 Г.

Отложенная отправка писем в Microsoft Outlook

Знакомо ли вам такое чувство, когда хочется выдернуть сетевой шнур из компьютера или выключить Wi-Fi на ноутбуке после того, как вы поняли, что только что отправили письмо, которое не нужно было отправлять? Психологам еще предстоит разобраться в этом феномене, но человеку свойственна «крепость задним умом», когда понимание правильного решение или (что чаще) совершенной ошибки приходит сразу после того, как исправить сделанное уже невозможно.

Почитать... *Other*
19 января 2008 г. 12:01:32 (Russian Standard Time, UTC+03:00) *Комментарии[13] Trackback*

4.8.3 IP-телефония

IP-телефония (Интернет-телефония) – технология, которая использует Интернет для передачи речевых сигналов.

При разговоре по ip-телефону голосовые сигналы (слова, которые мы произносим) преобразуются в сжатые пакеты данных. Эти пакеты данных посылаются через Интернет другой стороне. Когда пакеты данных достигают адресата, они декодируются в исходные голосовые сигналы. Преимущества ip-телефонии перед обычной телефонной связью:

- не нужно платить за междугородние звонки (только за Интернет);

- используется передовая технология сжатия голосовых сигналов, в результате более эффективно использует емкость линий связи.

Существуют 3 типа ip-телефонии:

1. с компьютера на компьютер (при этом к его звуковой плате нужно подключить микрофон);
2. с компьютера на телефон;
3. с телефона на телефон.

При передаче речевого сигнала сжатые пакеты данных поступают в Интернет с адресом назначения. Каждый пакет данных проходит собственный путь до адресата, по различным маршрутам. Для адресата пакеты данных перегруппировываются и декодируются в голосовые сигналы оригинала.

Широкое распространение ip-телефонии началось в 1995 г., когда израильская компания VocalTec предложила первую версию программы Internet Phone, разработанную для владельцев мультимедийных ПК, работающих под Windows. Программа работала с использованием сети серверов Internet Phone, и тысячи людей использовали эту программу для общения.

Общий принцип использования телефонных серверов IP-телефонии для организации междугородней (международной) связи 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), сжимает, разбивает на пакеты и отправляет через Интернет по назначению с использованием протокола Интернета (TCP/IP). Для пакетов, приходящих из Интернета на телефонный сервер и уходящих в телефонную линию, операция происходит в обратном порядке. Обе операции (вход сигнала в телефонную сеть и его выход из телефонной сети) происходят практически одновременно, что позволяет обеспечить полнодуплексный разговор. На основе этих базовых операций можно построить много различных конфигураций. Допустим, звонок телефон-компьютер или компьютер-телефон может обеспечивать один телефонный сервер. Для организации связи телефон-телефон нужно два сервера.

В 1996 г. было разработано специальное программное обеспечение и оборудование – VocalTec Telephone Gateway (VTG), которое использовало многоканальные голосовые платы Dialogic как интерфейс с обычными телефонными аппаратами и линиями. Голосовые платы позволяли, во-первых, одной системе VTG поддерживать до восьми независимых теле-

фонных разговоров через Интернет, во-вторых, убрали проблему адресации, взяв на себя преобразование обычных телефонных номеров в IP-адреса и обратно.

В настоящее время IP-телефония может быть также организована в масштабах одного предприятия, что исключает затраты на приобретение офисной АТС и организацию дополнительных кабельных соединений при наличии локальной компьютерной сети. На рисунке 4.11 показана одна из таких возможных схем. Программное обеспечение Cisco CallManager Express, установленное на маршрутизаторе, выполняя функции офисной АТС, служит для:

- организации внутренней телефонной сети, сопряженной с внешними телефонными сетями;
- осуществление звонков между внутренними и внешними абонентами;
- распределение входящих звонков и пр.

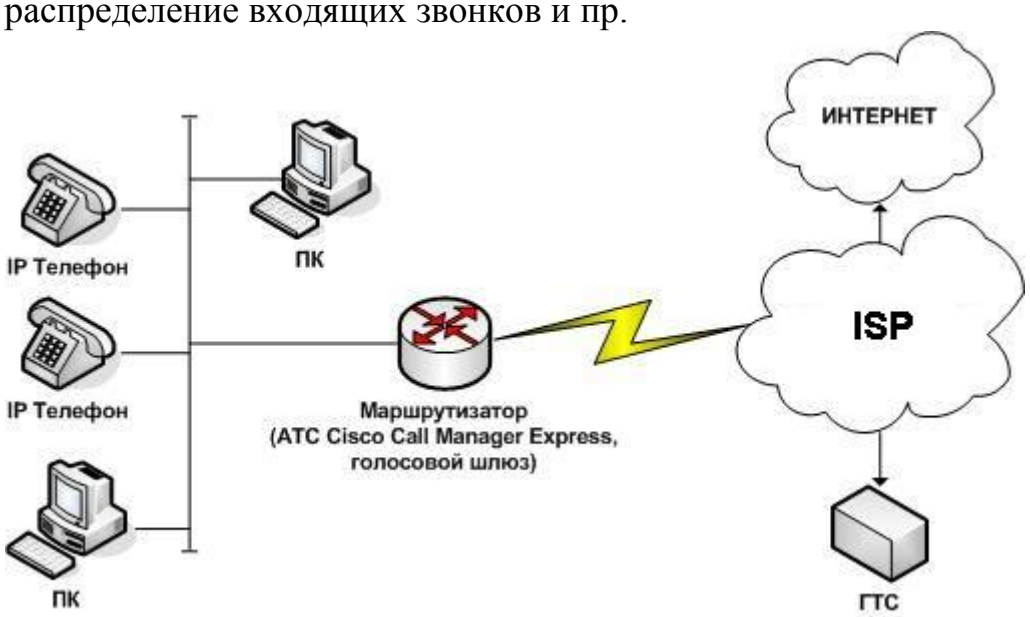


Рисунок 4.11. Схема организации IP-телефонии предприятия

4.9 Интернет-радио и Интернет-телевидение

Интернет вещание – радиоканалы и телеканалы, которые можно слушать и смотреть на Web-страницах сайтов, посвященных этому или через мультимедийные приложения – Windows Media Player, WinAmp, RealPlayer и пр.

Например, проект РадиоТВ.ОнРу.ру (*www.radiotv.onru.ru*) предоставляет всем посетителям возможность прослушивания интернет-радиостанций и просмотра интернет-телевидения со своего сайта (при установленном в системе Windows Media Player-е) с битрейтом на выбор: для радио 32, 48, 128 Кбит/с, телеканалы 128, 256, 300, 350 Кбит/с. Для просмотра телеканалов необходима скорость подключения к Интернету не менее 1 Мбит/с. На этом сайте можно слушать и смотреть как обычные радиостанции и телеканалы, так и радиостанции и телеканалы, вещающие исключительно в сети Интернет. Списки доступных Интернет-радиостанций и Интернет-телеканалов постоянно пополняются. На сайте можно зарегистрироваться и после этого пользоваться возможностью записывать выбранный канал в заданное время для последующего его прослушивания или просмотра.

Список некоторых радиостанций и телеканалов, вещающих через Интернет, на сайте OnRu.ru:

Интернет радио

Европа Плюс сайт: *http://www.europaplus.ru/*
частота: 106.2 FM (Москва)
48 Кбит/с – http://www.radiotv.onru.ru/fradio/europaplus_48.html
128 Кбит/с – http://www.radiotv.onru.ru/fradio/europaplus_128.html

Энергия ФМ сайт: *http://www.energyfm.ru/*
частота: 104.2 FM (Москва)
до 128 Кбит/с – http://www.radiotv.onru.ru/fradio/energy_128.html

MAXIMUM сайт: *http://www.maximum.ru/*
частота: 103.7 FM (Москва)
48 Кбит/с – http://www.radiotv.onru.ru/fradio/maximum_48.html
128 Кбит/с – http://www.radiotv.onru.ru/fradio/maximum_128.html

НАШЕ Радио сайт: *http://www.nashe.ru/*
частота: 101.7 FM (Москва)
32 Кбит/с – http://www.radiotv.onru.ru/fradio/nashe_32.html
48 Кбит/с – http://www.radiotv.onru.ru/fradio/nashe_48.html
128 Кбит/с – http://www.radiotv.onru.ru/fradio/nashe_128.html

Интернет телевидение

РТР-Планета сайт: *http://www.rtrplaneta.ru/*
Москва (Live)
до *64 Кбит/с* – http://www.radiotv.onru.ru/ftv/rtrplaneta_64.html
до *128 Кбит/с* – http://www.radiotv.onru.ru/ftv/rtrplaneta_128.html
до *230 Кбит/с* – http://www.radiotv.onru.ru/ftv/rtrplaneta_230.html

Мир ТВ
Москва (Live)
до *128 Кбит/с* – http://www.radiotv.onru.ru/ftv/mirtv_128.html
до *300 Кбит/с* – http://www.radiotv.onru.ru/ftv/mirtv_300.html

РБК ТВ	**РБК ТВ** сайт: *http://www.rbctv.ru/* Москва (Live) до *350 Кбит/с* – http://www.radiotv.onru.ru/ftv/rbk_350.html
Rambler	**Rambler ТелеСеть** сайт: *http://www.rambler-tv.ru/* Москва (Live) до *256 Кбит/с* – http://www.radiotv.onru.ru/ftv/ramblertv_256.html

На сайте *http://sradio.ru/* можно найти список 1878 радиостанций из разных стран и городов мира, значительная часть радиостанций имеет каналы вещания через Интернет. Сайт *E-Radio.Ru*, посвящен Интернет-радиостанциям России, стран СНГ, русскоязычным и ведущим музыкальным Интернет-радиостанциям мира, вещающим в потоковых форматах для Windows Media Player и Real Player. Имеется множество и других сайтов подобной тематики.

Интернет-телевидение предъявляет более высокие требования к скорости соединения пользователя с сетью. Имеется достаточно много сайтов, назначение которых – подключение пользователя к телеканалам, вещающим в Интернете. Например, TV-portal.ru (*http://www.tv-portal.ru/*) имеет ссылки почти на 400 каналов, при поиске канала позволяет задать язык, страну и тематику вещания, а также Интернет-TV-плеер, через который возможен просмотр – Windows Media Player, Real Player, RuTube и др. Фрагмент сайта с некоторыми кнопками выбора телеканалов показан на рисунке 4.12.

Рисунок 4.12. Некоторые кнопки выбора телеканалов на TV-portal.ru

4.10 Электронная коммерция

Электронная коммерция – форма взаимодействия продавцов и покупателей, при которой выбор и заказ товаров осуществляется через компьютерные сети, а расчеты между покупателем и поставщиком осуществляются с использованием электронных документов или средств платежа.

При классификации электронного бизнеса выделяют несколько категорий:

B2B (business-to-business) – бизнес для бизнеса – электронная коммерция между двумя компаниям;

B2C (business-to-consumer) – бизнес, ориентированный на конечного физического потребителя;

C2C (consumer-to-consumer) – бизнес, обеспечивающий взаимодействие между большим количеством физических потребителей;

C2B (consumer-to-business) – системы обработки ценовых заявок, по которым потребители хотели бы приобрести товары и услуги;

B2A (business-to-administration) – бизнес, оперирующий сделками между частными компаниями и правительственными организациями (администрацией);

C2A (consumer-to-administration) – бизнес, построенный на организации взаимоотношений физических лиц и государственных служб.

Глобальная сеть Интернет сделала электронную коммерцию доступной для фирм любого масштаба. Если раньше организация электронного обмена данными требовала заметных вложений в коммуникационную инфраструктуру и была по плечу лишь крупным компаниям, то использование Интернета позволяет сегодня вступить в ряды «электронных торговцев» и небольшим фирмам. Web-магазин дает любой компании возможность обслуживать покупателей всего мира. Подобный on-line бизнес формирует новый канал для сбыта – «виртуальный», требующий небольших материальных вложений. Если информация, услуги или продукция (например, программное обеспечение, книги и музыка в компьютерных форматах) могут быть поставлены через Web, то весь процесс продажи (включая оплату) может происходить в on-line режиме.

На сегодняшний день доминирующим платежным средством при on-line покупках являются кредитные карточки. Однако на сцену выходят и новые платежные инструменты: смарт-карты, цифровые деньги, микроплатежи и электронные чеки.

Электронная коммерция включает в себя не только on-line транзакции. В область, охватываемую этим понятием, входят и такие виды деятельности, как проведение маркетинговых исследований, определение

возможностей и партнеров, поддержка связей с поставщиками и потребителями, организация документооборота и пр. Таким образом, электронная коммерция является комплексным понятием и включает в себя электронный обмен данными как одну из составляющих.

Для простых потребителей наиболее важной категорией электронного бизнеса является B2C. К этой категории бизнеса относится весьма значительный круг предприятий электронной коммерции: Интернет-магазины, платные сервисы для физических лиц, электронные казино, многочисленные компании, продающие консультационные и информационные услуги. В структуру B2C-компании обычно входят следующие составляющие:

- Интерактивный Web-сайт, содержащий информацию о продукции, товарах и услугах, прайс-листы и каталоги. Как правило, сайт предоставляет клиенту возможность оформить on-line-заказ и отслеживать стадии его исполнения
- Площадка хостинга сайта – место, где сайт физически размещен. Это может быть собственный сервер компании, расположенный в офисе фирмы или на территории (хостинг-площадке) интернет-провайдера. Такой вариант характерен для крупных сетевых проектов. Для небольшого проекта возможно размещение сайта на сервере провайдера. Это значительно дешевле и позволяет сократить количество персонала, отвечающего за функционирование сайта.
- Бэк-офис – технический персонал и администрация компании, помещение, где она размещается и откуда осуществляется административное и техническое управление проектом.
- Служба доставки. В B2C-компаниях служба доставки играет весьма значительную роль. Чем быстрее товар будет доставлен клиенту и чем меньше неудобств он при этом испытает, тем больше вероятность того, что он совершит повторную покупку. Для различных компаний служба доставки может иметь самые разные формы. Иногда ее может вообще не быть как сколько-нибудь значимого подразделения (например, если компания оказывает консультационные или информационные услуги). В других случаях можно ограничиться курьерской службой. Встречаются ситуации, когда компания торгует товарами, имеющими значительные размеры и вес. Тогда в службе доставки может быть собственный или арендованный склад, парк автомобилей и т. п.
- Подразделение по работе с поставщиками. Как и в любой неэлектронной компании, хороший поставщик (надежный, с приемлемыми ценами) является неотъемлемой частью бизнеса. Обойтись без этого подраз-

деления могут только компании, оказывающие услуги целиком силами их собственного персонала.

- Система расчетов за товары и услуги. Заказав товар, клиент должен его оплатить. Хорошо, когда можно совместить оплату с доставкой товара. Но это не всегда возможно (например, когда товар отправляется клиенту по почте или вообще в электронном виде, если это информация или программное обеспечение). В таком случае клиент должен оплатить товар, не вступая в непосредственный контакт с персоналом компании. Конечно, можно оплатить покупку обычным банковским или почтовым переводом, но это связано с большими неудобствами, связанными, прежде всего, со значительным временем прохождения платежей. Для решения этих проблем в Интернете существуют системы электронных платежей. Формирование системы расчетов подразумевает заключение договоров с этими системами и установку на сайт программного обеспечения, позволяющего начать работу с ними.
- Маркетинговая служба. В современных условиях любой интернет-проект не может выжить без продуманной системы маркетинга. Маркетинг традиционно является одной из самых сложных составляющих сетевого бизнеса, где ошибки обходятся особенно дорого. Именно на этом направлении можно легко потерять или весьма нерационально израсходовать значительные ресурсы, чему есть немало печальных примеров. Вопросам маркетинга в Интернете посвящена вся третья часть книги.

Для промышленных предприятий наибольшее значение имеют категории электронной коммерции B2B и B2C. К категории B2B могут относиться взаимоотношения между подразделениями корпорации и между предприятиями-поставщиками и потребителями, а также межкорпоративные Интернет-биржи.

В настоящее время, говоря о B2B-рынке, часто делают основной упор на межкорпоративные торговые площадки в Интернете. Действительно, в мире значительную часть оборотов B2B-рынка полают именно они. В России тоже появились свои отраслевые электронные биржи: MetalsRussia.com (металлы), ChemForum (фармацевтика), eMatrix (компьютерная техника), Zerno OnLine (зерно, сахар, подсолнечник), Faktura (универсальная интернет-биржа), eMetex (трубы и комплектующие), Depo.ru (компьютерная техника).

Структура B2B-компании во многом схожа со структурой компании B2C.

При осуществлении электронной коммерции используются различные платежные системы. В зависимости от вида денежных средств системы расчетов в сети можно разделить на две большие группы:

- системы расчетов, работающие с реальными деньгами (расчеты наличными, банковскими и почтовыми переводами, платежные системы, работающие с банковскими счетами клиентов);
- системы расчетов, использующие электронную валюту (цифровые деньги), эмитируемую платежными системами. К этому типу относятся специализированные платежные системы в Интернете.

Платежные системы, обеспечивающие возможность расчетов в Интернете с помощью пластиковых карт (предоставляющие продавцам услуги так называемого Интернет-эквайринга), делятся на два основных типа:

1. Платежные системы, использующие для осуществления платежей только реквизиты кредитной/дебетовой карты плательщика, которые передаются через Интернет по защищенному каналу (обычно с применением протокола SSL). Такие системы просты в использовании плательщиком, но имеют ряд недостатков. Главный недостаток заключается в том, что возможен отказ клиента от платежа после получения оплаченного товара, особенно если клиент оплачивал программное обеспечение или услугу, оказываемую через Интернет. Клиент при этом ссылается на то, что списание с его счета сделано без его ведома, а продавец ничего не может доказать, так как отсутствует какой-либо подписанный клиентом документ о получении товара (оказании услуги). В такой ситуации продавец вынужден возвращать деньги клиенту;
2. Платежные системы, использующие технологию SET или другие аналогичные технологии, специально созданные для защиты платежей в Интернете. Эти технологии предусматривают выдачу каждому пользователю цифрового сертификата. Аутентификация пользователя с помощью цифровой подписи позволяет предотвратить отказы от платежей. Кроме того, информация о кредитной карте не поступает продавцу, а находится только в банке-экваере (банке, который обслуживает платежи в Интернете по кредитным картам).

Цифровые деньги – это электронный эквивалент реальных денег. Они эмитируются платежной системой и зачисляются на электронные счета клиентов в обмен на деньги, которые клиент вводит в систему банковским переводом или путем внесения наличных.

По месту хранения цифровых денег платежные системы делятся на три основных типа:

1) системы, хранящие цифровую наличность в электронном «кошельке», расположенном на компьютере клиента. В этом случае программное обеспечение платежной системы обеспечивает секретность операций клиента и невозможность несанкционированного доступа к его счету, но сами цифровые деньги хранятся на компьютере клиента, и ответственность за их сохранность лежит на нем (в случае потери данных на компьютере клиента его электронный «кошелек» не может быть восстановлен). Для подключения к такой системе клиент должен скачать с сайта системы программное обеспечение, установить его на своем компьютере и настроить в соответствии с правилами системы;
2) системы, в которых информация о состоянии электронных счетов клиентов хранится на серверах платежной системы. Для подключения к такой системе клиенту достаточно зарегистрироваться на сайте платежной системы. Иногда системы этого типа требуют от клиента скачать и установить программное обеспечение, позволяющее работать со счетом. Преимуществом таких систем для клиента является то, что потеря информации на его компьютере никак не отразится на состоянии его электронных счетов;
3) системы, в которых хранение цифровых денег и/или информации о счетах клиента осуществляется на смарт-картах (Smart-card). Смарт-карта это пластиковая карта с микросхемой. Для работы со смарт-картами и такими платежными системами пользователю необходимо иметь соответствующее периферийное оборудование, которое позволяет считывать в компьютер информацию со смарт-карт. Ведутся также работы по внедрению смарт-карт, использующих технологию SET.

Обычно физические и юридические лица работают с платежными системами по разным схемам.

Физические лица используют систему электронных «кошельков» или электронных счетов, с которых они могут оплачивать товары и услуги юридических лиц или совершать платежи друг другу.

Юридические лица, как правило, работают с платежными системами по договору комиссии, поручения или по агентскому договору. В соответствии с таким договором платежная система оказывает продавцу услуги по продаже его товаров и услуг и получает за это комиссию в виде фиксированного процента от выручки. В рамках указанного договора платежная система размещает на своем сайте, в разделе «Каталог магазинов», информацию о продавце, его товарах и услугах. Продавец, в свою очередь, раз-

мещает у себя на сайте информацию о том, что его товары и услуги можно оплатить с использованием данной платежной системы. Расчеты между платежной системой и продавцом происходят обычно один или два раза в месяц.

Пример платежной системы в Интернете – CyberPlat (*www.cyberplat.ru*), предоставляющая услуги для ведения электронной коммерции, включая обработку (процессинг) платежей и закрытый документооборот в режиме online. Расчетным банком системы является банк "Платина". К системе CyberPlat подключено более 190 Интернет-магазинов. Согласно отчету самой системы ее оборот за 2001 год составил 57,4 млн. долларов, а количество пользователей по состоянию на начало 2002 года превысило 600 тысяч. Компания продвигает на рынке Интернет-платежей следующие технологии:

- CyberCheck – защищенный документооборот по совершению сделок и их оплате, обеспечивающий обработку платежей и совершение сделок в сегменте B2B в режиме on-line;
- CyberPOS – обслуживание (эквайринг) платежных карт в Интернете. Обработка платежей в сегменте B2C в режиме on-line;
- Интернет-банкинг (банковское обслуживание клиентов через Интернет). Система предоставляет клиентам возможность управления счетами в банках-участниках системы через Интернет.

С подробным описанием и схемами этих видов услуг можно познакомиться на сайте компании.

Для защиты передаваемой информации в системе используется протокол SSL. Движение денежных средств через подсистему CyberPOS происходит только в закрытых межбанковских сетях, а реквизиты клиента известны только CyberPOS, что способствует защите банковского счета клиента от злоумышленников и недобросовестных сотрудников Интернет-магазинов. Запрос из магазина и ответ идут в шифрованном виде по стандарту выделенного сообщения (SSL) в Интернете, а сам номер карты вводится клиентом непосредственно в подсистему CyberPOS и, следовательно, становится известен только банку.

Другие отечественные платежные системы в Интернете: ASSIST (*www.ASSIST.ru*), Рапида (*www.rapida.ru*), Instant! (*www.paybot.com/ defaultrus.asp*), ЭлИТ (*www.elit.ru/card*), PayCash (*www.paycash.ru*), WebMoney Transfer (*www.webmoney.ru*), КредитПилот (*www.kreditpilot.ru*), EACCESS (*www.eaccess.ru*) и пр.

4.11 Обеспечение конфиденциальности информации в Интернете

Конфиденциальность передаваемой информации обеспечивается ее шифрованием. При помощи процедуры шифрования отправитель сообщения преобразует его из простого текста в набор символов, не поддающийся прочтению без применения специального ключа, известного получателю. Получатель сообщения, используя ключ, преобразует переданный ему набор символов обратно в текст.

Процесс преобразования с помощью ключа простого текста в зашифрованное сообщение и обратно называется алгоритмом шифрования. Обычно алгоритмы шифрования общеизвестны и не являются секретом. Конфиденциальность передачи и хранения зашифрованной информации обеспечивается за счет конфиденциальности ключа.

Ключ к шифру – конкретный набор символов и процедур, применяемых при шифровании и дешифровании сообщений. Обычно степень защищенности информации зависит не только от алгоритма шифрования, но и от длины ключа, измеряемой в битах. Чем длиннее ключ, тем лучше защита, но тем больше вычислительных процедур необходимо провести компьютеру для шифрования и дешифрования передаваемой информации, что замедляет передачу данных.

Существует два вида алгоритмов шифрования:

1) **Симметричные**. В алгоритмах этого вида и для шифрования, и для дешифрования информации применяется один и тот же секретный ключ, известный и отправителю, и получателю информации.
2) **Асимметричные**. Алгоритмы этого вида используют два ключа: один – для шифрования, другой – для дешифрования сообщения. Один из таких ключей является закрытым (секретным), другой – открытым (общедоступным).

Некоторые наиболее известные алгоритмы симметричного шифрования:

- **DES** (Data Encryption Standard). Разработан фирмой IBM и широко используется с 1977 года. В настоящее время несколько устарел, поскольку применяемая в нем длина ключа недостаточна для обеспечения устойчивости к вскрытию методом полного перебора всех возможных значений ключа;
- **Triple DES**. Это усовершенствованный вариант DES, применяющий для шифрования алгоритм DES три раза с разными ключами. Он значительно устойчивее к взлому, чем DES;

- **Rijndael**. Алгоритм разработан в Бельгии. Работает с ключами длиной 128, 192 и 256 бит. На данный момент к нему нет претензий у специалистов по криптографии;
- **Skipjack**. Алгоритм создан и используется Агентством национальной безопасности США. Длина ключа 80 бит. Шифрование и дешифрование информации производится циклически (32 цикла);
- **IDEA**. Алгоритм запатентован в США и ряде европейских стран. Держатель патента компания Ascom-Tech. Алгоритм использует циклическую обработку информации (8 циклов) путем применения к ней ряда математических операций;
- **RC4**. Алгоритм специально разработан для быстрого шифрования больших объемов информации. Он использует ключ переменной длины (в зависимости от необходимой степени защиты информации) и работает значительно быстрее других алгоритмов. RC4 относится к так называемым потоковым шифрам.

Наиболее распространенные алгоритмы асимметричного шифрования:

- **RCA**. Алгоритм разработан в 1977 году и использует открытые ключи, обеспечивающие преобразование информации «только в одну сторону» (шифрование) за счет сложности решения задачи факторизации (разложения на множители) больших чисел;
- **ECC** (Elliptic Curve Cryptography). Однонаправленность преобразований (шифрования) обеспечивается в этом методе сложностью математических вычислений, связанных с эллиптическими кривыми.

В настоящее время в системах документооборота широко используется электронная цифровая подпись (**ЭЦП**), которая является электронным эквивалентом собственноручной подписи. **ЭЦП** служит не только для аутентификации отправителя сообщения, но и для проверки его целостности.

При использовании **ЭЦП** для аутентификации отправителя сообщения применяются открытый и закрытый ключи. Процедура похожа на осуществляемую в асимметричном шифровании, но в данном случае закрытый ключ служит для шифрования, а открытый – для дешифрования.

Алгоритм применения **ЭЦП** состоит из ряда операций:

1) генерируется пара ключей: открытый и закрытый;
2) открытый ключ передается заинтересованной стороне (получателю документов, подписанных стороной, сгенерировавшей ключи);
3) отправитель сообщения шифрует его своим закрытым ключом и передает получателю по каналам связи;

4) получатель дешифрует сообщение открытым ключом отправителя.

Создать зашифрованное сообщение, при расшифровке которого открытым ключом получается исходный текст, может только обладатель закрытого ключа, т. е. отправитель сообщения. Использовать для этого открытый ключ невозможно.

Для защиты данных, передаваемых через Интернет, используется протокол **SSL** (Secure Socket Layer). Этот протокол основан на комбинации алгоритмов асимметричного и симметричного шифрования.

Протокол может работать в трех режимах:

1) при взаимной аутентификации сторон;
2) при аутентификации сервера и анонимности клиента;
3) при взаимной анонимности сторон.

При установлении соединения по протоколу SSL для данной сессии связи генерируется разовый ключ, который служит для симметричного шифрования данных, передаваемых в течение данной сессии. Разовый ключ генерируется на этапе установления соединения. При этом используются асимметричные алгоритмы шифрования.

Технология SET (Secure Electronic Transactions), используемая в системах электронной коммерции, появилась в 1996 году. Ее основными разработчиками были MasterCard International и Visa International.

SET предусматривает использование цифровых сертификатов всеми участниками сделки, что позволяет проводить их однозначную взаимную аутентификацию.

Технология **SET** направлена на организацию максимально защищенных транзакций с использованием кредитных карт.

Взаимная аутентификация сторон и использование **ЭЦП** позволяют избежать проблем с отказами сторон от обязательств по сделкам и полностью закрыть проблему необоснованного отзыва плательщиками своих платежей.

В основе процедур защиты информации, используемых **SET**, лежат технологии шифрования RSA и DES, что обеспечивает высокий уровень безопасности.

В общем случае алгоритм взаимодействия участников сделки по технологии **SET** выглядит следующим образом:

1) прежде чем начать работу с использованием **SET** все участники сделки получают цифровые сертификаты у сертифицирующей организации (например, VeriSign (*www.verisign.com*) или Thawte (*www.thawte.com*)). Таким образом, устанавливается однозначное соответствие между участником и его ЭЦП;

2) посетив сайт продавца, покупатель оформляет заказ и указывает способ оплаты при помощи кредитной карты;
3) покупатель и продавец предъявляют друг другу свои сертификаты;
4) продавец инициирует проверку платежной системой предоставленной клиентом информации. Платежная система передает продавцу результаты проверки;
5) при положительных результатах проверки по запросу продавца совершается перечисление денег.

4.12 Основы создания Web-страниц

Разработка Web-страниц может вестись в различных программных средствах: начиная от Блокнота, простейшего текстового редактора Windows, приложения Microsoft Office Word с режимом сохранения «веб-страница», и до специализированных средств, таких, как Microsoft Office FrontPage, Dreamweaver, Adobe GoLive, Serif WebPlus и многих других.

Для просмотра Web-страниц, как было сказано ранее, используются Интернет-обозреватели, среди которых лидирующее положение занимает Internet Explorer – стандартная программа Windows. Интернет-обозреватели отображают Web-страницы в соответствии с правилами их формирования, описанными в стандарте гипертекстовых документов HTML 4.01. Однако, они могут показывать и простые текстовые файлы без специальной разметки, в чем нетрудно убедиться. Для текстового файла можно выбрать команду «открыть с помощью… Internet Explorer» и мы увидим простой текст, без эффектных стилей оформления, которые обычно присутствуют на Web-страницах.

Язык HTML (HyperText Markup Language – язык гипертекстовой разметки) позволяет сформировать структурированный текстовый файл, состоящий из отдельных фрагментов, начало и конец которых обозначается с использованием так называемых тегов. Теги используются для задания форматирования или назначения тех или иных элементов Web-страницы. Особые теги используются для размещения на Web-страницах графических изображений, аудио- и видеоклипов и прочих так называемых внедренных объектов.

Теги пишутся в угловых скобках < > (т. е. между знаками меньше и больше), они бывают парными и одиночными. Теги могут иметь атрибуты. Набор возможных тегов определен в стандарте HTML 4.01. В стандарте HTML 4.0 впервые появилась возможность использования таблиц стилей для конкретного элемента или для группы элементов. Стилевая информа-

ция может быть установлена в документе HTML или во внешней таблице стилей. В текст Web-страниц могут включаться сценарии (скрипты) на языках VBScript, JavaScript, TCL (Tool Command Language) для создания динамических страниц и использования HTML как средства создания сетевых приложений. Приведем простейший пример HTML-документа, построенного с использованием тегов:

```
<HTML>
   <HEAD>
      <TITLE>Надпись на заголовке окна</TITLE>
   </HEAD>
   <BODY>
     <H1>Заголовок 1</H1>
     <H2>Заголовок 2</H2>
      <P>Текст основной части страницы
      <P>Второй абзац основной части страницы
   </BODY>
</HTML>
```

В данном примере документ состоит из раздела-заголовка между тегами **<HEAD>** и **</HEAD>** и тела – основной части Web-страницы между тегами **<BODY>** и **</BODY>**. Тело в этом примере состоит из фрагментов двух заголовков (теги **<H1>** и **</H1>**, **<H2>** и **</H2>**,) и двух параграфов (тег **<P>**).

Элемент **HEAD** содержит информацию о документе, такую как название **TITLE** (отображается в области заголовка окна), ключевые слова, которые могут оказаться полезными при использовании машин поиска, и другие данные, не являющиеся содержимым документа и не отображаемые на основной странице документа.

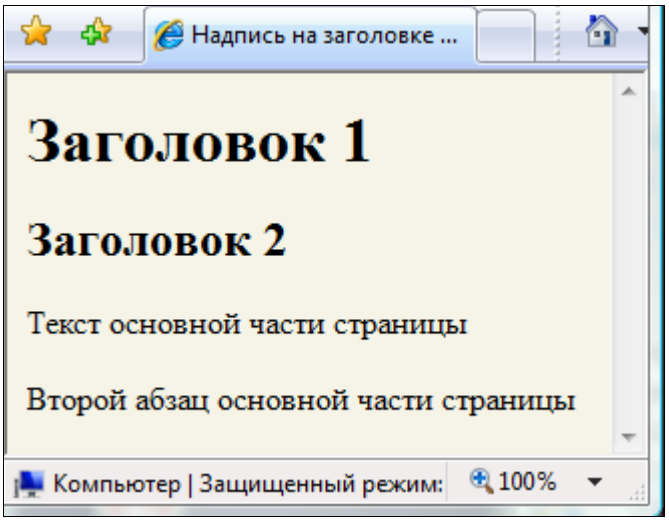


Рисунок 4.13. Пример простейшего HTML-документа

Если текст примера записать в файл с расширением *.html и открыть его, документ будет иметь вид, показанный на рисунке 4.13.

Корректный документ HTML 4 состоит из трех частей:

1) строка, содержащая информацию о версии HTML;

2) объявляющий раздел (тег HEAD);
3) тело, содержащее собственно сам документ (теги BODY или FRAMESET).

Таким образом, для полноценного функционирования предыдущего примера его нужно дополнить информацией о версии HTML, как это показано в следующем примере.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
   "http://www.w3.org/TR/html4/strict.dtd">
<HTML>
<HEAD>
   <TITLE>RU-CENTER Центр регистрации доменов</TITLE>
</HEAD>
<BODY>
 <table
  <tr>
   <td width=110
    <P style="font-size: 18pt; color: Green;
        font-weight: bold;"
    <p><a href="http://www.nic.ru/">
       <img src=RU_center.png
       alt="Эмблема RU-CENTER"></a> </p> </td>
   <td width=500
    <TD style="font-size: 18pt; color: Green;
        font-weight: bold;"
    <p>(размер 18, зеленый, жирный) </P>
    <p>RU-CENTER - аккредитованный Регистратор: </P>
   </td> </tr>
 </table>
 <UL>
  <LI>
   <P style="font-size: 14pt;
      color: red;font-style: italic;"
   <P>в домене.RU -
     <A href="http://www.cctld.ru/ru/regru/">
      Координационным центром национального домена
      Сети Интернет</A> </P>
   <LI>
     <P>в домене.SU -
     <A href="http://www.fid.su/projects/Registrars/">
        Фондом Развития Интернет</A> </P>
   <LI>
      <P>в доменах NET, COM, ORG, BIZ, INFO -
```

```
            <A href=http://www.icann.org/registrars/>
            ICANN (Интернет корпорация по распределению
            доменных имен и адресного пространства)</A>
        </P></LI></UL>
    </BODY>
</HTML>
```

Во втором примере информация размещена в таблице, состоящей из двух ячеек в одной строке. В первой ячейке находится ссылка на адрес **http://www.nic.ru/**, которая срабатывает при щелчке на рисунке из файла **RU_center.png**. Во второй ячейке таблицы присутствуют два абзаца, для которых заданы параметры шрифта с использованием внедренных стилей. Ниже таблицы оформлен маркированный список со ссылками на Web-сайты.

Вид второго примера в Web-браузере показан на рисунке 4.14.

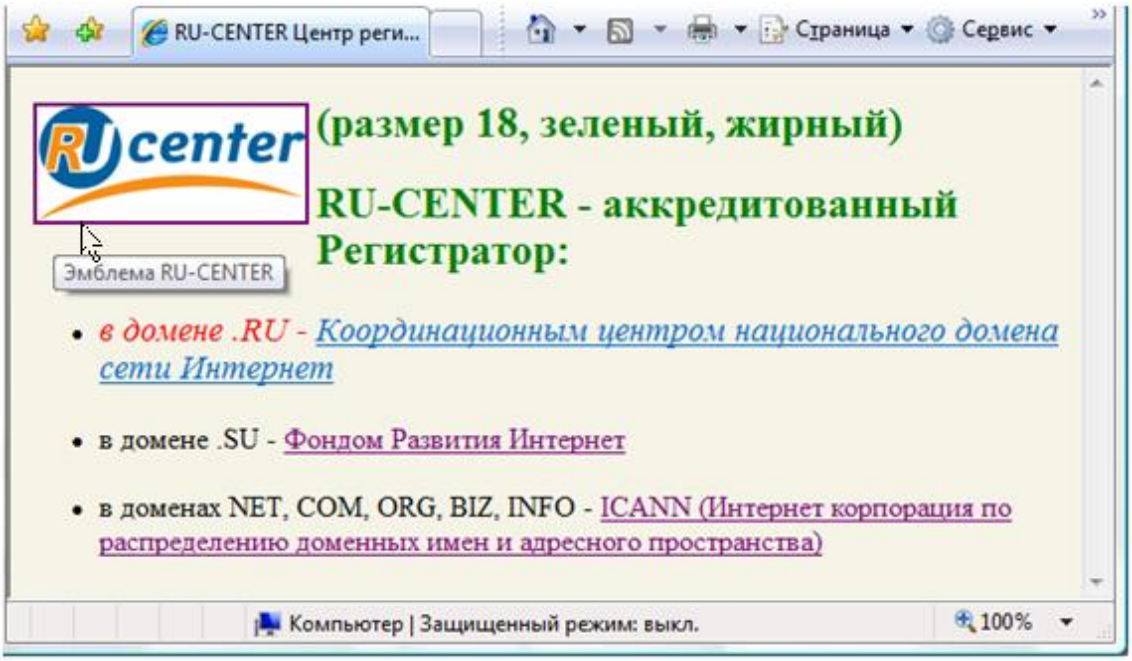


Рисунок 4.14. Пример 2 HTML-документа

Впрочем, знать в деталях стандарт HTML 4.01, его новую версию HTML 5 или новую спецификацию XHTML 2.0 Web-дизайнерам не обязательно – разработка html-страниц ведется обычно в визуальном режиме. Простой пример – документ, подготовленный в Microsoft Office Word с использованием всех возможностей этой системы, сохраненный в формате «Веб-страница, *.html», реализует воспроизведение этого документа в Web-браузере с соблюдением всех требований стандарта HTML 4 с использованием внедренных таблиц стилей CSS и xml-кода. Однако, Web-

страница, созданная таким образом, содержит очень большой объем описаний параметров шрифтов и стилей абзацев, предопределенных по умолчанию для стандартного шаблона документа (Normal.dot). Поэтому для разработки Web-страниц лучше использовать специализированные средства, например, Microsoft Office FrontPage.

Web-страницы могут быть связаны с программами–скриптами, которые загружаются на клиентский компьютер вместе с документом HTML или присутствуют в нем. Программа выполняется на клиентской машине при загрузке документа или в другое время, когда, например, активируется гиперссылка.

Скрипты позволяют авторам расширить возможности документов HTML – создавать активные или интерактивные страницы. Скрипты могут выполняться по мере загрузки документа для динамической модификации содержимого этого документа, могут обрабатывать ввод данных пользователем, могут запускаться событиями, происходящими в документе: передача фокуса элементу, движение мыши, закрытие документа и т. п., могут быть связаны с элементами управления (например, кнопками, списками и пр.).

Контрольные вопросы к главе 4

1) Что такое Интернет и Web (WWW)?
2) Какие организации осуществляют перспективное и текущее планирование функционирования Интернета?
3) Как подразделяются операторы Интернет-доступа по функционально-географическому признаку?
4) Какие способы используются для адресации компьютеров в Интернете?
5) Какие используются способы подключения к Интернету конечных пользователей?
6) Какие поисковые системы Интернета наиболее известны?
7) Какие типы электронной почты используютя?
8) Что такое RSS-канал?
9) Какие средства существуют для общения в Интернете в реальном времени?
10) Что такое Интернет-радио и Интернет-телевидение?
11) Какие существуют категории электронной коммерции?
12) Какие методы используются для обеспечения конфиденциальности информации в Интернете?
13) Что такое язык HTML?

Глава 5. Основы разработки программного обеспечения

Задача данной главы – дать общие представления о разработке программ для персональных компьютеров, не касаясь сложных аспектов, таких, как объектно-ориентированное программирование, использование библиотек Windows API, MFC, ATL, COM (ActiveX), FCL и прочих профессиональных инструментов и технологий.

Современная библиотека классов .NET Framework (FCL), используемая разработчиками программного обеспечения, содержит более 7000 типов: классов объектов, структур, интерфейсов и пр. Многие классы FCL содержат более 100 методов, свойств и других компонентов, так что изучить FCL нелегко даже профессиональным программистам.

Средства разработки ПО начинались с простейших трансляторов (интерпретаторов и компиляторов – программ для перевода текстовой программы на алгоритмическом языке в двоичный машинный код), затем появились средства разработчиков типа системы Turbo Pascal. В настоящее время лидирующее положение занимает система Microsoft Visual Studio – интегрированная визуальная среда разработки для создания сложных Windows- и Web-приложений с использованием технологии.NET (dot NET) с одновременным использованием трех основных алгоритмических языков: Visual C#, Visual Basic, Visual C++. В то же время в области программирования для Интернета большую роль играет язык Java и скриптовые языки: язык серверных скриптов PHP и языки клиентских скриптов JavaScript, VBScript.

Технологии и стандарты разработки сложных программных систем тоже постоянно совершенствовались: от принципов структурного программирования до стандартов **UML** – языка для спецификации, визуализации, конструирования и документирования сложных информационно-насыщенных объектных систем и **UEML –** унифицированного языка моделирования систем масштаба предприятия.

В нашей стране процесс разработки программного обеспечения регламентируется группой стандартов по информационным технологиям (ГОСТ 34.×××).

5.1 Проектирование программного обеспечения

Разработка программного обеспечения крупных информационных систем ведется большими коллективами специалистов (постановщики задач, программисты, администраторы баз данных, тестеры, документаторы и пр.), для согласованной работы которых необходимо использование специальных методологий.

К международным стандартам разработки **ИС** можно отнести методологии семейства **IDEF** (Icam DEFinition, ICAM – Integrated Computer-Aided Manufacturing – первоначально разработанная в конце 70-х гг. программа ВВС США интегрированной компьютерной поддержки производства). С их помощью можно эффективно проектировать, отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах (инжиниринг и реинжиниринг). В настоящий момент к семейству **IDEF** относятся стандарты **IDEF0** – **IDEF14**, наиболее важными при разработке **ИС** являются:

- **IDEF0** – Function Modeling – методология функционального моделирования информационных систем. С помощью наглядного графического языка IDEF0 система представляется в виде набора взаимосвязанных функциональных блоков. Основана на разработанной компанией SofTech, Inc. в конце 60-х гг. технологии **SADT** - структурированного анализа и разработки (Structured Analysis and Design Technique). Является первым этапом изучения, проектирования и модернизации любой системы;
- **IDEF1** – Information Modeling – методология моделирования и анализа информационных потоков предприятия. В основе лежит метод Сущность-Связь-Ключ-Атрибут (Entity-Link-Key-Attribute – ELKA) и построение диаграмм Сущность-Связь (ERD – Entity-Relationship Diagram);
- **IDEF1X** (IDEF1 Extended) – Data Modeling – методология проектирования реляционных баз данных. Заключается в построении диаграмм «Сущность-Связь» в нотации этого стандарта;
- **IDEF2** – Simulation Model Design – методология динамического моделирования систем. В настоящее время существуют алгоритмы и их компьютерные реализации, позволяющие превращать набор статических диаграмм IDEF0 в динамические модели, построенные на базе «раскрашенных сетей Петри» (CPN – Color Petri Nets);
- **IDEF3** – Process Description Capture – методология документирования процессов, происходящих в системе, которая используется при исследовании технологических процессов на предприятиях. IDEF3 имеет прямую взаимосвязь с методологией IDEF0 – каждая функция (функциональный

блок) может быть представлена в виде отдельного процесса средствами IDEF3;

- **IDEF4** – Object-Oriented Design – методология построения объектно-ориентированных систем. Средства IDEF4 позволяют наглядно отображать структуру объектов и заложенные принципы их взаимодействия, позволяя анализировать и оптимизировать сложные объектно-ориентированные системы;
- **IDEF14** – Network Design – методология моделирование компьютерных сетей. Позволяет выполнять представление и анализ данных при проектировании вычислительных сетей на графическом языке с описанием конфигураций, очередей, сетевых компонентов, требований к надежности и т.п.

Описание стандартов **IDEF0 – IDEF5, IDEF9** можно найти на сайтах *http://www.idef.ru*, *http://www.idef.com* или в интернет-библиотеке **Верникова** (*http://www.vpg.ru/main.mhtml?PubID=6*).

В стандарте **IDEF0** функциональный блок графически изображается в виде прямоугольника (см. рисунок 5.1) и олицетворяет собой некоторую конкретную функцию в рамках рассматриваемой системы.

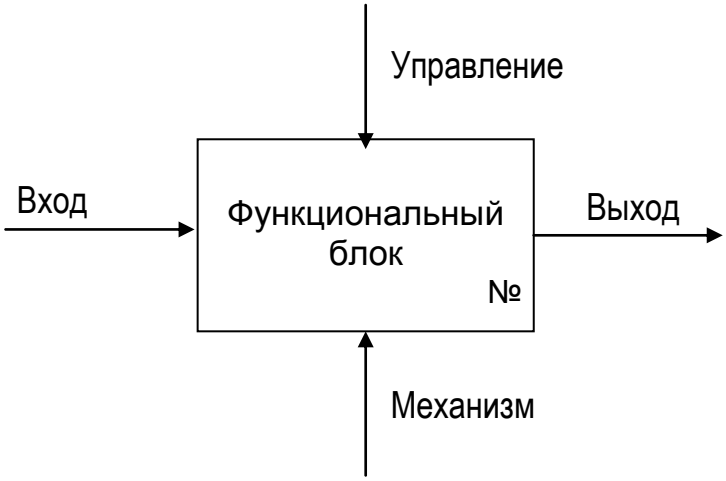


Рисунок 5.1. Функциональный блок по стандарту **IDEF0**

По требованиям стандарта название каждого функционального блока должно быть сформулировано в глагольном наклонении («выполнять операцию», а не «выполнение операции»). Каждая из четырех сторон функционального блока имеет свое определенное значение (роль), при этом:

- верхняя сторона имеет значение «Управление» (Control);
- левая сторона имеет значение «Вход» (Input);
- правая сторона имеет значение «Выход» (Output);
- нижняя сторона имеет значение «Механизм» (Mechanism).

Принципы изображения функциональных моделей стандарта **IDEF0** используют инструментальные средства проектирования **ИС** (**CASE-средства –** Computer-Aided Software System Engineering), такие, как **BPwin** фирмы Computer Associates и др. Применение CASE-средств целесообразно при разработке сложных **ИС** коллективом исполнителей, когда ставится задача повысить производительность труда, улучшить качество программных продуктов, обеспечить согласованное взаимодействие разрабатываемых модулей системы.

Метод **IDEF1**, являясь методом анализа, описывает:

- как информация собирается, хранится и обрабатывается на предприятии;
- правила и логику управления информацией.

В стандарте **IDEF1** сущности и связи концептуальной модели изображаются, как показано на рисунок 5.2.

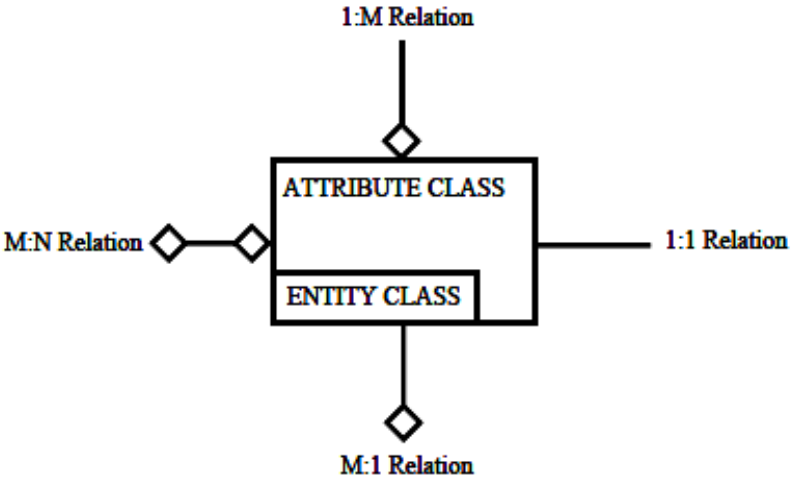


Рисунок 5.2. Изображение сущности и связей концептуальной модели по **IDEF1**

Важное место в моделировании информационных систем занимает методология **UML** – унифицированный язык моделирования (**Unified Modeling Language**).

UML – язык для спецификации, визуализации, конструирования и документирования сложных информационно-насыщенных объектных систем. В настоящее время зарегистрирован, как международный стандарт **ISO/IEC 19501:2005** «Information technology – Open Distributed Processing – Unified Modeling Language (**UML**)». Инструментальные средства, поддерживающие методологию **UML – Rational Rose** (Rational Software), **Paradigm Plus** (CA/Platinum), **ARIS** (IDS Sheer AG), **Together Designer** (Borland) и др.

В феврале 2006 г. Object Management Group, Inc. (OMG, http://www.omg.org) – международный консорциум по разработке спецификаций в компьютерной индустрии, опубликовал финальную версию языка моделирования бизнес-процессов **BPML** (Business Process Modeling Language - *http://www.omg.org/cgi-bin/doc?dtc/2006-02-01*).

Основные графические элементы языка показаны на рисунке 5.3.

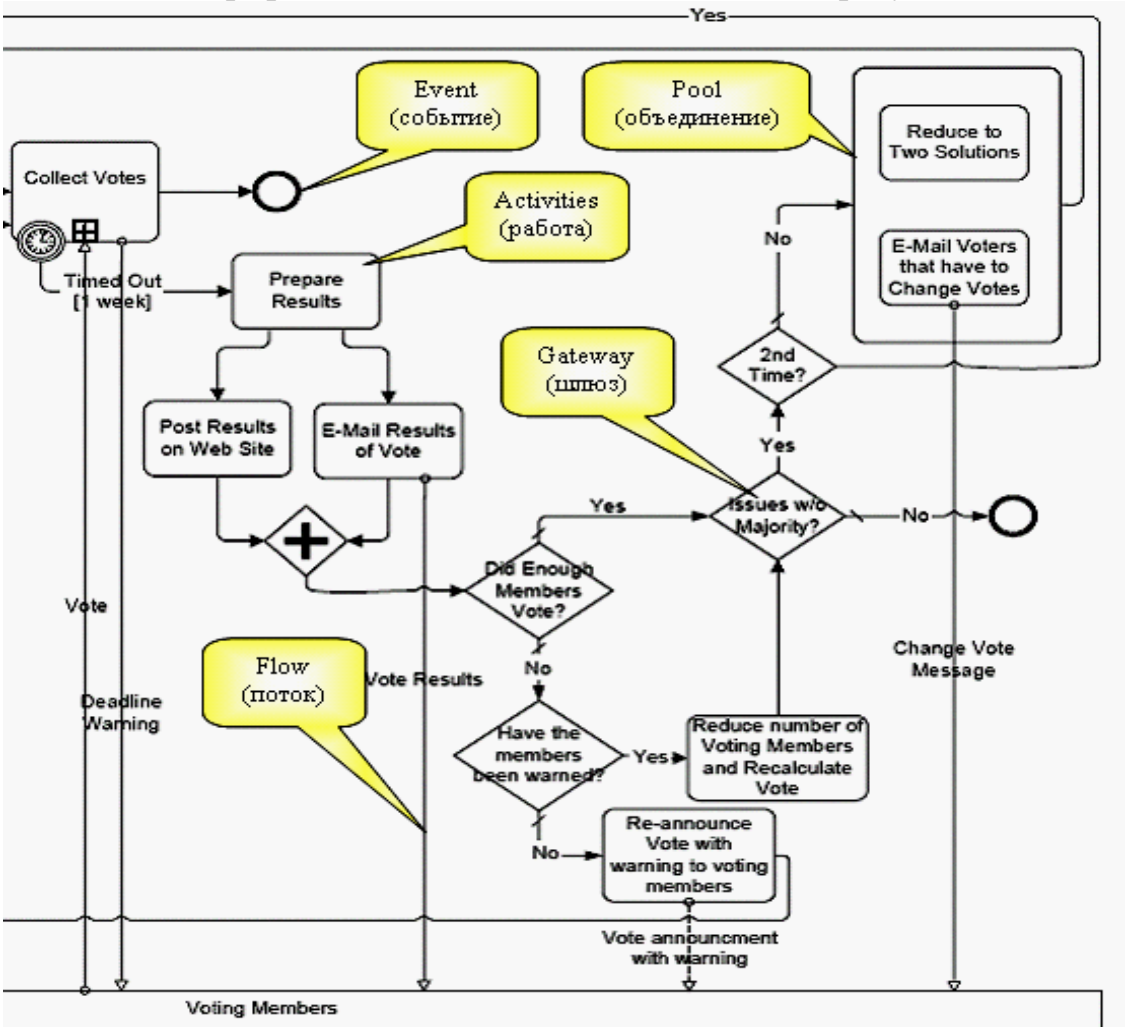


Рисунок 5.3. Фрагмент модели бизнес-процесса по стандарту **BPML**

Одной из последних разработок в области моделирования предприятия является создание специального унифицированного языка моделирования **UEML** (Unified Enterprise Modeling Language). Разработка **UEML** – сетевой проект (IST–2001–34229), финансируемый Евросоюзом (см. *http://athena.troux.com/akmii/Default.aspx?WebID=249*).

Проект **UEML** включает разработку:

- общего, визуального, базированного на шаблонах языка для коммерческих инструментальных средств моделирования предприятий и программных систем класса workflow;
- стандартизованных, независимых от инструментов механизмов передачи моделей между проектами;
- репозитория моделей предприятий.

Данный проект осуществляется в соответствии с международными стандартами:

- **ISO 14258** Rules and Guidelines for Enterprise Models (Правила и руководящие принципы для моделей предприятия);
- **ISO 15704** Requirements for enterprise-reference architectures and methodologies (Требования и методологии по описанию архитектуры предприятия).

Инструментальные средства моделирования предприятий, поддерживающие язык **UEML** – Metis (Computas), e-MAGIM (GraiSoft), MO²GO (IPK) и др.

Другие методологии, используемые при моделировании **ИС**:

- **DFD** – технология анализа «потока данных» (Data Flow Diagrams);
- **Workflow** – технология анализа «потока работ».

В нашей стране в списке действующих ГОСТов по разработке автоматизированных (программных) систем (по данным Стандартинформ *http://www.vniiki.ru/catalog_v.asp?page=1*) следующие:

- ГОСТ 34.003-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Термины и определения
- ГОСТ 34.201-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем
- ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания
- ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.

Стадии и этапы создания автоматизированных систем (АС) в соответствии с **ГОСТ 34.601-90** приведены в таблице 5.1. В приложении к этому стандарту подробно описывается содержание работ по всем этапам.

Стадия 3 «Техническое задание» разрабатывается в соответствии с требованиями *ГОСТ 34.602*.

Таблица 5.1. Стадии и этапы создания АС по ГОСТ 34.601-90

Стадии	Этапы работ
1. Формирование требований к АС	1.1. Обследование объекта и обоснование необходимости создания АС 1.2. Формирование требований пользователя к АС 1.3. Оформление отчёта о выполненной работе и заявки на разработку АС (тактико-технического задания)
2. Разработка концепции АС	2.1. Изучение объекта 2.2. Проведение необходимых научно-исследовательских работ 2.3. Разработка вариантов концепции АС, удовлетворяющего требованиям пользователя 2.4. Оформление отчёта о выполненной работе
3. Техническое задание	3.1. Разработка и утверждение технического задания на создание АС
4. Эскизный проект	4.1. Разработка предварительных проектных решений по системе и её частям 4.2. Разработка документации на АС и ее части
5. Технический проект	5.1. Разработка проектных решений по системе и её частям 5.2. Разработка документации на АС и ее части 5.3. Разработка и оформление документации на поставку изделий для комплектования АС и (или) технических требований (технических заданий) на их разработку 5.4. Разработка заданий на проектирование в смежных частях проекта объекта автоматизации
6. Рабочая документация	6.1. Разработка рабочей документации на систему и её части. 6.2. Разработка или адаптация программ.
7. Ввод в действие	7.1. Подготовка объекта автоматизации к вводу АС в действие 7.2. Подготовка персонала 7.3. Комплектация АС поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями) 7.4. Строительно-монтажные работы 7.5. Пусконаладочные работы 7.6. Проведение предварительных испытаний 7.7. Проведение опытной эксплуатации 7.8. Проведение приёмочных испытаний

Продолжение таблицы 5.1

Стадии	Этапы работ
8. Сопровождение	8.1. Выполнение работ в соответствии с гарантийными обязательствами 8.2. Послегарантийное обслуживание

5.1 Классификация языков программирования

Язык программирования – система синтаксических правил для написания команд, из которых состоит программа, непосредственно исполняемая на компьютере (язык низкого уровня) или исполняемая на компьютере после преобразования (трансляции) в исполняемую программу (язык высокого уровня).

Языки низкого уровня относятся к машинно-ориентированным языкам, т. к. используют возможности определенного вида процессоров, например, процессоров Intel Core 2 Duo, AMD Athlon, Intel Xeon, и пр.

К языкам низкого уровня относятся:

- программирование в машинных кодах;
- ассемблер;
- макроассемблер.

Как выглядит программа в машинных кодах можно увидеть, если открыть любой исполняемый файл в Блокноте (двоичные коды будут показаны, как символы ASCII) или в Hex-редакторе (коды в шестнадцатеричном виде – см. рисунок 5.6).

```
0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68 ;
69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F ;
74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20 ;
6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00 ;
A7 DC F4 A6 E3 BD 9A F5 E3 BD 9A F5 E3 BD 9A F5 ;
```

Рисунок 5.4. Фрагмент программы в машинных кодах (шестнадцатеричный код)

Конечно, ни одну сложную программу в наше время никто не будет писать на языке машинного кода. Да и Ассемблер, который представляет собой мнемоничесую запись тех же машинных команд, используется системными программистами достаточно редко. Фрагмент того же кода в машинных командах и на Ассемблере будет выглядеть следующим образом:

Машинный код	Ассемблер
0E	PUSH CS
1F	POP DS
BA0E00	MOV DX,000E
B409	MOV AH,09
CD21	INT 21
B8014C	MOV AX,4C01
CD21	INT 21
54	PUSH SP
68	DB 68
69	DB 69
7320	JNB 0033
7072	JO 0087
6F	DB 6F
67	DB 67
7261	JB 007A
6D	DB 6D
206361	AND [BP+DI+61],AH
6E	DB 6E
6E	DB 6E
6F	DB 6F
и т.д.	

Мнемокод Ассемблера использует буквенные обозначения машинных операций (**PUSH, POP, MOV** и т. д.) и регистров компьютера (**CS, DS, DX, AH** и т. д.). Регистры – специализированные 16-, 32- или 64-разрядные ячейки памяти процессора, которые используются для управления исполнением машинных команд, адресации и выполнения математических операций.

Процесс трансляции программы с языка Ассемблера в машинный код называется ассемблированием. Обратный процесс – дизасемблированием.

Современный макроассемблер для Windows (например, Masm32) занимает промежуточное положение между Ассемблером и языками высокого уровня: он использует команды обычного Ассемблера и множество системных и пользовательских процедур, описания которых присутствует во включаемых файлах (*.inc) и объектных библиотеках (*.lib), генерируемых из системных библиотек при установке Masm32 в операционной системе.

Пример простейшей программы на Masm32, создающей диалоговое окно, показанное на рисунке 5.5:

```
.386
.model flat, stdcall
option casemap :none
include \masm32\include\windows.inc
```

```
include \masm32\include\user32.inc
include \masm32\include\kernel32.inc
includelib \masm32\lib\user32.lib
includelib \masm32\lib\kernel32.lib
.code
start:
 jmp @F
    Titl  db "Masm32",0
    Msg  db  "Это Ассемблер!",0
  @@:
  invoke MessageBox,0,ADDR Msg,ADDR Titl,MB_OK
  invoke ExitProcess,0
end start
```

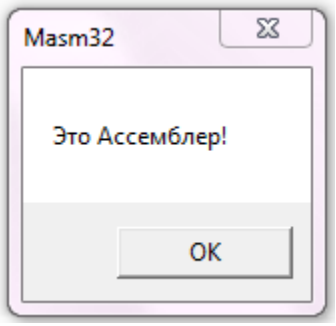


Рисунок 5.5. Окно, созданное программой на Masm32

Языки высокого уровня разрабатывались с целью повышения производительности труда программистов за счет использования команд (операторов), использующих слова английского языка, соответствующих последовательности из многих машинных инструкций.

Например, фрагмент текста программы на языке ***Pascal*** может выглядеть следующим образом:

```
Assign(fp,'workers.fdt');  {назначить для файловой переменной fp }
                           {файл с именем 'workers.fdt'}
Reset(fp);                 {открыть файл}
Seek(fp,FileSize(fp)-1);   {перейти на последнюю запись}
Read(fp,worker);           {прочитать запись в переменную worker}
Writeln(worker.fio);       {показать на экране поле fio записи}
Close(fp);                 {закрыть файл}
```

Для выполнения на компьютере программ, написанных на языке высокого уровня, необходимо выполнить их трансляцию (компиляцию или интерпретацию) в машинные коды или компиляцию в промежуточный двоичный код (байт-код), который далее интерпретируется в различных операционных системах. Для этого служат специальные программы – компиляторы и интерпретаторы.

Рисунок 5.6 демонстрирует различные методы исполнения программ процессором ПК в ОС Windows.

Все языки высокого уровня можно разделить на универсальные и проблемно-ориентированные.

Наибольшее значение и развитие получили универсальные языки: C#, C++, Basic, Pascal (Delphi), используемые в настоящее время для разработки Windows-приложений. На начальных этапах развития программирования большую роль сыграли также языки Fortran, PL/1, Cobol, Algol, C и многие другие.

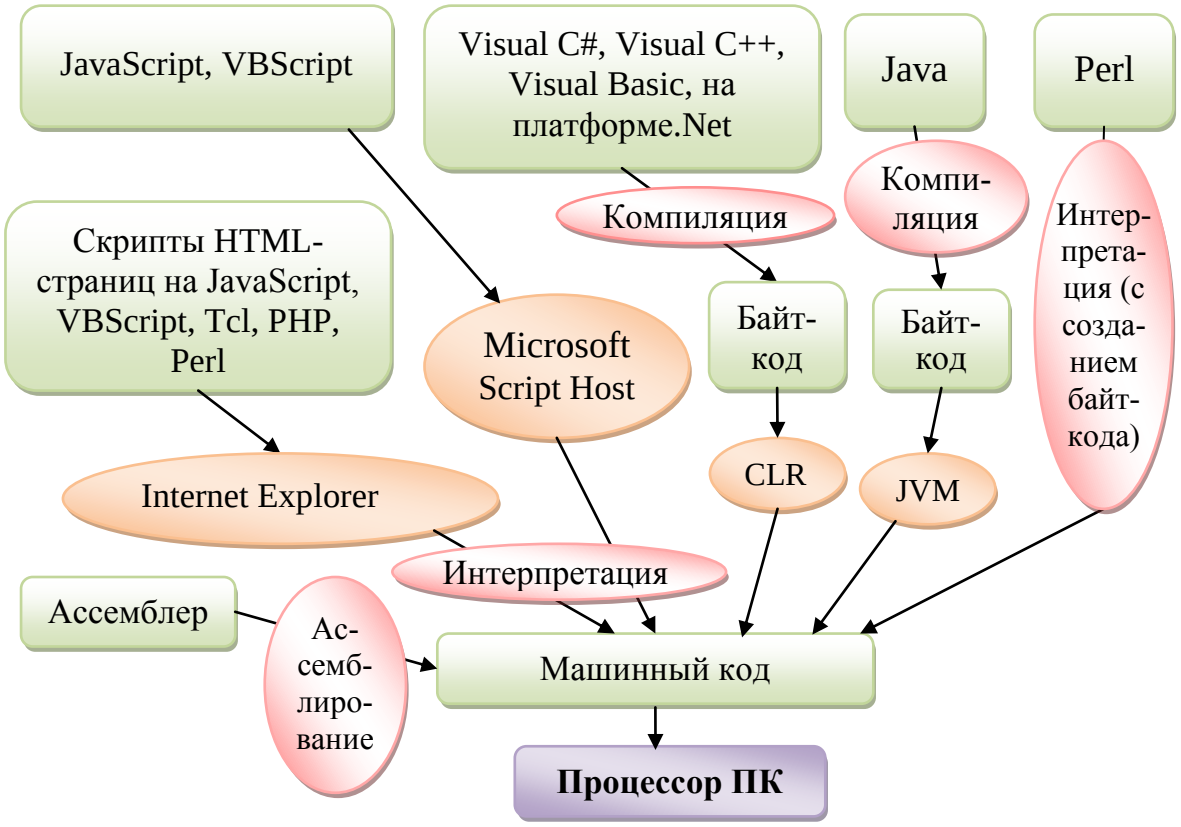


Рисунок 5.6. Методы исполнения программ в ОС Windows

В связи с повсеместным распространением Интернета, Интранета и Экстранета и Интернет-браузеров, как основы пользовательского интерфейса в любых задачах, решаемых на ПК (в том числе и в ERP-системах), большое значение приобрели языки программирования Интернет-приложений, которые скорее можно отнести к универсальным языкам, чем к проблемно-ориентированным. К таким языкам относятся современные версии языков C#, J#, Basic, из системы Microsoft Visual Studio.

Большое значение имеют проблемно-ориентированные языки, используемые на Интернет-серверах (PHP, Perl) и клиентских Интернет-приложениях (JavaScript, VBScript, Tcl).

К проблемно-ориентированным языкам программирования задач искусственного интеллекта относятся языки Lisp, Prolog, Multilisp, Commonlisp, Рефал, Planner и пр.

В Интернете можно найти рейтинг языков программирования. Например, на сайте компании TIOBE Software *http://www.tiobe.com/* ежемесячно рассчитывает глобальный рейтинг 50-ти языков программирования и отслеживается динамика его изменения. Рейтинг основан на расчете коли-

чества разработчиков, использующих различные языки по всему миру, количества различных курсов и производителей, использующих те или иные языки. Для расчета рейтинга используются поисковые системы Google, MSN, Yahoo! и YouTube. В таблице 5.2 показан рейтинг для первой двадцатки языков за декабрь 2009 года.

Таблица 5.2. TIOBE Programming Community Index

Язык программирования	Категория языка	Позиция		Изменение позиции	Рейтинг, %	Δ, %	Статус
		дек. 09	дек. 08		дек. 09	дек. 08	
Java	ОО	1	1	0	17,06	-2,31	A
C	П	2	2	0	16,28	+0,12	A
PHP	ОО, П	3	4	1	9,77	+0.29	A
C++	ОО	4	3	-1	9,18	-1,72	A
(Visual) Basic	ОО, П	5	5	0	7,78	-1,70	A
C#	ОО	6	6	0	6,26	+1,61	A
Python	ОО	7	7	0	5,2	+0.62	A
JavaScript	ОО, П, Ф	8	9	1	3,52	+0,45	A
Perl	П	9	8	-1	2,69	-0. 91	A
Ruby	ОО	10	11	1	2,65	+0,34	A
Delphi	ОО	11	10	-1	2,30	-0,75	A
PL/SQL	П, Л	12	13	+1	1,49	+0,35	A
Objective-C	ОО	13	35	+22	1,16	+1,00	A
SAS	П	14	14	0	0,9	+0,07	A
Lisp/Scheme	Ф	15	19	+4	0,88	+0,37	A-
ABAP	П	16	17	+1	0,72	+0,12	A-
Pascal	П	17	15	-2	0,70	-0.01	B
ActionScript	ОО	18	21	+3	0,66	+0,17	B
D	ОО	19	12	-7	0,59	-0,60	A
Lua	П, Ф	20	20	0	0,56	+0,09	B

Статус в таблице 5.2 – принадлежность языка программирования к группе основных используемых (A) или второстепенных (B). A- означает сдвиг в сторону второстепенных.

Категория языка в таблице:

- ОО – объектно-ориентированный;
- П – процедурный;
- Ф – функциональный;
- Л – логический.

Распределение языков по категориям по данным компании TIOBE Software представлена в таблице 5.3.

Таблица 5.3. Рейтинг категорий языков

Категория языков	Рейтинг, дек. 2008	Δ, дек. 2007
Объектно-ориентированные	55.7%	+1.2%
Процедурные	40.4%	+0.1%
Функциональные	2.8%	-0.7%
Логические	1.2%	-0.6%

На сайте компании TIOBE Software приведена динамика изменения рейтинга языков таблицы 5.2. На рисунке 5.7 показаны две диаграммы за период 2001 – 2010 годы – для языка C#, который считается наиболее перспективным языком и популярность которого неуклонно растет, и языка C++, который до недавнего времени считался основным языком профессиональных программистов для широкого круга задач, но рейтинг его падает. В то же время, значение рейтинга C# в 2010 г. около 6%, а C++ – 9%.

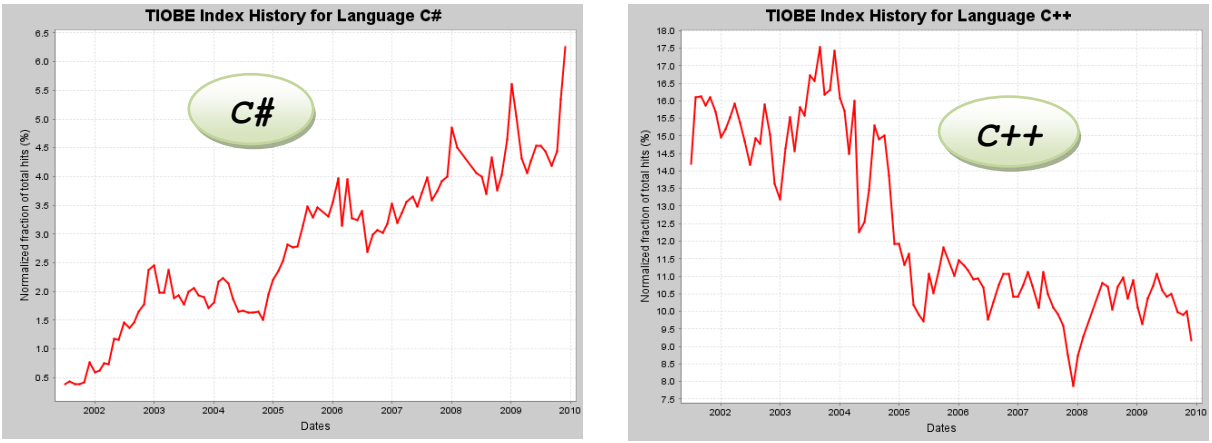


Рисунок 5.7. Динамика изменения рейтинга языков C# и C++

Классификация языков по категориям связана с методами, используемыми при написании программ.

Процедурные языки – языки высокого уровня, в которых используется метод декомпозиции программы на отдельные связанные друг с другом модули – подпрограммы (процедуры и функции). Головная программа и подпрограммы состоят из последовательности операторов, значительная часть которых использует библиотечные процедуры и функции. Основные представители: Fortran (год создания 1954, был первым процедурным языком), Cobol (1960 г., был создан как язык для создания коммерческих приложений, отличительной особенностью языка является возможность эф-

фективной работы с большими массивами данных), Algol (1960 г), Pascal (1970 г., последние версии имеют объектно-ориентированные возможности организации программ; является основой визуальной системы Delphi), C (1972 г., создавался первоначально как язык для операционной системы UNIX, позволяет работать с данными так же эффективно, как и ассемблер), Ada (1983 г., широко использовался Департаментом Защиты США).

Объектно-ориентированные языки – дальнейший уровень развития процедурных языков с основной концепцией организации программы, как совокупности программных объектов. Алгоритм решения задачи представляет собой последовательность создания экземпляров описанных предварительно пользователем или библиотечных объектов и использования их методов. Основные представители: Simula и SmallTalk (1967 г., первые объектно-ориентированных языки программирования), Ada-95 (используется Департаментом Защиты США), C++ (1983 г., широко используется во многих областях), Java (потомок C++, используется в основном для Интернет-программирования).

Функциональные языки – языки искусственного интеллекта. Программа представляет собой последовательность функций и выражения, которые нужно вычислить. Основная структура данных – связный список. Программирование на них принципиально отличается от процедурного. Основные представители: Lisp (разработан в конце 1950-х годов), Scheme, ML (MetaLanguage), Miranda, Haskel.

Логические языки – ориентированы на решение проблем без описания алгоритмов, языки искусственного интеллекта. Действительно используется только один язык – PROLOG, на котором написаны многие экспертные системы.

Языки сценариев или скрипты (scripts) – объектно-ориентированные языки для создания программ, исполняемых в определенной программной среде. Например, файлы программ на языках VBScript, JScript, Tcl могут исполняться как сервером сценариев Windows Script Host (WSH, компонент Windows), так и Интернет-браузером, когда в нем присутствует ссылка на файл скрипта в html-документе. Кроме того, текст программы на этих языках может включаться в тело html-документа. Языки Perl (1986 г.) и Python (1990 г.) возникли как скриптовые языки для операционной системы Unix, в последующем появились версии интерпретаторов языков для Windows и Macintosh. В программах на скриптовых языках часто используются вызовы процедур и функций системных библиотек, знание которых необходимо для использования этих языков.

Языки, ориентированные на данные – созданы специально для работы с одним определенным типом данных. Например, APL настроен на работу с матрицами и векторами без циклов, Snobol (1962 г.) и его преемник Icon (1974 г.) предназначены для обработки строк., SETL (1969 г.) – язык для описания операций над множествами.

Особое положение занимают языки работы с базами данных, такие, как 4GL, 3GL, PL/SQL, Transact-SQL, FoxPro (xBase), а также библиотеки работы с базами данных, такие, как ActiveX Data Objects (ADO), Microsoft Data Access Objects (DAO), доступ к объектам которых можно получить из многих универсальных языков.

Достаточно большие возможности по работе с документами Word, электронными таблицами Excel и базами данных Access имеет встроенный язык системы Microsoft Office – Visual Basic for Application (VBA).

5.2 Алгоритмизация

Алгоритм – описание последовательности действий для решения поставленной задачи.

Описание алгоритма может быть представлено в виде:

1) текстовой инструкции;
2) графической схемы;
3) программы на одном из языков программирования.

Каждый алгоритм должен иметь:

1) название, отражающее суть решаемой задачи,
2) описание исходной информации,
3) описание последовательности действий,
4) описание выходной информации.

Основные свойства алгоритмов:

1) **Универсальность** (массовость) – алгоритм решения задачи разрабатывается в общем виде, т. е. он должен быть применим для некоторого класса задач, различающихся лишь исходными данными. При этом исходные данные могут выбираться из некоторой области, которая называется областью применимости алгоритма.
2) **Дискретность** – алгоритм должен представлять процесс решения задачи, как последовательное выполнение простых операций.
3) **Однозначность** – плавила и порядок выполнения операций алгоритма не допускают неоднозначного толкования.

4) **Результативность** (или конечность) – за конечное число шагов алгоритм либо должен приводить к решению задачи, либо останавливаться из-за невозможности получить решение с выдачей соответствующего сообщения, либо неограниченно продолжаться в течение времени, отведенного для исполнения алгоритма, с выдачей промежуточных результатов.

Существуют 3 основных типовых алгоритма:

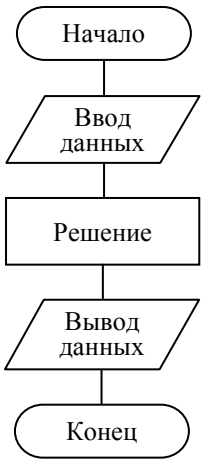
1) линейный – неизменная последовательность операций от его начала до конца без повторов действий,
2) разветвляющийся – последовательность выполняемых действий может изменяться в зависимости от каких-либо условий,
3) циклический – группа операций, которые могут повторяться многократно, кратность повтора определяется некоторым условием.

Для графического представления сложных алгоритмов, как показано ранее в разделе 5.1, используются стандарты группы **IDEF**, языки **UML, BPML, UEML**.

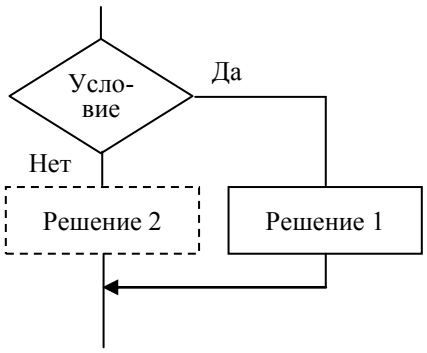
Для элементарных алгоритмов можно использовать так называемые «структурные схемы», на которые ранее существовал стандарт ГОСТ 19.701–90 (в настоящее время не относится к действующим).

Структурные схемы типовых алгоритмов могут быть представлены следующим образом:

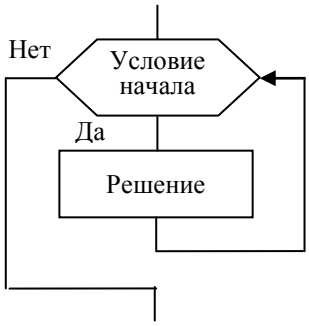
1 Линейный



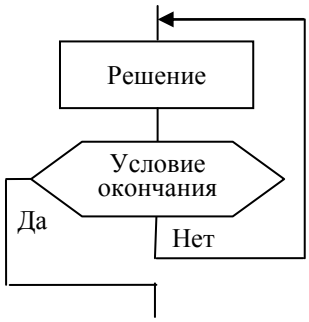
2 Разветвляющийся



3.1 Цикл с предусловием



3.2 Цикл с постусловием



В процедурных алгоритмических языках при написании программ большое значение имеют ***принципы структурного программирования***:

1) Программа разбивается на отдельные логические модули, которые могут образовывать иерархическую структуру.
2) Разработка сложной системы ведется «сверху-вниз», сначала ведется разработка головной программы, в которой присутствуют вызовы модулей (подпрограмм) более низких уровней иерархии, на первом этапе не выполняющих никаких действий («заглушки»). Далее разрабатываются модули верхнего уровня, далее более низких уровней.
3) Все названия в программах должны быть значимыми, говорить о их назначении.
4) Текст программы должен быть написан наглядно с использованием отступов в строках для выделения синтаксических конструкций (условных переходов, циклов и т. п.).
5) Запрещается использование безусловных переходов и меток. Вместо этого используется вызов модулей – процедур и функций, на которые разбита программа.

Некоторые достоинства структурного программирования:

1) Тексты программ, даже довольно крупных, становятся легко читаемыми, в результате программу без труда понимает не только её автор, но и другие программисты. Это позволяло разрабатывать достаточно крупные программные комплексы силами коллективов разработчиков, и сопровождать эти комплексы в течение многих лет.
1) В структурированных программах достаточно наглядно представлен алгоритм задачи, что позволяет обходиться без блок-схем и других графических форм изображения алгоритмов (по сути, сама программа является собственной блок-схемой).

2) Упрощается процесс тестирования и отладки структурированных программ.

Важнейшим понятием объектно-ориентированных алгоритмических языков является объект.

Объект – абстрактная именованная сущность, обладающая набором свойств (атрибутов), которая может выполнять определенные операции и реагировать на внешние события с использованием своих методов (процедур и функций).

Основными принципами объектно-ориентированного программирования являются:

1) ***Инкапсуляция*** – объединение данных с процедурами и функциями в единый блок программного кода (свойства и методы рассматриваются как поля структуры типа объект).
2) ***Наследование*** – новый объект может быть создан на основе ранее существовавших объектов, при этом он сохраняет все свойства и методы родительских объектов, но может приобретать дополнительные свойства и методы и замещать часть существующих новыми. В результате реализуются принципы агрегации (один объект содержит в себе другие объекты) и иерархического построения сложных объектов.
3) ***Полиморфизм*** – единое имя для однотипных действий, которые поразному осуществляется для различных объектов (Load, SetFocus, AfterUpdate, Click, Refresh, Close …).

В объектно-ориентированных системах широко используется ***компонентное программирование*** – сохранение объектов в виде независимо распространяемых исполняемых модулей. Известны 2 типа компонентов:

a) Компоненты конкретного языка программирования (Visual Basic, Visual C#, Object Pascal, Java).

b) Компоненты, поддерживаемые на уровне операционной системы (COM, ActiveX,.NET).

В современных объектно-ориентированных системах широко используется визуальное программирование (Visual Basic, Visual C#, Microsoft Office Access, Visual FoxPro и пр.), когда объекты пользовательского интерфейса создаются с помощью конструкторов с использованием библиотек стандартных объектов (см. пример разработки форм в системе Access в разделе 3.3.4).

5.3 Программирование на языке Visual Basic

Язык Visual Basic был выбран, как основа для изучения принципов программирования по следующим причинам:

1) Basic был первым алгоритмическим языком персональных компьютеров, по легкости читаемости программ превосходит C, C++, C#, J.
2) Этот язык имеет высокий рейтинг на протяжении многих лет (см. *http://www.tiobe.com/*).
3) Его разновидность – Microsoft Visual Basic Scripting Edition (VBScript) является встроенным языком Windows (начиная с Windows 98), программы на этом языке исполняются в любой версии Windows его серверами сценариев (Microsoft Windows Based Script Host –WScript.exe или Microsoft Console Based Script Host –CScript.exe), а также могут включаться в состав html-страниц для браузера Internet Explorer.
4) Его разновидность Microsoft Visual Basic for Application является встроенным языком программирования в системе Microsoft Office, имеющей широчайшее распространение.
5) Visual Basic традиционно является одним из основных языков системы профессиональной разработки программного обеспечения Microsoft Visual Studio (вместе с C# и C++).

5.3.1 Язык Microsoft Visual Basic Scripting Edition

Программы на языке Microsoft Visual Basic Scripting Edition (VBScript) в системе Windows имеют, главным образом административное назначение. Имеются большие библиотеки программ на VBScript (*http://www.microsoft.com/technet/scriptcenter/default.mspx*) для решения множества задач в ОС Windows, которые используют объекты, поддерживающие спецификацию Windows OLE Automation, получившую в настоящее время новое название – Microsoft ActiveX.

Система Microsoft Windows Script Technologies позволяет не только использовать библиотеки объектов, но и создавать собственные библиотеки Windows Script Component (исходный текст *.wsc, после генерации библиотеки *.tlb).

Скрипты на этом языке в составе html-страниц также достаточно распространены. Многие положения синтаксиса этого языка являются общими с диалектами Visual Basic for Application и Visual Basic.

Синтаксис языка достаточно простой и традиционный для Basic'a, дополнен возможностями работы с объектами Windows OLE Automation (ActiveX) и прочими (однако, нет возможности создавать новые виды объектов). Справку по языку можно найти в файле …*Microsoft Office\Office12\1049\Vbscrip5.chm*, если на компьютере установлен Microsoft Office.

Программа на языке VBScript состоит из инструкций языка (statement) в виде текстовых строк. Несколько инструкций языка можно объединить в одну строку в текстовом файле программы с использованием разделителя строк – символа двоеточия (:) и наоборот, одну строку программы можно написать на нескольких строках в тексте с использованием символа подчеркивания (_). В русском языке инструкции языка программирования обычно называют операторами языка, хотя это не совсем точно: операторами в английских первоисточниках называют символы для обозначения математических, логических и строковых операций (=, +, -, /, and, or, eqv, & и пр.). Далее будет использоваться традиционная русская терминология с использованием слова операторы для обозначения инструкций языка.

Для создания простейшей программы делаем следующее:

1) открываем Блокнот (а лучше – редакторы EmEditor, Aditor, UltraEdit);
2) пишем в Блокноте строку: ***`MsgBox "Привет!"`***
3) сохраняем текстовый файл с именем Prg1.vbs;
4) в свойствах файла в пункте «Открыть…» проверяем, что для работы с ним задана программа Microsoft Windows Based Script Host, если этого нет, выбираем в папке \Windows\system32\ файл **`wscript.exe;`**
5) двойным щелчком мыши открываем файл.

Результат работы этой программы – окно в системе Windows (Windows-форма), показанное на рисунке 5.8.

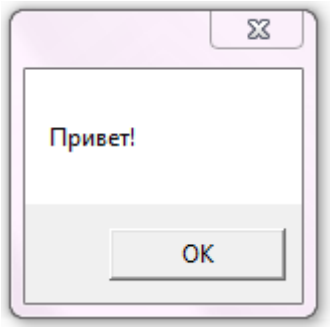


Рисунок 5.8. Пример работы простейшей программы на языке Vbscript, исполняемой системой Windows Based Script Host (wscript.exe)

При выполнении этой программы используется стандартная функция языка VBScript со следующим полным синтаксисом (в описании в квадратных скобка приводятся необязательные элементы):

*[<**p**>=]* ***MsgBox(prompt[,buttons][,title][,helpfile,context])***

где назначение аргументов функции следующее:

prompt – текст в диалоговом окне;

buttons – кнопки, значок и номер кнопки по умолчанию в окне (например, vbYesNoCancel + vbInformation + vbDefaultButton3);

title – надпись на заголовке окна (например, «Мое первое окно»);

helpfile, context – файл справки и идентификатор ее раздела.

Аргументы функции следует писать в скобках, если слева стоит переменная (***p***), которой присваивается значение, возвращаемое функцией, иначе аргументы следует писать за именем функции через пробел без скобок.

Дополним текст в файле Prg1.vbs указанными дополнительными параметрами (текст пишем в одну строку или используем знак подчеркивания **_** в конце первой строки для продолжения текста функции на следующей строке):

```
MsgBox "Привет!", vbYesNoCancel + vbInformation _
 + vbDefaultButton3, "Мое первое окно","tst.hlp",1
```

Окно запущенной программы будет иметь вид, показанный на рисунке 5.9.

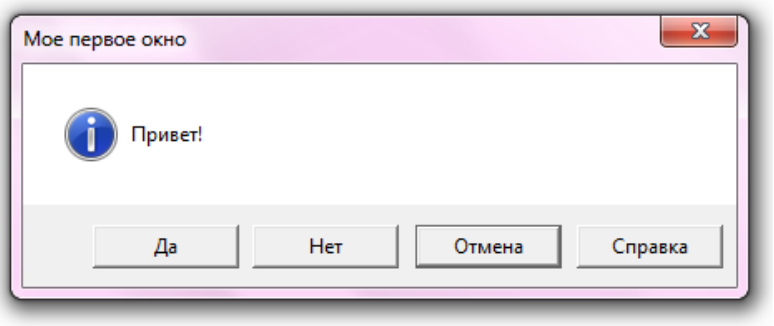


Рисунок 5.9. Пример использования функции MsgBox языка Vbscript с заданием набора кнопок и иконки окна

Функция MsgBox может возвращать значение нажатой в окне кнопки (например, vbOK, если нажата кнопка OK, либо vbYes, vbNo, vbCancel и пр.). Для этого синтаксис использования функции следует несколько изменить: слева стоит переменная **btn**, которой будет присвоено возвращаемое функцией значение, далее следует символ присваивания (**=**) и функция, у которой её параметры написать в круглых скобках:

```
btn = MsgBox("Привет!", vbYesNoCancel + vbInformation _
    + vbDefaultButton3, "Мое первое окно")
```

Диалоговое окно будет иметь тот же вид, что и раньше (см. рисунок 5.11), но переменная **btn** после нажатия кнопки в окне будет иметь соответствующее значение.

Еще одна функция языка, позволяющая открывать окно для ввода пользователем с клавиатуры строки текста:

```
[<s>=]InputBox(prompt[,title][,default][,xpos][,ypos]_
      [,helpfile, context])
```

где новые параметры функции (по сравнению с **MsgBox**):

default – строковое значение в поле ввода, которое будет показано по умолчанию при открытии окна (если нет, поле пустое);

xpos, ypos – координаты левого верхнего угла окна в единицах twips (1 twips = 1/1440 дюйма или 1/567 см) по отношению к левому верхнему углу экрана.

Дополним программу Prg1.vbs возможностью ввода данных с клавиатуры:

```
S = InputBox(vbLf & "Напишите строку текста:", _
  "Окно ввода текстовой строки", Now & " Пиши здесь" _
   & " и сейчас...", 4000, 2000, "tst.hlp",1)
MsgBox S, vbYesNo+vbInformation, "Мое первое окно",_
   "tst.hlp",1
```

Сохраним текст, как новый файл ***Prg2.vbs***, откроем его двойным щелчком мышкой и увидим окно функции **InputBox** (рисунок 5.10). В этом окне напишем новый текст в поле ввода и нажмем кнопку [Да], после чего появится окно, показанное на рисунке 5.11 (если в первом окне нажать кнопку [Нет], текст в окне **MsgBox** будет отсутствовать).

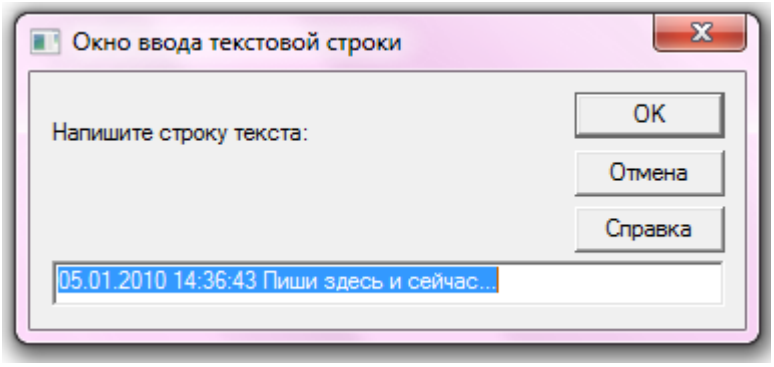


Рисунок 5.10. Пример использования функции **InputBox**

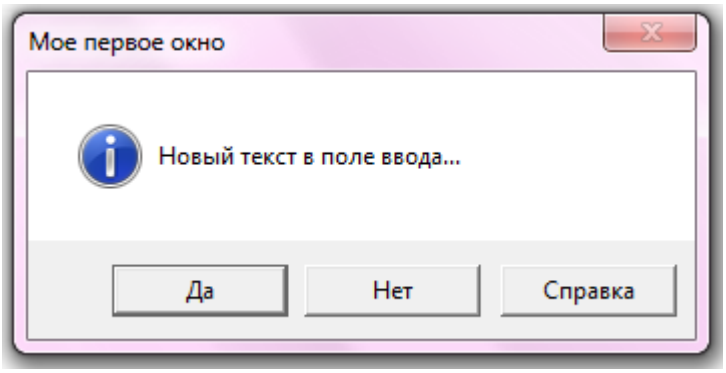


Рисунок 5.11. Функция **`MsgBox`** с показом текста, написанного ранее в поле окна **`InputBox`**

В изложенном выше материале уже использовались такие основополагающие понятия языка программирования, как строковые константы (символы между апострофами), переменные (**`btn`**, **`S`**), операция присваивания (**`=`**) и функции (**`MsgBox`**, **`InputBox`** и **`Now`**). Далее приведены более подробные сведения по синтаксису языка VBScript, причем многие положения являются аналогичными и для других диалектов языка Visual Basic.

5.3.1.1 Алфавит языка программирования

Алфавит языка – это набор разрешенных символов, с помощью которых могут быть образованы лексемы, выражения и операторы данного языка.

Лексемами называются минимальные значимые единицы текста программы.

Лексемы представлены такими категориями, как константы (числовые, строковые, дата-время и пр.), идентификаторы, комментарии, разделители, специальные символы.

Далее перечислены различные виды лексем и символы, которые можно использовать для их образования:

- Числовая константа (число)
  - десятичные числа:
    - цифры от 0 до 9,
    - символ точка (.) и латинская буква E (или e).
  - шестнадцатеричные числа:
    - &h – первые два символа числа,
    - далее цифры от 0 до 9 или латинские буквы от A до F.
- Строковая константа (строка)

  - символы таблицы ASCII с кодом от 32 до 255, заключенные между двойными кавычками " ".

- Дата
  - первый и последний символ решетка (#),
  - цифры от 0 до 9,
  - английские названия месяцев полностью или их первые 3 символа: Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec,
  - разделитель косая черта (/) или дефис (-).
- Время
  - первый и последний символ решетка (#),
  - цифры от 0 до 9,
  - разделитель точка (.) или двоеточие (:).
- Идентификатор (имя)
  - латинские строчные и прописные буквы (A – Z, a – z),
  - первый символ – только латинская буква,
  - арабские цифры (0– 9),
  - символ подчеркивания _ (но не в начале имени).
- Комментарий
  - строка после одиночного апострофа ' или после REM.
- Разделитель
  - символ пробела,
  - символ продолжения строки (_),
  - символ табуляции,
  - символ новой строки.
- Специальные символы и пары символов:
  @ # $ % ! & < > = >= <= <>

Ключевые слова – зарезервированные строковые лексемы, смысл которых задан для данного языка программирования, и которые известны транслятору.

Ключевыми словами VBScript, например, являются Abs, Add, And, Array, Asc, AtEndOfLine, …, WriteLine, Xor, Year – всего 229 слов (включая операторы +, & и др.).

5.3.1.2 Основные правила образования лексем

Число – это лексема, служащая для записи числовых значений. В качестве разделителя целой и дробной части действительных чисел используется точка (123.456). Число можно представить в экспоненциальной форме с использованием латинской буквы E или e. Число, стоящее перед E называется *мантиссой*, число после E называется *порядком*. Произведение мантиссы на 10 в степени порядка дает числовое значение. Например, лексема 1.234E+6 обозначает число $1{,}234 \cdot 10^6 = 1234000$. Признаком шестнадцатеричных чисел служат два символа в начале &h, после которых могут идти значения от 0000 до 7FFFFFFF, например: &h12ABCDEF, &h1A2B3C4D.

Строка – последовательность символов, заключенная между двойными кавычками, например: "Символы ASCII с кодом от 32 до 255, в том числе µ ђ ™ ® € є ® ‰ ‡ ħ » « ° ± • … ".

Дата – последовательность символов, заключенная между символами #. Основной формат даты #мм/дд/гггг#, #мм-дд-гг# или #Mes-дд-гггг, т. е. на первом месте стоит месяц, на втором – день, на третьем – год с разделителями косая черта с правым наклоном (/) или дефис (-). Например, #12/31/2008#, #12/31/8#, #12-31-2008 (31 декабря 2008 года). Однако, при написании названия месяца (или 3 букв названия) может использоваться формат, #дд-Mes-гг#, например, #31-Dec-08#.

При написании года двумя цифрами формат даты имеет изменяемый характер. Например, #12-04-08# означает 4 декабря 2008 г., но #13-mm-08# означает 08 <мм> 2013 г., однако #13-02-29# = 13 февраля 2029 г. Если на первом месте стоит число больше 12, формат преобразуется в #гггг-мм-дд#. Если на последнем месте стоит число, больше 28 или 29 для високосного года, формат преобразуется в #дд-мм-гггг#. Если на последнем месте стоит число, больше 30 или 31 для месяцев, в которых 30 и 31 день, формат преобразуется в #дд-мм-гггг#, как показано в следующем примере:

```
MsgBox #13-01-31#  '31.01.2013
MsgBox #13-02-31#  '13.02.1931
MsgBox #13-03-31#  '31.03.2013
MsgBox #13-04-31#  '13.04.1931
MsgBox #13-05-31#  '31.05.2013
MsgBox #13-06-31#  '13.06.1931
MsgBox #13-07-31#  '31.07.2013
MsgBox #13-08-31#  '31.08.2013
MsgBox #13-09-31#  '13.09.1931
MsgBox #13-10-31#  '31.10.2013
MsgBox #13-11-31#  '13.11.1931
```

```
MsgBox #13-12-31#  '31.12.2013
```

В связи с таким сложным поведением даты для однозначного ее написания лучше пользоваться четырьмя цифрами года в дате.

Функция **`CDate`** конвертирует строку в дату с учетом региональных установок Windows. В русских настройках даты (Панель управления – Язык и региональные стандарты – текущий формат – русский) по умолчанию задан формат даты dd.MM.yyyy. Поэтому **`Cdate("13-12-08")`** = 13 декабря 2008 г., тогда как **`#13-12-08#`** = 8 декабря 2013 г.

При неверном задании даты возникает системная ошибка (например, дата #02/29/08# правильная – високосный год, но #02/29/07# – неверная дата!).

Время – последовательность символов, заключенная между символами # в формате #чч.мм.сс# (часы.минуты.секунды) или #чч:мм:сс#, например #23.59.59#, #0.0.0#, #23:59:59# При неверном задании времени (например, #24.00.00#) после запуска программы появляется окно сообщения о синтаксической ошибке во время компиляции программы.

Дата и время – могут быть заданы, как одна лексема в формате #мм/дд/гггг чч.мм.сс# или других разновидностях форматов даты и времени. Если использовать только задание даты, в операциях с датами/временем подразумевается формат #мм/дд/гггг 00.00.00#.

Идентификатор (имя) – последовательность символов, указанных в 5 пункте, длиной не более 255 знаков; первый символ данной последовательности обязан быть буквой; последовательность должна быть отлична от ключевых слов; буквы в верхнем и нижнем регистре *не* различаются.

Комментарий – строка, идущая после символа апостроф ' или слова REM. Например:

```
Option explicit     'Запрет использовать переменные
                    ‘ без их предварительного описания
MsgBox "Привет!"    REM показ окна сообщений с заданной строкой
```

5.3.1.3 Типы данных

Каждый язык программирования предназначен для обработки информации (данных) различных типов. Используемые типы данных и методы их обозначения и обработки могут несколько различаться в различных языках.

Типы данных определяют:

- формат представления данных в памяти компьютера;
- область или диапазон возможных значений;
- множество допустимых операций, применимых к данным.

В языке *Microsoft Visual Basic Scripting Edition* определен единственный тип данных – **Variant**. Это специальный тип, который может содержать в себе различные виды информации. Все функции языка также возвращают данные типа **Variant.**

Различные категории информации, которая может содержаться в типе **Variant**, называются подтипами.

В таблице 5.4 приведены подтипы данных, которые могут содержаться в типе **Variant** (информация из справки системы Microsoft Visual Studio 2008).

Таблица 5.4. Подтипы языка VBScript

Подтип	Описание
Byte	Содержит целые в диапазоне от 0 до 255
Boolean	Содержит *True* или *False*
Integer	Содержит целые в диапазоне от -32768 до 32767
Long	Содержит целые в диапазоне от -2 147 483 648 до 2 147 483 647
Single	Содержит данные одинарной точности с плавающей точкой в диапазоне от -3.402823E38 до -1.401298E-45 для отрицательных значений; от 1.401298E-45 до 3.402823E38 для положительных значений.
Double	Содержит данные двойной точности с плавающей точкой в диапазоне от -1.79769313486232E308 до -4.94065645841247E-324 для отрицательных значений; 4.94065645841247E-324 до 1.79769313486232E308 для положительных значений
Currency	-922 337 203 685 477.5808 до 922 337 203 685 477.5807
Date (Time)	Содержит числа, которые представляют даты и время в диапазоне между 1-01-100 0:0:0 до 31-12-9999 23:59:59
Object	Содержит объект
String	Содержит строку переменной длины, которая максимально может содержать 2 миллиона символов
Empty	Неинициализированное значение (0 для числовых переменных, строка нулевой длины ("") для строковых переменных)
Null	Содержит неверные для подтипа данные
Error	Содержит номер ошибки

Функция ***VarType*** возвращает информацию о том, как данные сохранены в типе **Variant**. Для преобразования одного подтипа в другой могут использоваться соответствующие функции (***Cbyte, Cdate, CSng, Cdate, CDbl*** и др.).

Другие диалекты языка Visual Basic также имеют тип **`Variant`**, но наряду с ним могут определять переменные различных типов, таких же, как подтипы языка VBScript.

5.3.1.4 Константы

В тексте программы могут использоваться числа, строки текста, даты и время, которые являются константами.

Константа – некоторое неизменяемое значение в тексте программы. Константа может иметь имя (*идентификатор*).

Для тех констант, которые используются часто, можно задать имена. Задание имен константам делает программы легко читаемыми. Для этого в любом месте текста программы можно использовать следующее описание:

```
Const N = 1.15e-15
Const FIO = "Иванов Иван Иванович"
Const Data_r = #05-13-1988 06:30:00#
Const Time_pv = #21:00#
Const Gorod = "Архангельск"
```

В языке VBScript существует большое количество предопределенных констант, которые сгруппированы по категориям:

- ***`Color Constants`*** – определяют 8 основных цветов:
  - *vbBlack, vbRed, vbGreen, vbYellow, vbBlue, vbMagenta, vbCyan, vbWhite;*
- ***`Comparison Constants`*** – определяют тип операции сравнения:
  - *vbBinaryCompare* – двоичное сравнение;
  - *vbTextCompare* – текстовое сравнение;
- ***`Date and Time Constants`*** – определяют константы для дат и времени для функций работы с ними, некоторые из них:
  - номера дней недели: *vbSunday... vbSaturday* (1 … 7, функция *WeekdayName(vbSunday)* вернет значение «воскресенье»);
  - *vbUseSystem* – использовать формат даты и региональные установки компьютера;
  - *vbUseSystemDayOfWeek* – использовать первый день недели, заданный на компьютере;
- ***`Date Format Constants`*** – определяют форматы дат и времени (*vbGeneralDate, vbLongDate, vbShortDate, vbLongTime, vbShortTime*);
- ***`MsgBox Constants`*** – используются в функции **MsgBox** (*vbOKOnly, vbOKCancel, vbAbortRetryIgnore, vbYesNoCancel, vbYesNo, vbRetryCancel, vbCritical, vbQuestion, vbExclamation, vbInformation, vbDefaultButton1, vbDefaultButton2, vbDefaultButton3, vbDefaultButton4, vbApplicationModal, vbSystemModal, vbOK, vbCancel, vbAbort, vbRetry, vbIgnore,vbYes, vbNo*);

- ***String Constants*** – определяют скрытые символы, используемые для манипуляции со строками:
  - *vbCr* (Chr(13)) – возврат каретки (переход в начало следующей строки);
  - *vbLf* (Chr(10)) – новая строка;
  - *vbCrLf* (Chr(13) + Chr(10)) – новая строка;
  - *vbNewLine* (Chr(10) или Chr(13) + Chr(10)) – новая строка;
  - *vbNullChar* (Chr(0)) – символ с нулевым значением;
  - *vbNullString* (Chr(0)) – строка с нулевым значением;
  - *vbTab* (Chr(9)) – горизонтальная табуляция ;
  - *vbVerticalTab* (Chr(11)) – вертикальная табуляция;
- ***VarType Constants*** – определяют форматы для различных подтипов (*vbEmpty, vbNull, vbInteger, vbLong, vbSingle, vbSingle, vbCurrency, vbDate, vbString, vbObject, vbError, vbBoolean, vbVariant, vbDataObject, vbDecimal, vbByte, vbArray)*;
- ***File Attribute Constants*** – атрибуты файлов (*Normal, ReadOnly, Hidden, System, Volume, Directory, Archive, Alias, Compressed*);
- ***DriveType Constants*** – типы дисковых устройств (*Unknown, Removable, Fixed, Remote, CDROM, RAMDisk*);
- ***File Input/Output Constants*** – типы ввода-вывода для файлов *(ForReading, ForWriting, ForAppending).*

5.3.1.5 Переменные и массивы

Переменная – имя (*идентификатор*) в программе, связанное с областью оперативной памяти компьютера, предназначенное для хранения какой-либо информации, которая может изменяться во время работы программы.

Все переменные в языке VBScript имеют один тип – ***Variant*** и во время использования могут хранить данные разных подтипов.

Правила задания идентификаторов для переменных следующие:

1) идентификатор должен начинаться с латинской буквы;
2) может состоять из латинских строчных и прописных букв, цифр и символа подчеркивания (последний лучше не использовать, могут быть проблемы, т. к. этот же символ используется, как признак переноса строки);
3) длина не более 255 символов;
4) должен быть уникален в области определения.

Для объявления переменных могут служить выражения:

```
Dim X, Y, Z
Public A, B, C
Private X1, X2, X3
```

Однако, переменные в языке Basic можно и не объявлять с помощью этих описаний, достаточно написать в программе новый идентификатор и присвоить ему значение, после чего транслятор будет знать, что это переменная, например:

```
Z = 1.2345
S = "Строка текста"
DT = #12-31-08#
```

Если в начале программы написать строку:

```
Option Explicit
```

использование переменных без их явного описания (в выражениях ***Dim***, ***Public*** и ***Private)*** будет запрещена. Попытка использовать необъявленную переменную вызовет сообщение об ошибке при выполнении программы.

Считается, что явное описание переменных способствует уменьшению количества ошибок при программировании, в частности, предотвращает конфликты между переменными, используемыми в основной программе и в процедурах при одинаковых именах.

Переменная, которая хранит единственное значение, называется скалярной. Переменная может использоваться и для хранения серии чисел, такая переменная является массивом.

Массив – переменная, предназначенная для хранения пронумерованной серии значений (элементов массива).

Массив в языке VBScript после его объявления имеет тип ***Variant***, поэтому его элементам можно присваивать значения различных типов.

Массив обязательно должен быть описан перед его использованием.

Для объявления массивов используются те же операторы, что и для переменных, только после имени переменной в круглых скобках указывается количество индексов и их максимальное значение у элементов массива.

Полный синтаксис этих описаний следующий:

```
Dim varname[([subscripts])][, varname[([subscripts])]]...
Public varname[([subscripts])][, varname[([subscripts])]]...
Private varname[([subscripts])][, varname[([subscripts])]]...
```

где:

Varname – имя переменной;

Subscripts – имеет формат: ***макс.инд.1*** **[,*макс.инд.2*]...** – максимальные значения индексов, минимальное значение равно 0; массив может быть одномерный, двумерный и т. д. до 60.

Пример:

```
Dim X(99), Y(24,24), Z(99,99,99)
```

Массив может быть динамический (изменяемой размерности), при его описании в круглых скобках размерность не указывают:

```
Dim varname()
```

Для инициализации динамического массива следует использовать оператор:

```
ReDim [Preserve] varname(subscripts)[,varname(subscripts)]...
```

Параметр ***Preserve*** используется, если выполняется повторная инициализация для изменения размера массива и необходимо сохранить значения переменных, которые им уже были присвоены.

Пример:

```
Dim X()
ReDim X(10, 10, 10)
...
ReDim Preserve X(10, 10, 15)
```

Разница между описаниями ***Dim, Public, Private*** существенна при использовании внутри программы подпрограмм и функций пользователя. В этом случае различается область действия переменных и массивов.

Для описания ***Dim*** область действия различается в зависимости от места его расположения:

1) на уровне программы – переменные доступны в основной программе и во всех её подпрограммах (глобальные переменные);
2) описание в подпрограмме (процедуре или функции) – переменные используются только подпрограмме, где они описаны, инициализируются и используются при исполнении этого модуля, после завершения ее работы уничтожаются (локальные переменные).

Описания ***Public*** и ***Private*** используются только на уровне программы, переменные в этом случае доступны в основной программе и во всех её подпрограммах.

Другой способ создания переменной типа ***Variant***, содержащей одномерный массив, – с помощью функции ***Array***:

```
A = Array(10,20,30,40)
```

В этом случае значения элементов массива будут следующими:

`A(0)=10, A(1)=20, (2)=30 и A(3)=40.`

Пример программы с использованием различных описаний переменных и массивов:

```
Option Explicit
Dim Y(99), Z(99,99,99)
Y(0) = "Это 1-ый элемент массива Y - строка"
Y(1) = 123.456          ' Тип второго элемента - число
Y(99)= #12-30-2007#     ' Тип 100-го элемента - дата
Z(99,99,99)="Это элемент трехмерного массива Z "_
            & "с индексом 99,99,99"
MsgBox Y(1) & vbLf & "2-ой элемент массива Y - число: " _
    & Y(2) & vbLf & "100-ый элемент массива Y - дата: " _
    & Y(99)& vbLf & Z(99,99,99),,"Окно 1"
Dim X                   'Глобальная переменная X
X="Это переменная X основной программы"
MsgBox X,,"Окно 2"
Call my_proc
 Sub my_proc
  Dim X                 'Локальная переменная X
  X="Это переменная X подпрограммы"
  MsgBox X,,"Окно 3"
 End Sub
MsgBox X,," Окно 4 - снова глобальная X "
```

Окна, которые показывает эта программа, показаны на рисунках 5.12а и 5.12б.

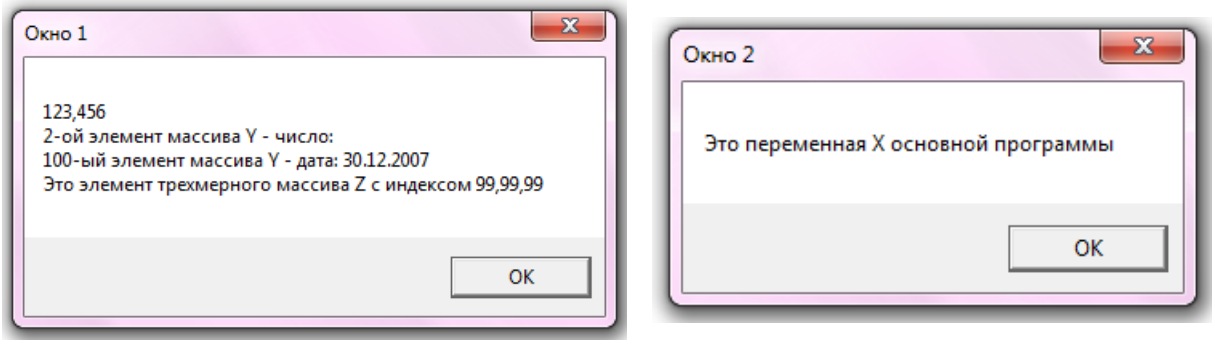


Рисунок 5.12а. Окна 1 и 2 программы с различным описанием переменных и массивов

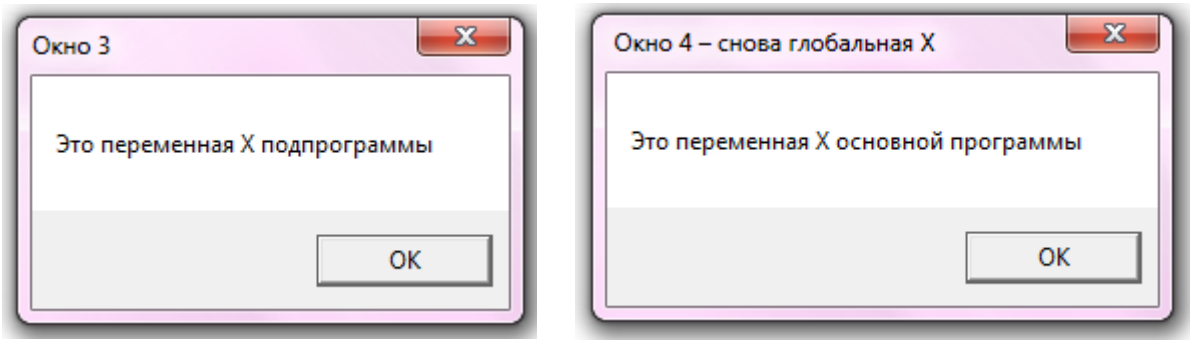


Рисунок 5.12б. Окна 3 и 4 программы с различным описанием переменных и массивов

5.3.1.6 Управляющие структуры в языке VBScript

К управляющим структурам относятся операторы условного перехода и цикла.

Оператор условного перехода позволяет выполнить те или иные строки программы в зависимости от логических условий.

В языке VBScript он может использоваться в двух различных видах (строчный и блочный синтаксисы).

Строчный синтаксис:

```
If <условие> Then <операторы1> [Else  <операторы2>]
```

Строчный синтаксис следует использовать для коротких, простых условных операторов.

Пример строчного синтаксиса:

```
If A <= 9 Then A = A + 1 : B = B + A Else B = B + A
```

Блочный синтаксис является более структурированным, имеет большие возможности, легче читается и отлаживается.

Блочный синтаксис оператора условного перехода:

```
If <условие> Then
   [операторы]
[ElseIf <условие-n> Then
   [операторы-n]]...
[Else
    [else-операторы]]
End If
```

где:

условие – логическое выражение, результатом вычисления которого может быть истина (**True**), ложь (**False**) или **Null** которое при-

равнивается к **False** (правила написания приведены далее в подразделе 5.3.1.11);

операторы – один оператор или более (разделенных двоеточиями для строкового синтаксиса); выполняются, если условие истинно (**True**);

условие-n – то же, что и ***условие***;

операторы-n – выполняются, если ***условие-n*** является истиной (**True**);

else-операторы – один оператор или более, выполняющиеся, если предшествующие условия не были истинны.

Когда выполняется блочный ***If,*** проверяется ***условие,*** и, если оно истинно (***True***), выполняются ***операторы,*** следующие за ***Then***. Если ***условие*** не является истинным (***False***), каждое ***условие-n***, идущее за ***ElseIf*** (если они есть) проверяется. Когда истинное значение найдено, выполняются ***операторы-n***, следующие за ***Then*** после истинного условия, после чего программа выходит за ***End If*** (т. е. последующие ***ElseIf***, если они есть, не проверяются). Если истинных условий для ***ElseIf*** не найдено, выполняются ***else-операторы,*** следующие за ***Else***.

Пример блочного синтаксиса:

```
a = 1
If a > 10 Then
   MsgBox "a > 10"
ElseIf a > 0 Then
    MsgBox "a > 0"  'будет выполнено только это!
ElseIf a = 1 Then
     MsgBox "a = 1"
Else
    MsgBox "Нет данных для заданного значения a "
End If
```

Пример использования оператора условного перехода:
расчет суммы страховых взносов в Пенсионный фонд РФ, где суммы перечислений (**Sp** и **Np**) зависят от налоговой базы (**S**) и года рождения (**G**):

S	**G < 1967**		**G >= 1967**	
	Sp	**Np**	**Sp**	**Np**
S <= 280000	10.3•**S**/100	0	4.3•**S**/100	6.0•**S**/100
280000<**S**<=600000	28840+5.5(**S**–280000)/100	0	12040+3.1(**S**–280000)/100	16800+2.4(**S** –280000)/100
S>600000	46440	0	21960	24480

Текст программы:

```
Dim Sm, G, Sp, Np
S=InputBox("Укажите налоговую базу","Расчет взносов в ПФ")
G=InputBox("Укажите год рождения", "Расчет взносов в ПФ")
Np = 0 : Sp = 0
If G < 1967 Then
   If S < 280000 Then Sp = 10.3*S/100
   If S > 280000 and S <= 600000 Then _
         Sp = 28840+5.5*(Sm-280000)/100
   If S > 600000 Then Sp = 46440
Else     'далее расчет для G >= 1967
   If S < 280000 Then Sp = 4.3*S/100 : Np = 6.0*S/100
   If S > 280000 and S <= 600000 Then
        Sp = 12040+3.1*(S-280000)/100
        Np = 16800+2.4*(S-280000)/100
   End If
   If S > 600000 Then Sp = 21960 : Np = 24480
End If
MsgBox "Np =" & Np & "  Sp = " & Sp
```

Оператор выбора Case позволяет выполнить те или иные операторы в зависимости от множества значений заданного выражения или переменной.

Синтаксис оператора выбора:

```
Select Case <тест-выражение>
   [Case <список_выр-n>
      [<операторы-n>]]...
   [Case Else
      [<else-операторы-n>]]
End Select
```

где:

тест-выражение –любое числовое или строковое выражение;

список_выр-n – список из одного или более выражений для соответствующего ***Case;***

операторы-n – один оператор или несколько, выполняющихся, если ***тест-выражение*** имеет то же значение, что и значение одного из выражений ***списка-n;***

else-операторы-n – один оператор или несколько, выполняющихся, если ***тест-выражение*** не совпадает ни с одним из значений ***Case***-структур.

Существует ряд особенностей в выполнении структуры **Select Case**: для целых чисел условие отбора сработает и для соответствующего строкового подтипа, но для действительных чисел такое соответствие не наблюдается. как показано в следующем примере:

```
A = 1
Select Case A
 Case 1.1, 1.2, 1.3 MsgBox "A = 1.1,1.2,1.3"
 Case "0.5","1","1.5" MsgBox "A = ""0.5"","1","1.5"""
      'будет выполнено только для Case "0.5","1","1.5"!
 Case 0.5,1,1.5 MsgBox "A = 0.5,1.55,1.56"
 Case Else MsgBox "Нет данных"
End Select
```

```
A = 1.55
Select Case a
   Case 1.1,1.2,1.3  MsgBox "A = 1.1,1.2,1.3"
   Case "0.5","1.55","1.56"  MsgBox "A = ""0.5..."""
   Case 0.5,1.55,1.56 MsgBox "A = 0.5,1.55,1.56"
       'будет выполнено только для Case 0.5,1.55,1.56!
   Case Else MsgBox "Нет данных"
End Select
```

Если же определить переменную **a =** "**1.55**" (строковое значение), в приведенном примере возникнет ошибка при выполнении с сообщением о несоответствии типов.

Оператор цикла позволяет выполнить группу операторов несколько раз в соответствии с заданными условиями повтора.

Существует несколько видов операторов цикла:

1) **Do... Loop**
2) **While... Wend**
3) **For... Next**
4) **For Each... Next**.

Синтаксис оператора **Do...Loop** следующий (здесь и далее в фигурных скобках {} приведены два возможных варианта, разделенных вертикальной чертой |, один из которых необходимо использовать)**:**

1) первый вариант – проверка условия в начале цикла

```
Do [{While | Until} <условие>]
   [<операторы>]
   [Exit Do]
   [<операторы>]
Loop
```

2) второй вариант – проверка условия в конце цикла

```
Do
   [<операторы>]
   [Exit Do]
   [<операторы>]
Loop [{While | Until} <условие>]
```

где:

условие – логическое выражение, которое имеет значение истина (**True)** или ложь (**False)**; значение условия **Null** то же, что и **False**; для ***While*** (англ. пока) выполнение цикла продолжается, пока условие истинно, для ***Until*** (англ. до) – выход из цикла, когда условие истинно;

операторы – один или несколько операторов, выполнение которых повторяется, пока условие после ***While*** истинно (**True)** или условие после ***Until*** ложно (**False**);

Exit Do – может использоваться, как альтернативный выход из цикла (на следующую строку программы после ***Loop***); любое количество ***Exit Do*** может быть помещено внутри цикла. Обычно эта команда используется с вычисляемым логическим выражением оператора ***If...Then***.

Пример использования операторов циклов ***Do While...Loop*** и ***Do Until...Loop***:

```
eps = 1e-7
a = 1 : s = a : n = 2
Do While Abs(a) > eps          ' цикл выполняется, пока |a| > eps
   ' или Do Until Abs(a) <= eps  – выход из цикла при |a| <= eps
  a = - a * (2*n - 3)/(2*n - 1)
  s = s + a :  n = n + 1
Loop
MsgBox(" Расч. Pi = " & 4*S & vbLf & "n = " & n)
```

Результатом работы программы будет число π = 3,141592**8**5358975 при n=5000002, продолжительность расчета менее 30 сек. (π = 3,141592**6**53589793238462643383279 5 на калькуляторе Windows). Если задать **eps = 1e-8,** то время расчета увеличится до 3 – 4 мин., результат расчета π = 3,1415926**7**359025 при n=50000002.

Внимание! При неверном написании условий окончания цикла программа может зациклиться (будет работать до завершения работы операционной системы). Чтобы прекратить выполнение зациклившейся программы, необходимо открыть средство Windows *Диспетчер Задач* (Task Manager) с использованием

сочетания клавиш Ctrl+Alt+Delete или, щелкнув правой кнопкой мыши на пустом месте панели задач, и выбрав в контекстном меню это средство, далее в разделе процессов найти и выделить ***wscript.exe*** и нажать кнопку ***Завершить процесс*** (End Process).

Пример использования операторов циклов ***Do...Loop While*** и ***Do...Loop Until*** (результат работы – рисунок 5.13):

```
Randomize
Otvet1 = vbYes : Otvet2 = vbNo
Do
 Do
   MySum = FormatNumber(1000000*Rnd,2)
         'Случайное число от 0.00 до 1000000.00
   SSum = "Случайная сумма = " & MySum
   Otvet2 = MsgBox(SSum & " руб." & vbLf & vbLf _
            & " Вам нравится такая сумма?", _
            vbYesNo,"Максимум - 1 000 000 руб.!")
 Loop Until Otvet2 = vbYes 'выход из цикла, если нажата кнопка Да
 proc = FormatNumber(MySum/1000000*100,1)
 s = MySum & " руб. составляет " & proc & " % от 1 млн."
 If proc>95 Then
    s = s & vbLf & " Очень хорошо!"
 ElseIf proc>85 Then
   s = s & vbLf & " Совсем неплохо!"
 ElseIf proc>75 Then
   s = s & vbLf & " Бывает и больше...!"
 Else
  s = s & vbLf & " Маловато!"
 End If
 Otvet1 = MsgBox(S & vbLF & vbLF & "Повторить генерацию" & _
      "случайных чисел?",vbYesNo,"Оценка результата... ")
Loop While Otvet1 = vbYes ' цикл повторяется, если нажата кнопка Да
```

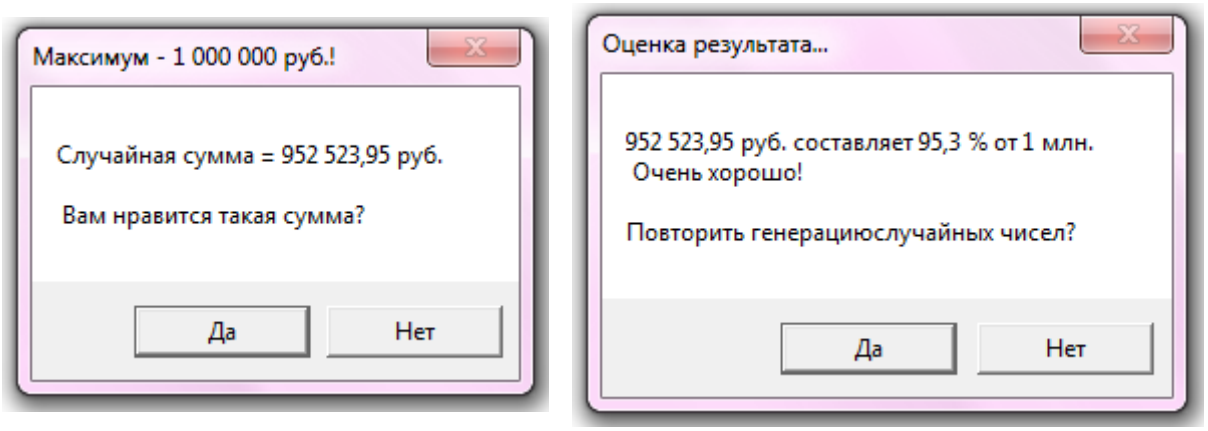


Рисунок 5.13. Пример использования циклов ***Do...Loop***

Синтаксис оператора цикла ***While...Wend*** следующий:

```
While <условие>
    [<операторы>]
Wend
```

Выполнение операторов цикла повторяется, пока ***<условие>*** истинно (**True**).

Пример использования оператора ***While...Wend*** (результат работы программы – рисунок 5.14):

```
Xn = -3 : Xk = 3 : N = 15
H = (Xk - Xn)/(N-1)
X = Xn
S =     "----------------" & vbLF
S = S + "|    X     |      Y    |" & vbLF
S = S + "----------------" & vbLF
While X <= Xk
   Y = Sin(X)
   If X>=0 Then XT = " " & FormatNumber(X,3) _
        Else XT = FormatNumber(X,3)
   If Y>=0 Then YT = "  " & FormatNumber(Y,3) _
        Else YT = FormatNumber(Y,3)
   S = S & XT & "  | " & YT & vbLF
   X = X + H
Wend
S = S + "----------------" & vbLF
MsgBox S, vbYes, "    Y = SIN(X)"
```

Синтаксис оператора цикла ***For...Next*** следующий:

```
For <счетчик> = <нач.знач.> To <кон.знач.> [Step <шаг>]
    [<операторы>]
    [Exit For]
    [<операторы>]
Next
```

где:

счетчик – числовая переменная, используемая как счетчик цикла; может быть положительной или отрицательной величиной

нач.знач. – начальное значение счетчика;

кон.знач. – конечное значение счетчика;

шаг – шаг изменения счетчика; на данную величину автоматически изменяется ***счетчик*** после каждого выполнения операторов цикла; если ***шаг*** не указан, значит он равен 1;

операторы – выполняются повторно столько раз, сколько определено значениями, заданными для ***счетчика***: один раз, много раз или ни

одного;

Exit For – может использоваться, как альтернативный выход из цикла; обычно используется с проверкой условия выхода в операторе ***If...Then;*** выход выполняется на строку программы, следующую за ***Next***.

Пример использования цикла ***For...Next*** (см. рисунке 5.15):

```
S=""
For i = 1 to 4.5 step 1/5.25
 s = s & i & vbLF
Next
MsgBox S,, " For... step 1/5.25"
```

Пример программы с альтернативным выходом (см. рисунок 5.16):

```
S = "    X         Y" & vbLF
S = S & "---------------" & vbLF
For X = 1 to 5 step 0.11
  Y = FormatNumber(Tan(X),3)
    If Abs(Y) < 1 Then
    s = s & "---------------" & vbLf _
        & "Выход из цикла" & vbLf & "при |Y| < 1"
    Exit For
    End If
    s = s & FormatNumber(X,3) & "    " & Y & vblf
Next
MsgBox S,," For...Exit For...Next"
```

Синтаксис оператора цикла ***For Each...Next*** следующий:

```
For Each <элемент> In <группа>
    [операторы]
    [Exit For]
    [операторы]
Next [<элемент>]
```

где:

элемент – переменная, которая используется для перебора всех элементов коллекции или массива;

группа – имя коллекции объектов или массива.

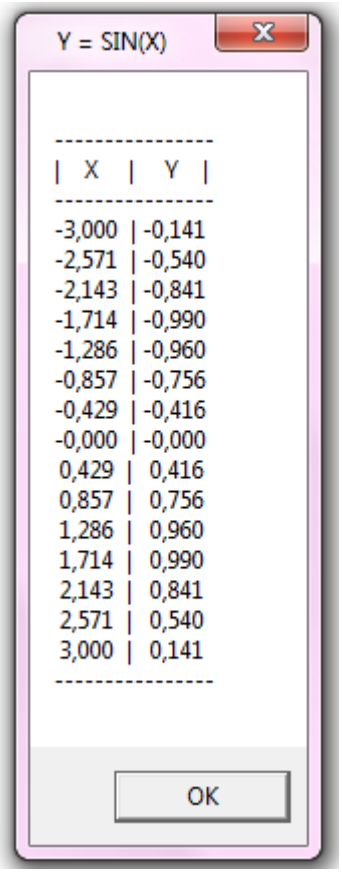


Рисунок 5.14. ***While...Wend***

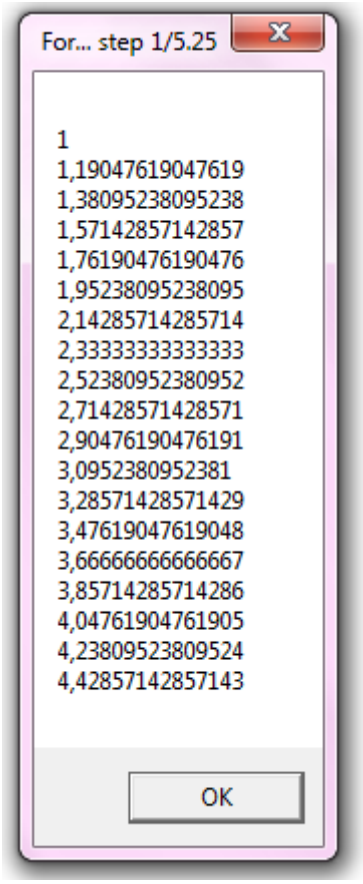


Рисунок 5.15. ***For...Next***

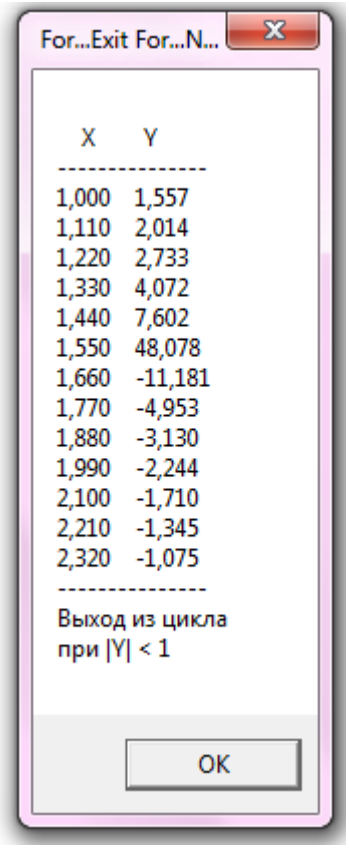


Рисунок 5.16. ***For...Exit For ...Next***

Пример программы с использованием оператора ***For Each*** для работы с массивом (работа с коллекциями объектов будет рассмотрена далее):

```
Dim x(9)
Randomize
For i = 0 to 9
    x(i) = 10*Rnd + 1
Next
S = ""
For Each iks in X
    S = S & iks & vblf
Next
MsgBox S,," For Each... "
```

В данной программе цикл ***For Each*** использует все значения массива *X* для формирования строки *S* (см. рисунок 5.17).

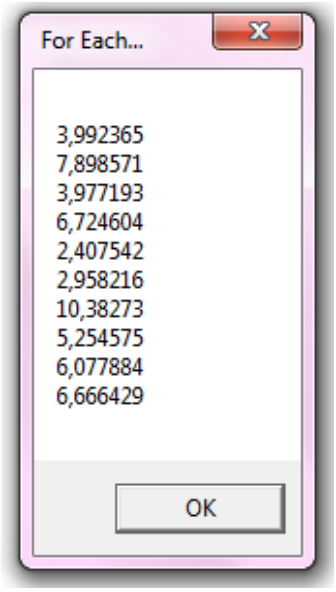


Рисунок 5.17. Цикл ***For Each***

5.3.1.7 Процедуры и функции пользователя в языке VBScript

Процедуры и функции используются для структурирования сложных программ, повышающего наглядность и читаемость программ, что облегчает процессы отладки. Процедуры и функции могут многократно использоваться при исполнении программы, для исполнения их с различными исходными данными может использоваться список аргументов.

Разница между процедурой и функцией:

имя функции после ее выполнения приобретает некоторое значение (имя функции возвращает вычисленное значение), в результате она может использоваться в составе выражений (математических, строковых и др.) программы;

имя процедуры служит только для запуска ее в работу, результатом выполнения могут быть значения переменных, массивов и пр. компонентов языка, являющихся аргументами процедуры или компонентами, общими для вызывающей программы и процедуры.

Синтаксис описания процедуры следующий:

```
[Public | Private] Sub <имя_проц.> [(<сп.арг.>)]
    [<операторы>]
    [Exit Sub]
    [<операторы>]
End Sub
```

где:

Public – показывает, что процедура доступна во всех других процедурах программы (является значением По-умолчанию);

Private – показывает, что процедура доступна только в тех процедурах программы, где она объявлена;

имя_проц. – имя процедуры, составленное по правилам написания идентификаторов;

сп.арг. – список аргументов, которые передаются процедуре, когда она вызывается для исполнения; элементы списка разделяются запятыми;

операторы – любая группа операторов, которая будет исполняться в процедуре.

Список аргументов имеет следующий синтаксис:

```
[ByVal | ByRef] <имя_переменной>[( )]
```

где:

ByVal – аргумент передается, как его значение (может быть переменной или константой при вызове процедуры, не может возвращать из процедуры значения);

ByRef – аргумент передается, как ссылка; этому аргументу при обращении к процедуре должна соответствовать переменная (или массив) вызывающей программы и ее значение может изменяться после выполнения процедуры; по умолчанию параметры процедуры имеют тип ***ByRef;***

имя_переменной – имя переменной, являющейся аргументом процедуры, составленное по правилам написания идентификаторов..

Процедуры могут использовать локальные и глобальные переменные. Локальные переменные могут быть объявлены в процедуре с использованием структуры ***Dim*** *<переменные>*. Переменные, которые используются в процедуре, но явно не объявлены, также локальные, если они явно не объявлены на более высоком уровне за пределами процедуры.

Значения локальных переменных в процедурах не сохраняются между повторными вызовами процедур (области памяти под локальные переменные используются только во время работы процедуры).

Процедура не может быть описана внутри другой процедуры (т. е. не допускается использование вложенных процедур).

Exit Sub может использоваться для выхода из любого места процедуры с возвратом на следующую строку в программе после ее вызова.

Вызов процедуры необходимо выполнять с использованием структуры ***[Call] <имя_проц.> [(<сп.арг.>)]***. Можно не использовать ключевое слово ***Call*** при вызове процедуры, в этом случае список аргументов должен быть написан без круглых скобок сразу именем процедуры: ***<имя_проц.> <сп.арг.>***. Если используется ключевое слово ***Call***, список аргументов должен быть написан в круглых скобках: ***Call <имя_проц.> (<сп.арг.>)***.

Пример использования процедуры в программе:

```
 N = InputBox("Задайте число", _
    "Вычисление факториала числа")
Sub F_N(ByVal NF, FN)
  FN=1
  For i = 1 to NF
     Fn = Fn * i
  Next
   NF = 0 'только для доказательства, что N не изменится!
End Sub
Call F_n(N, Fct)
MsgBox "Факториал числа " & N & " равен " _
       & FCT,, "Результат расчета"
```

Результат работы программы показан на рисунке 5.18.

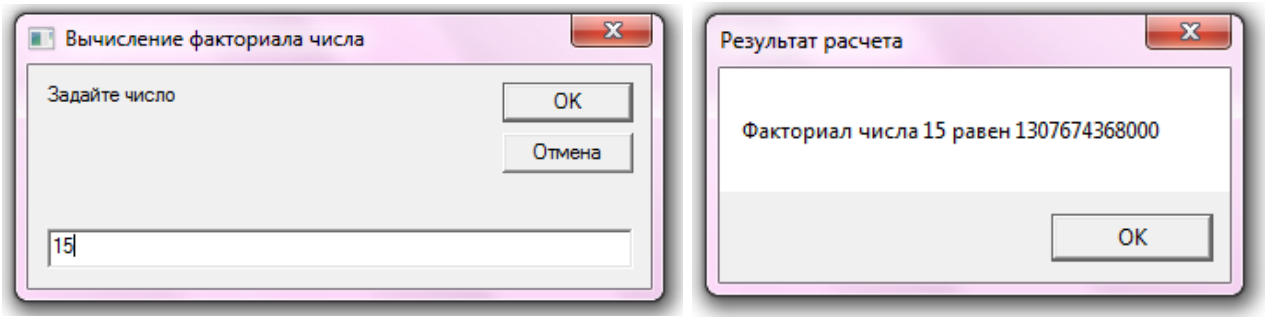


Рисунок 5.18. Использование процедуры Sub

Синтаксис описания функции следующий:

```
[Public | Private] Function <имя_функц.>  [(<сп.арг.>)]
    [<операторы>]
    [<имя_функц.> = <выражение>]
    [Exit Function]
    [операторы]
    [<имя_функц.> = <выражение>]
End Function
```

где:

имя_функц. – имя функции, составленное по правилам написания идентификаторов;

выражение – вычисляемое значение, которое возвращает имя функции;

остальные обозначения те же, что и для процедуры.

Значение, возвращаемое функцией, по умолчанию имеет тип ***Public***, если не указано иное (***Private***).

Аналогично процедуре, функция не может быть вложенной в другую функцию.

Пример использования функции в программе (для того, чтобы показать разницу в использовании процедур и функций, использован тот же алгоритм расчета факториала, что и в примере для процедуры):

```
N = InputBox("Задайте число",_
    "Вычисление факториала числа")
Function F_N(NF)
  F_N=1
  For i = 1 to NF
     F_N = F_N * i
  Next
End Function
MsgBox "Факториал числа " & N & " равен " _
       & F_N(N),, "Результат расчета"
```

Результат работы программы будет тот же, что и ранее (см. рисунок 5.18).

5.3.1.8 Работа с числовой информацией

При выполнении математических вычислений используются символы математических операций:

= присваивание, **+** сложение, **-** вычитание, **/** деление, ***** умножение, **^** возведение в степень.

Математических функция в языке VBScript достаточно много:
Abs Atn Cbool CByte CCur CDbl CInt CLng Cos CSng CStr Exp Fix Int FormatCurrency FormatNumber FormatPercent Hex \ Lbound Log Mid Mod Oct Randomize Rnd RGB Round Sgn Sin Tan TypeName Ubound VarType. Описание функций приведено в **Приложении 7.**

Порядок выполнения операций в математических выражениях – общепринятый в математике (приоритет математических операций):

1) математические функции,
2) возведение в степень,
3) умножение и деление,
4) сложение и вычитание.

Круглые скобки **( )** могут использоваться для изменения порядка выполнения этих операций. Вначале выполняются операции в скобках, затем вне их. Если используются вложенные скобки, вначале выполняются действия во внутренних скобках, затем во внешних.

Например, для математической записи формулы:

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

текст в программе на языке VBScript будет следующий:

```
X = (- b + (b^2 - 4*a*c)^(1/2))/(2*a)
```

Особенность записи формул – нельзя пропускать знаки умножения, как это часто делается в математике.

Для всех тригонометрических формул (**Atn, Cos, Sin, Tan**) единица измерения углов – радианы, например, число π (пи) можно вычислить следующим образом:

```
pi = 4 * Atn(1)
```

Для сложных математических формул иногда бывает целесообразно выполнять вычисление по частям, например:

$$y = \frac{tg^2(2b+c)}{\sqrt{\left|\sqrt[3]{(a-b)^2} - \frac{e^{-b^2} \cdot \ln a}{a + \ln(2b+c)}\right|}}$$

можно вычислить следующим способом:

```
y1 = (Tan(2*b + c))^2
y2 = (a - b)^(2/3)
y3 = (Exp(-b^2)*Log(a))/(a + Log(2*b +c))
y4 = abs(y2 - y3)^(1/2)
y = y1/y4
```

При записи такой формулы в одну строку получилось бы следующее выражение:

```
y = (Tan(2*b + c))^2/abs((a - b)^(2/3) - _
    (Exp(-b^2)*Log(a))/(a + Log(2*b +c)))^(1/2)
```

которое достаточно трудно проверить на правильность написания.

Для вычисления логарифмов с разными основаниями можно использовать следующую программу (в примере – вычисление натурального логарифма, см. рисунок 5.19):

```
n = exp(1)
x = 123.45
 MsgBox " Ln(" & x &") = " _
     & logn(x,n),,"Logn(x)"
Function logn(xf,nf)
  logn = Log(xf) / Log(nf)
End Function
```

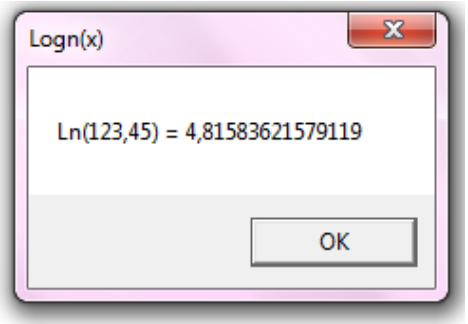


Рисунок 5.19. Вычисление логарифма ***LN(x)***

5.3.1.9 Работа со строковой информацией

Функции, которые могут использоваться при работе со строками, следующие (описание см. в **Приложении 8**):

Asc Chr & InStr InStrRev Join Lcase Left Len LTrim Mid RTrim Trim Replace Right Space Split String StrComp StrConv StrReverse Tab TypeName Ucase VarType.

Пример использования:

a) присвоить значение переменной ***FIO***

```
FIO = "Иванов Петр Сидорович"
```

b) написать ***FIO*** прописными буквами

```
FIO_p = Ucase(FIO)
```

c) разделить *FIO* на 3 переменные: фамилию, имя и отчество

1) первый вариант:

```
n1 = InStr(FIO," ")   'n1= 7, позиция первого пробела в FIO
F1 = Left(FIO,n1-1)   'F1="Иванов", 6 символов слева
n2 = InStr(n1+1,fio," ",1)      'm2 = 12
F2 = Mid(FIO, n1+1, n2-n1-1)
     'F2 = "Петр", 4 (12-7-1) символа начиная с 8
L  = Len(FIO)         'L = 21
F3 = Mid(FIO, n2+1, L-n2) 'F3 = "Сидорович"
```

2) второй вариант

```
f = Split(FIO)
      ' f(0) = "Иванов"
      ' f(1) = "Петр"
      ' f(2) = "Сидорович"
```

d) получить строку – инициалы и фамилия

```
F4 = Left(F2,1) & ". " & Left(F3,1) & ". " & F1
     'F4 ="П. С. Иванов"
```

На рисунке 5.20 показаны два примера выполнения программы, содержащей описанные выше операций работы со строками.

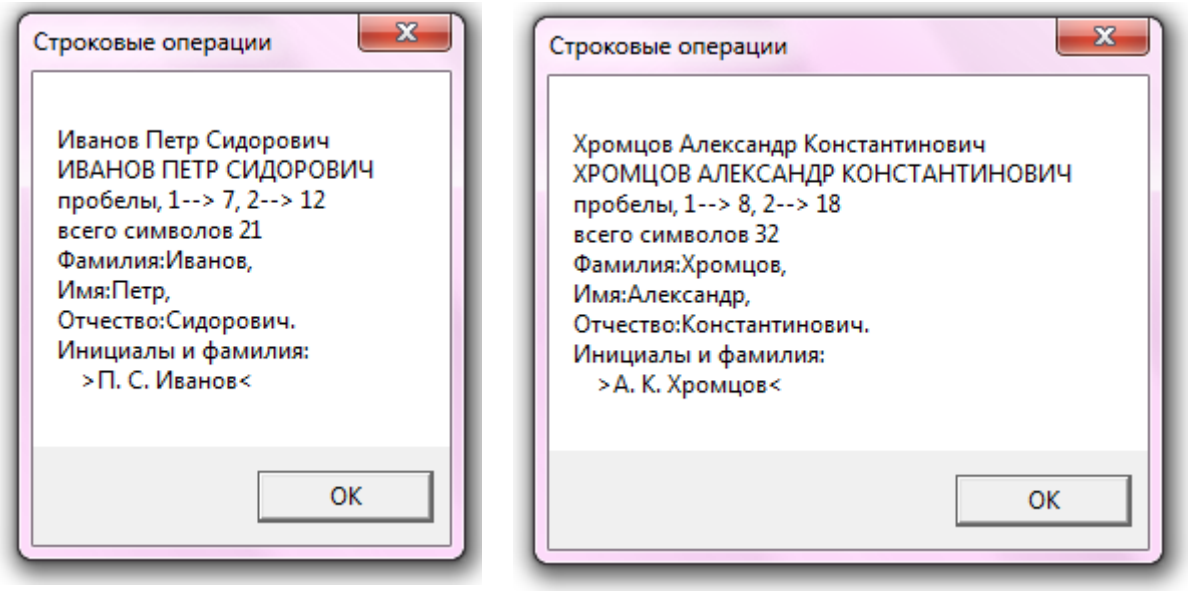


Рисунок 5.20. Использование функций для обработки строковой информации

5.3.1.10 Работа с информацией дата и время

Функции, которые могут использоваться при работе с данными типа дата и время, следующие (описание см. в **Приложении 9**):

CDate *DateAdd DateDiff DatePart DateSerial DateValue Day FormatDateTime Hour Minute Month MonthName Now Second* Time TimeSerial TimeValue TypeName VarType Weekday WeekdayName Year.

Дату и время на часах компьютера возвращают функции **Now** и **Time.** Если заданы два значение типа дата и время, операция вычитания даст разницу между ними в днях в виде действительного числа.

Например, разница #05-02-2008 18:00# - #05-01-2008 12:00# будет равна 1.25 дня.

Если необходимо вычислить разницу в определенных единицах (годах, кварталах, месяцах, неделях, днях, часах, минутах и секундах), следует использовать функцию **DateDiff**, как показано в следующем примере:

```
Dt1 = #31-Dec-2005 12:00:00#
Dt2 = # 14-Apr-2009 18:01:01#
MsgBox "Дата 1: " & Dt1 & vbLf &"Дата 2: " & Dt2 & vbLf & _
 "лет (yyyy)     " & DateDiff("yyyy",dt1, dt2) & vbLf & _
 "кварталов (q)  " & DateDiff("q", dt1,dt2) & vbLf & _
 "месяцев (m)    " & DateDiff("m", dt1,dt2) & vbLf & _
 "недель (ww)    " & DateDiff("ww",dt1,dt2) & vbLf & _
 "дней года (y)  " & DateDiff("y", dt1,dt2) & vbLf & _
 "дней (d)       " & DateDiff("d", dt1,dt2) & vbLf & _
 "часов (h)      " & DateDiff("h", dt1,dt2) & vbLf & _
 "минут (n)      " & DateDiff("n", dt1,dt2) & vbLf & _
 "секунд (s)     " & DateDiff("s", dt1,dt2),, _
 "Функция DateDiff"
```

Результат работы данной программы показан на рисунке 5.21.

Аналогичные параметры задания единицы измерения имеет функция **DateAdd**, позволяющая прибавить заданный диапазон даты и времени к начальному значению, как показано в примере (результат выполнения программы – рисунок 5.22):

```
Dt1 = #31-Dec-2005 12:00:00#
MsgBox  "Дата: " & Dt1 & vbLf & vbLf & _
"+1 год (yyyy)     " & DateAdd("yyyy",1,dt1) & vbLf & _
"+1 квартал (q)    " & DateAdd("q", 1,dt1) & vbLf & _
"+1 месяц (m)      " & DateAdd("m", 1,dt1) & vbLf & _
"+1 неделя (ww)    " & DateAdd("ww",1,dt1) & vbLf & _
"+1 день года (y)  " & DateAdd("y", 1,dt1) & vbLf & _
"+1 день (d)       " & DateAdd("d", 1,dt1) & vbLf & _
"+1 час (h)        " & DateAdd("h", 1,dt1) & vbLf & _
"+1 минута (n)     " & DateAdd("n", 1,dt1) & vbLf & _
"+1 секунда (s)  " & DateAdd("s", 1,dt1),, _
"Функция DateAdd"
```

Те же параметры задания возвращаемой части даты-времени у функции **DatePart** (результат показан на рисунке 5.23):

```
Dt1 = #31-Dec-2005 12:01:01#
MsgBox  "Дата: " & Dt1 & vbLf & vbLf & _
 "год (yyyy)      " & DatePart("yyyy",dt1) & vbLf & _
 "квартал (q)     " & DatePart("q",dt1) & vbLf & _
```

```
"месяц (m)       " & DatePart("m",dt1) & vbLf & _
"неделя (ww)     " & DatePart("ww",dt1) & vbLf & _
"день года (y)   " & DatePart("y",dt1) & vbLf & _
"день месяца(d) " & DatePart("d",dt1) & vbLf & _
"час (h)         " & DatePart("h",dt1) & vbLf & _
"минута (n)      " & DatePart("n",dt1) & vbLf & _
"секунда (s)     " & DatePart("s",dt1),, _
"Функция DatePart"
```

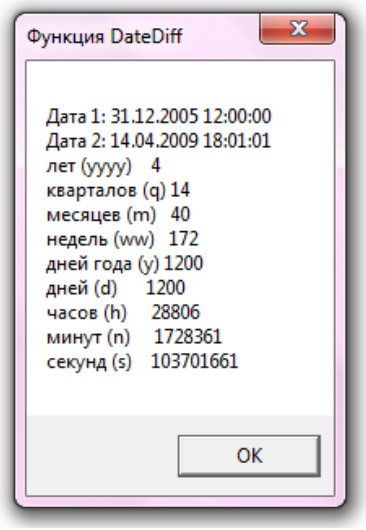


Рисунок 5.21. Использование функции ***DateDiff***

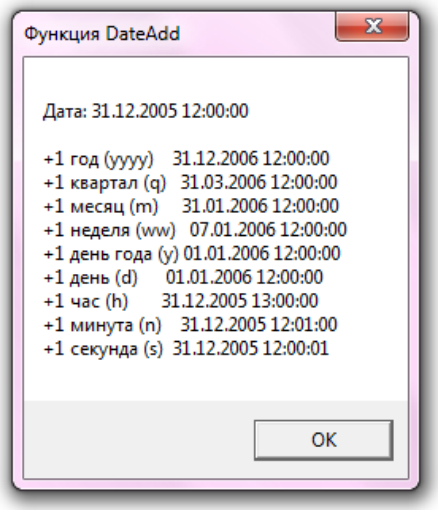


Рисунок 5.22. Использование функции ***DateAdd***

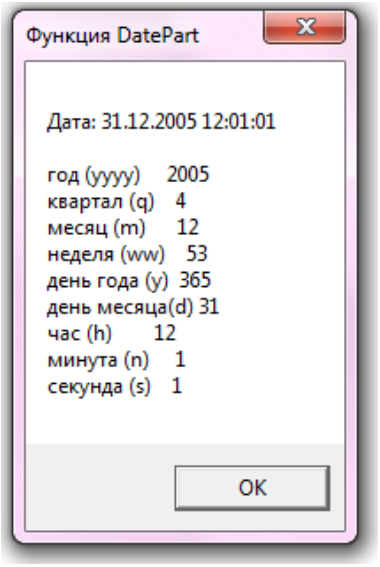


Рисунок 5.23. Использование функции ***DatePart***

Год, месяц, день, час, секунду для заданной даты и времени можно также определить с помощью функций **Year, *Month, Day, Hour, Minute, Second.***

При задании в тексте программы данных подтипа дата и время можно использовать английские названия месяцев (нельзя русские). Однако, при использовании функции преобразования строки в дату-время все наоборот, если в региональных установках Windows задан русский формат дат:

```
StrDt = "1 Окт 1999"      ' Строковое значение
Data1 = CDate(StrDt)      ' Преобразование в дату:
                          ' Data1 будет равна #10-01-1999#
```

Тот же результат дает функция ***DateValue:***

```
Date2 = DateValue(StrDt)
```

Если заменить в региональных настройках формат дат на English, система перестает понимать русские названия месяцев и признает только английские (***StrDt = "1 Oct 1999": Data1 = CDate(StrDt)***).

Функция ***Weekday(Data1, vbMonday)*** покажет день недели для заданной даты. В этой функции задан первый день недели – понедельник, если опустить этот параметр, первым днем недели будет воскресенье (что соответствует английскому календарю).

Формат вывода информации подтипа дата-время можно определить с использованием функции ***FormatDateTime***, в которой существует 5 форматов (***vbGeneralDate, vbLongDate, vbShortDate, vbLongTime, vbShortTime***).

Название 7-го дня недели позволяет определить функция ***WeekDayName(7,False,vbUseSystem)*** – в данном случае при русских региональных настройках операционной системы вернет «воскресенье».

5.3.1.11 Работа с логическими выражениями

При написании логических условий используются операции сравнения данных (подтипов число, строка, дата и время):

= равно,

> не равно,

< меньше,

<= меньше или равно,

> больше,

>= больше или равно.

Кроме того, используются логические функции и операторы (их описание и синтаксис приведены в Приложении 10):

And Eqv Imp Is IsArray IsDate IsEmpty IsNull IsNumeric Not Or Xor.

Результатом выполнения логической операции является одно из двух возможных значений: ***True*** (*Истина*) или ***False*** (*Ложь*).

Переменной можно присвоить логическое значение ***True*** или ***False*** (*L1 =* ***True*** или *L1 = False*), но нельзя использовать русские значения *Истина Ложь* в присвоении и логических выражениях, не смотря на то, что русские версии Windows в окне сообщений показывают именно эти значения (см. рисунок 5.24).

При сравнении символьных и строковых значений учитывается регистр букв (прописные или строчные).

Результат вычисления логического выражения может быть присвоен переменной, показан в окне ***MsgBox,*** использован для организации разветвляющихся алгоритмов в операторах ***If…., Select Case***… и циклов в структурах ***Do [While | Until]…, While.***

Приоритет при вычислении выражений, в которых присутствуют логические компоненты:

1) арифметические операторы;
2) операторы объединения (конкатенации) строковых значений (**&, +**);
3) операции сравнения данных, которые используют символы **=**, **<>**, **<**, **<=**, **>**, **>=;**
4) логические операции ***And, Or, Not, Xor, Eqv, Imp.***

В операциях одного приоритета порядок вычислений слева-направо. Порядок может быть изменен при использовании круглых скобок.

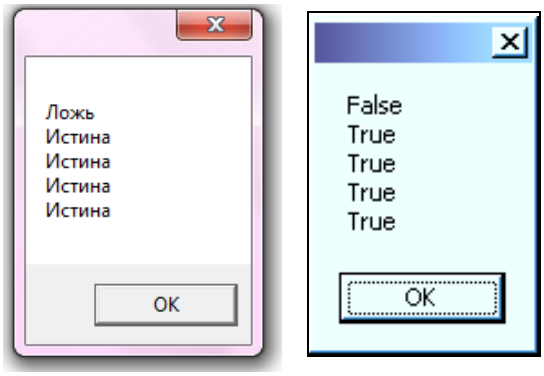


Рисунок 5.24. Результат вычисления логических функций в русской и английской версиях Windows

Пример программы с использованием логических выражений:

```
a = 1 > 10                          'False
b = "я" > "Я"                       'True
c = "Иванов" < "Петров"             'True
d = 1 < 10 and "я" > "Я"            'True
e = 1 > 10 or #31-12-2007# < #01-01-2008# 'True
MsgBox a & vbLF & b & vbLF & c & vbLF & d & vbLF & e
```

Результат выполнения приведенного выше текста программы показан на рисунке 5.24.

Краткое описание логических операций (здесь и в таблице 5.5 ***E1*** и ***E2*** – логические выражения.):

- ***And*** – логическое ***И*** (если выражения слева и справа от него истинны, результат ***True,*** иначе ***False*** или ***Null***);
- ***Or*** – логическое ***ИЛИ*** (должно быть истинным хотя бы одно из выражений);
- ***Not*** – логическое отрицание (возвращает ***True***, если условие ложно и наоборот);
- ***Xor*** – логическое исключение (выражение ***E1 Xor E2*** возвращает ***True***, если только ***E1 = True*** или только ***E2 = True***, иначе – ***False***);
- ***Eqv*** – эквивалентность двух выражений, возвращает ***True***, если они имеют одинаковое значение;
- ***Imp*** – импликация (***E1 Imp E2*** возвращает ***False***, если ***E1 = True*** и ***E2 = False***, иначе – ***True***).

Полное описание результатов вычисления логических операций приведено в таблице 5.5.

Таблица 5.5. Результаты логических операций

E1	*E2*	*E1 And E2*	*E1 Or E2*	*E1 Imp E2*	*E1 Xor E2*
True	True	True	True	True	False
True	False	False	True	False	True
True	Null	Null	True	Null	
False	True	False	True	True	True
False	False	False	False	True	False
False	Null	False	Null	True	
Null	True	Null	True	True	
Null	False	False	Null	Null	
Null	Null	Null	Null	Null	

5.3.1.12 Работа с объектами

Объект (в ООП) – оформленный по определенным правилам фрагмент программы, в котором описаны данные и процедуры для их обработки. Данные называются свойствами объекта, а процедуры – его методами.

Программный код объекта может быть сохранен в библиотеке классов. Каждый класс описывает переменные – свойства и процедуры – методы одного объекта. В программах объекты могут создаваться, как экземпляры библиотечных классов. В языке ***VBS*** можно использовать множество готовых библиотек классов.

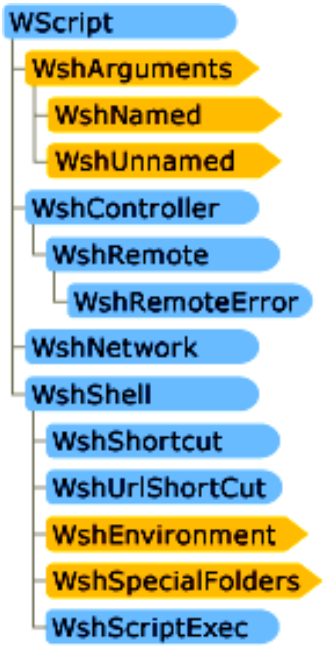


Рисунок 5.25. Объектная модель *Windows Script Host*

Программа на языке VBScript сама по себе является объектом ***WScript*** сервера сценариев *Windows Script Host*, (т. е. после запуска программы этот объект уже существует, не нужно давать команду о его создании). Объект WScript – корневой в объектной иерархии *Windows Script Host*, в которую входят четыре *COM*–библиотеки *(COM – Component Object Model)*: *WshShell, WshNetwork, WshController* (см. рисунок 5.25 с сайта http://msdn.microsoft.com/en-us/library/a74hyyw0(VS.85).aspx).

В программах на VBScript могут использоваться и другие библиотеки классов Windows (например, объекты библиотеки ***Microsoft ADO*** – классы для работы с базами данных и пр.), библиотеки других систем, поддерживающих интерфейс

ActiveX (OLE Automation), например, библиотеки Microsoft Offfice, Internet Explorer и пр.

Библиотеки классов могут содержать описания объектов, образующих иерархическую структуру.

Для создания в программе объекта (не ***Wscript***!) используется следующий синтаксис оператора присваивания:

```
Set <Переменная> = CreateObject("<Библиотека.Класс>")
```

С использованием этой функции создается переменная подтипа ***Object***, после чего можно узнать свойства созданного объекта и пользоваться его методами (процедурами и функциями объекта).

Свойство объекта имеет определенное значение, которое можно узнать или изменить.

Значение свойства объекта можно использовать в операторе присваивания и в любых других выражениях с использованием синтаксиса ***<Имя объекта>.<Имя свойства>*** (имя объекта и имя его свойства, разделенные точкой), например:

```
<Переменная> =  <Имя объекта>.<Имя свойства>
```

или

```
MsgBox <Имя объекта>.<Имя свойства>
```

В некоторых случаях свойство может возвращать ссылку на коллекцию объектов или значений, в этом случае возможно только присвоение этого свойства переменной с использованием слова ***Set***, как при создании объекта:

```
Set <Переменная> = <Имя объекта>.<Имя метода>
```

Чтобы изменить свойство объекта, достаточно присвоить ему какое-либо значение:

```
<Имя объекта>.<Имя свойства> = <Значение>
```

Присваиваемое значение может быть константой, выражением, свойством другого объекта, возвращаемым значением какого либо метода:

```
<Имя объекта>.<Имя свойства> = <Имя объекта2>.<Имя
метода([параметры])>
```

Метод объекта – это его процедура, которая может выполнять какие-либо действия, получать и возвращать значения (параметры метода).

При использовании метода с параметрами следует использовать следующий синтаксис:

1) если метод не возвращает значений:

<Имя объекта>.<Имя метода> <п1> [,<п2>, …]>

где ***п1, п2*** и т. д. – параметры;

2) если метод возвращает какое-либо значение (но не объект и не коллекцию объектов), обязательно использование круглых скобок:

<Переменная> = <Имя объекта>.<Имя метода>([<п1> [,<п2>, …]])

3) если метод возвращает ссылку на новый объект или их коллекцию – обязательно использование слова ***Set***, как при создании объекта:

Set <Переменная> = <Имя объекта>.<Имя метода> <п1> [,<п2>, …]>

При работе с объектами может использоваться структура ***With*** языка VBScript:

```
With <Имя объекта>
      операторы
End With
```

Для операторов между ***With*** и ***End With*** можно использовать методы и свойства заданного объекта, которые начинаются с точки (.) без указания предшествующего имени объекта, как показано далее в примерах программ.

Объект ***WScript*** имеет следующие методы: *Sleep Quit Echo CreateObject ConnectObject DisconnectObject GetObject*,

и свойства: *Arguments BuildVersion FullName Interactive Name Path ScriptFullName ScriptName Timeout Version StdIn StdOut StdErr* (последние 3 свойства можно использовать только для программы, запущенной с помощью *CScript.exe*, т. е. в окне *Cmd*).

Объект ***Wscript*** имеет подчиненный объект ***Shell***, с методами:

AppActivate CreateShortcut Exec ExpandEnvironmentStrings LogEvent Popup Run SendKeys и пр.,

и свойствами: *CurrentDirectory Environment SpecialFolders*. Описание этих объектов, их методов и свойств можно найти в справке системы Microsoft Visual Studio или на сайте корпорации Microsoft.

Методы объекта ***WScript*** позволяют не только управлять выполнением программы, но и устанавливать связь с другими объектами *ActiveX (OLE Automation)* – ***Wscript.ConnectObject(<ProgID>, <Prefix>)***, создавать объекты существующих библиотек – ***Wscript.CreateObject(<ProgID>, <Prefix>)***, а также создавать объект

из его описания в файле – ***Wscript.GetObject(<Pathname>, <ProgID>, <Prefix>)***.

Свойства данного объекта позволяют узнать параметры запущенной программы, кроме того свойством ***Wscript.Timeout*** можно задать максимальную продолжительность работы программы (в миллисекундах), а заданием значения свойства ***WScript.Interactive = False*** можно запретить вывод программой диалоговых окон.

Пример программы, сообщающей все свойства о самой себе (результат выполнения программы показан на рисунок 5.26):

```
With WScript
 .echo(.BuildVersion & vbLF &.FullName & vbLF & _
       .Interactive & vbLF  &.Name & vbLF & _
       .Path & vbLF &.ScriptFullName & vbLF & _
       .ScriptName & vbLF &.Timeout & vbLF &  _
      .Version)
End With
```

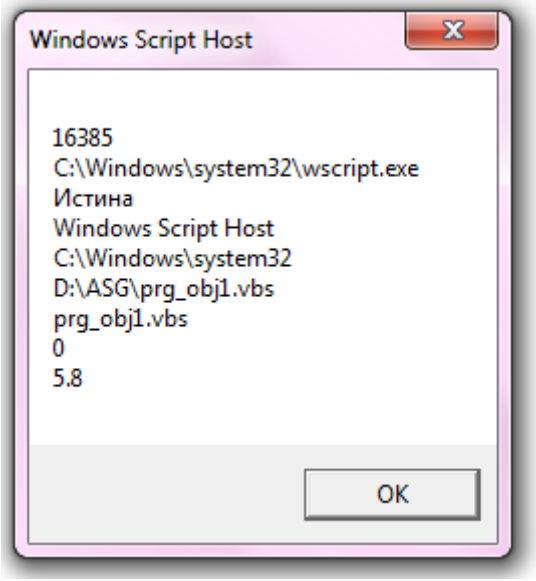


Рисунок 5.26. Свойства объекта ***WScript***

Пример программы, использующей метод WScript.Sleep, чтобы остановить работу программы на заданное количество миллисекунд (результат выполнения программы показан на рисунок 5.27):

```
t1 = time
WScript.Sleep 25545
t2 = time
h1 = Hour(t1)
m1 = Minute(t1)
s1 = Second(t1)
```

```
h2 = Hour(t2)
m2 = Minute(t2)
s2 = Second(t2)
ds = (h2-h1)*3600 + (m2-m1)*60 + (s2-s1)
MsgBox("Метод WScript.Sleep 25545" & vbLF & vbLF &_
  "Тнач = " & t1 & " Ткон = " & t2 & vbLF & vbLF &_
  "Программа проспала " & ds & " секунд"), _
   vbExclamation, "Использование метода WScript.Sleep"
```

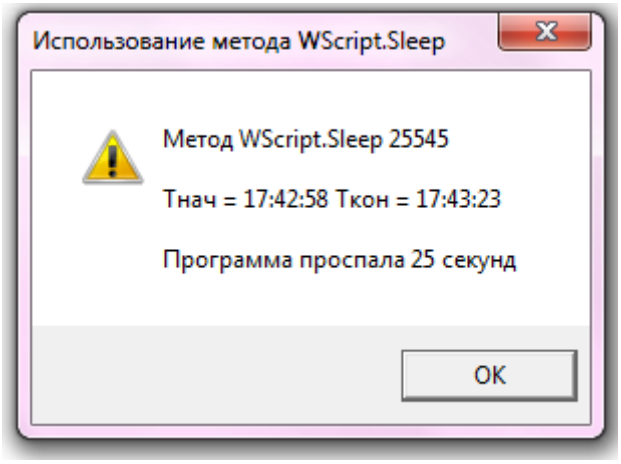


Рисунок 5.27. Использование метода ***WScript.Sleep***

Пример программы, использующей метод ***Popup*** объекта ***WScript.Shell*** – еще одна Windows-форма, которая отличается от ***MsgBox*** тем, что для него можно задать время существования, после чего оно закроется, если пользователь не нажал какую-либо его кнопку, с возвращаемым кодом -1 (результат выполнения программы показан на рисунок 5.28):

```
Set WS = CreateObject("WScript.Shell")
WS.Popup "Это окно сообщения, " & vbLF & _
  "которое закроется через 7 сек.," & vbLF & _
  "если Вы не нажмете какую-либо его кнопку",7, _
  "Окно Popup библиотеки WScript.Shell", 3+48
```

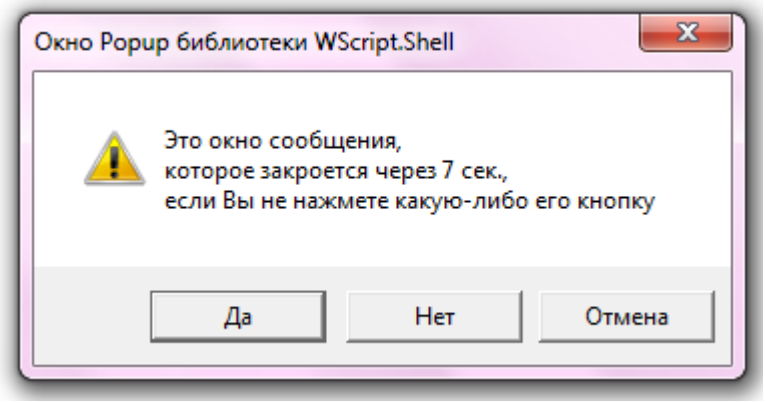


Рисунок 5.28. Использование метода ***Popup*** объекта ***WScript.Shell***

Работа с информацией файловой системы

Система Microsoft Visual Basic Scripting Edition для создания объектов, работающих с файловой системой использует библиотеку классов с именем ***Scripting*** (файл …\windows\system32\ *Scrrun.dll*).

Для дальнейшего понимания назначения объектов, их свойств и методов необходимо знание английской компьютерной терминологии.

Список объектов, которые могут быть созданы при работе с этой библиотекой, показан на рисунке 5.29. Полное описание объектов можно найти, например, в файле *...\Microsoft Office\Office12\1049\Vbscrip5.chm.*

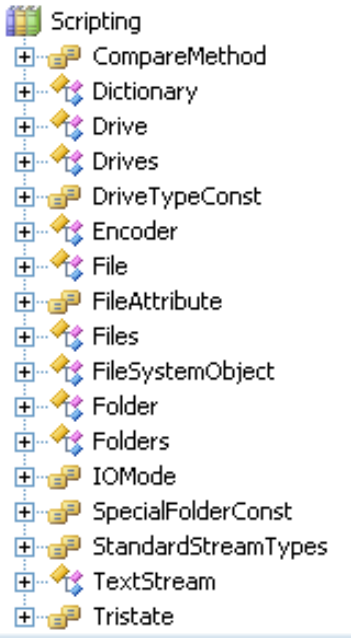


Рисунок 5.29. Библиотека классов ***Scripting***

Главный объект библиотеки классов ***Scripting*** системы Microsoft Visual Basic Scripting Edition – ***Scripting.FileSystemObject***, который имеет вложенные объекты ***Drive***, ***Folder***, ***File*** и ***TextStream.***

Методы объекта ***FileSystemObject***: *BuildPath CopyFile CopyFolder CreateFolder CreateTextFile DeleteFile DeleteFolder DriveExists FileExists FolderExists GetAbsolutePathName GetBaseName GetDrive GetDriveName GetExtensionName GetFile GetFileName GetFolder GetParentFolderName GetSpecialFolder GetTempName MoveFile MoveFolder OpenTextFile VarType.* Объект имеет одно свойство – *Drives*, возвращающее ссылку на **коллекцию** объектов ***Drive*** – дисковых устройств в системе.

Коллекция – упорядоченное множество однотипных объектов.

Любая коллекция имеет свойство *Count* (количество объектов в коллекции).

Ссылки на коллекции возвращает также метод *GetFolder:* на коллекцию ***SubFolders*** (объекты ***Folder***) и на коллекцию ***Files*** (объекты ***File***).

Свойства объекта ***Drive***: *AvailableSpace DriveLetter DriveType FileSystem FreeSpace IsReady Path RootFolder SerialNumber ShareName TotalSize VolumeName.* Методов объект не имеет.

Методы объектов ***Folder*** и ***File***: *Copy Delete Move OpenAsTextStream.*

Свойства объектов ***Folder*** и ***File***: *Attributes DateCreated DateLastAccessed DateLastModified Drive Name ParentFolder Path ShortName ShortPath Size Type.*

Методы объекта ***TextStream***: *Close Read ReadAll ReadLine Skip SkipLine Write WriteLine WriteBlankLines.*

Свойства объекта ***TextStream***: *AtEndOfLine AtEndOfStream Column Line.*

Далее приведены некоторые примеры использования объектов.

Пример 1. Информация о дисках компьютера (результат исполнения – рисунок 5.30).

```
On Error Resume Next   'на случай, если в CD-приводе нет диска
Dim fs, dr, ndr, d, d1, drv, s
Set fs = CreateObject("Scripting.FileSystemObject")
Set dr = fs.Drives
ndr = dr.Count
d = "Всего дисков - " & ndr & ": "
For Each d1 in dr
  name_d = d1.DriveLetter & ":"
  d = d & name_d  & " "
  Set drv = fs.GetDrive(fs.GetDriveName(name_d))
  s = s & vbLF & "Имя диска " & drv.Path & " - "
  s = s & drv.VolumeName
  Select Case drv.DriveType
   Case 0: t = "Неизвестный"
   Case 1: t = "Удаляемый (флэш и пр.)"
   Case 2: t = "Раздел винчестера"
   Case 3: t = "Сетевой"
   Case 4: t = "CD-ROM"
   Case 5: t = "RAM диск"
 End Select
  s = s & ", тип " & t & vbLF
  s = s & "Серийный номер диска " & _
      drv.SerialNumber & vbLF
  s = s & "Всего: " & _
      FormatNumber(drv.TotalSize/1024/1024, 0)
  s = s & " Мб,"
  s = s & " свободно: " & _
      FormatNumber(drv.FreeSpace/1024/1024, 0)
  s = s & " Мб" & vbLf
 Next
s = d & vbLf & s
MsgBox s,, "Это все FileSystemObject!"
```

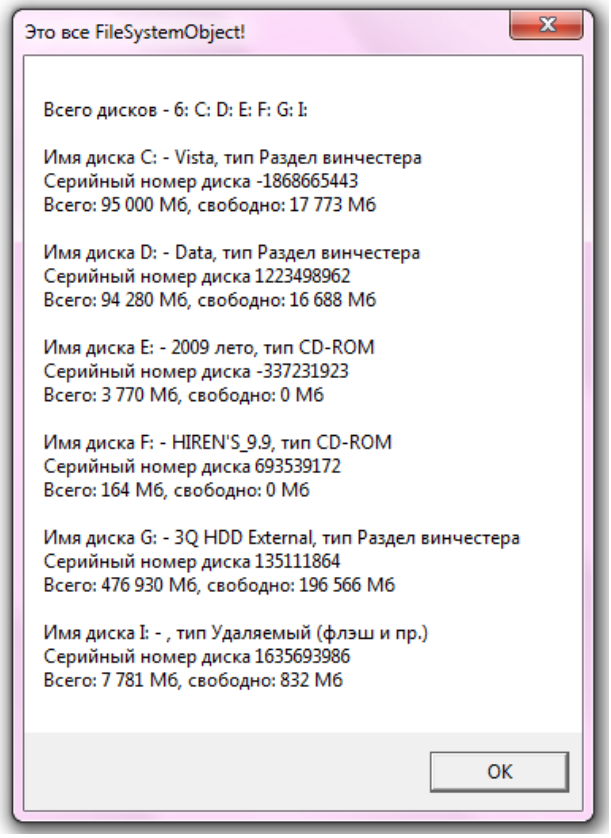


Рисунок 5.30. Пример 1

Пример 2. Информация о папках (результат исполнения – рисунок 5.31).

```
Dim fso, f, f1, fc, s
Set fso = CreateObject("Scripting.FileSystemObject")
s = "C:\inetpub"
Set f = fso.GetFolder(s)
Set fc = f.SubFolders       'коллекция объектов SubFolders
s = s & ", папки:  " & vbLF
For Each f1 in fc
  s = s & f1.name & vbLF     'папки в коллекции SubFolders
Next
MsgBox s,, "Коллекция SubFolders"
```

Пример 3. Информация о файлах и их типах в папке (результат исполнения – рисунок 5.32).

```
Dim fso, f, f1, fc, s
Set fso = CreateObject("Scripting.FileSystemObject")
Set f = fso.GetFolder("C:\inetpub\wwwroot")
Set fc = f.Files       'коллекция объектов Files
```

```
For Each f1 in fc
      s = s & f1.name & "  тип: " & f1.Type & vbLF
Next
MsgBox s,, "Коллекция Files"
```

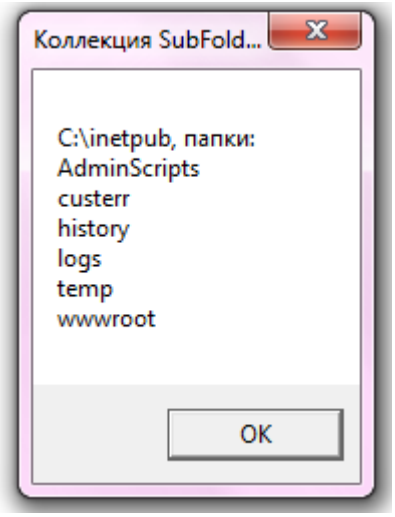


Рисунок 5.31. Пример 2

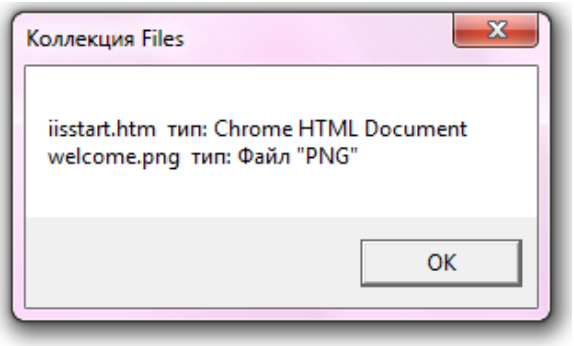


Рисунок 5.32 Пример 3

Пример 4. Чтение информации о файле (результат исполнения – рисунок 5.33).

```
Dim fso, fld, F_name, f, s
Set fso = CreateObject("Scripting.FileSystemObject")
F_name = "D:\ASG\Microsoft Office 2010 - обзор.htm"
Set f = fso.GetFile(F_name)      'Создан объект File
s = "Диск - " & f.Drive.DriveLetter & vbLf
s = s & "Путь - " & f.Path & vbLf
s = s & "Путь в формате 8.3 - " & f.ShortPath & vbLf
s = s & "Имя - " & f.Name & vbLf
s = s & "Короткое имя - " & f.ShortName & vbLf
s = s & "Размер - " & f.Size & " байт" & vbLf
s = s & "Тип файла - " & f.Type & vbLf
s = s & "Дата создания - " & f.DateCreated & vbLf
s = s & "Дата последнего доступа - " & _
        f.DateLastAccessed & vbLf
s = s & "Дата последней модификации - " & _
        f.DateLastModified
MsgBox s,, "Свойства файла"
```

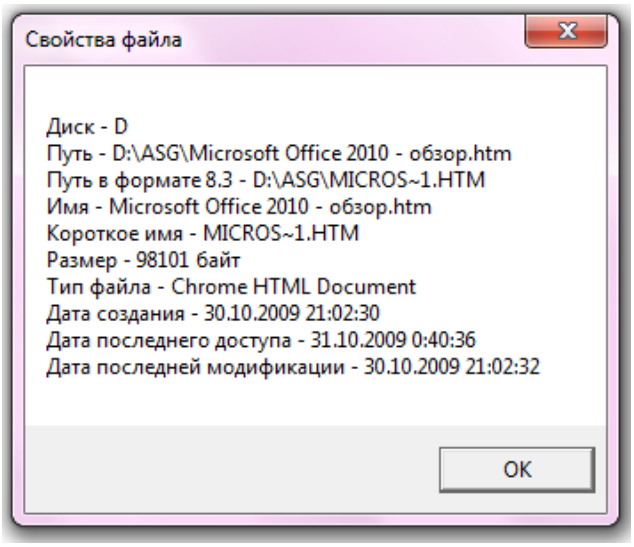


Рисунок 5.33. Пример 4

Пример 5. Чтение строк из файла (результат исполнения – рисунок 5.34).

```
Dim fso, F_name, f_txt, s
Set fso = CreateObject("Scripting.FileSystemObject")
F_name = "D:\VBScript\Prg_obj5.vbs"
Set f_txt = fso.OpenTextFile(F_name, 1)
Do While f_txt.AtEndOfStream <> True
    s = s + f_txt.ReadLine + vbLF
Loop
f_txt.Close
MsgBox S,, "Чтение строк из файла"
```

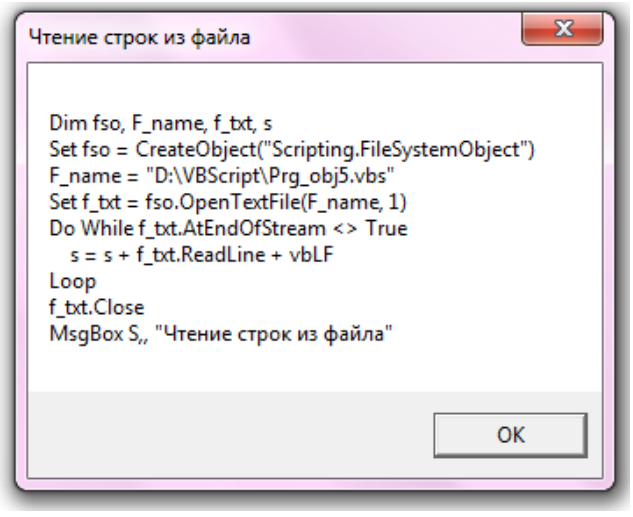


Рисунок 5.34. Пример 5

Пример 6. Создание папки, текстового файла и запись в файл:

```
Dim fso, fld, F_name, f_txt, ns, s
s = ""
Set fso = CreateObject("Scripting.FileSystemObject")
fld = "C:\vbscripts\"
If Not (fso.FolderExists(fld)) Then
   fso.CreateFolder(fld)
   s = "Создана папка " & fld & "!" & vbLF
End If
F_name = fld & "vbs_test1.txt"
If Not (fso.FileExists(F_name)) Then
   Set f_txt = fso.CreateTextFile(F_name, True)
   f_txt.WriteLine("1. Файл " & F_name & " создан!")
   s = s & "Создан файл " & F_name & "!" & vbLF
   s = s & " о чем записано в 1-ой строке файла!" & vbLF
End If
Set f_txt = fso.OpenTextFile(F_name, 1)
                'Файл открыт для чтения, как объект TextStreem
f_txt.ReadAll
ns = f_txt.Line    'Номер последней строки минус 1
Set f_txt = fso.OpenTextFile(F_name,8)
                    'Открыт для дозаписи в конец файла
f_txt.WriteLine(ns & "-я строка текстового файла.")
f_txt.Close
s = s & "В конец файла " & F_name & vbLF
s = s & "  добавлена " & ns & "-я строка"
s = s & vbLF & vbLF & "Всего строк в файле - " & ns
MsgBox s,, "Добавление строк в файл"
```

После запуска на исполнение этой программы первый раз (при отсутствии папки и файла) будет показано сообщение, приведенное на рисунке 5.35. После запуска на исполнение программы 6 три раза файл **`vbs_test1.txt`** будет содержать текст:

1. Файл C:\vbscripts\vbs_test1.txt создан!
2-я строка текстового файла.
3-я строка текстового файла.
4-я строка текстового файла.

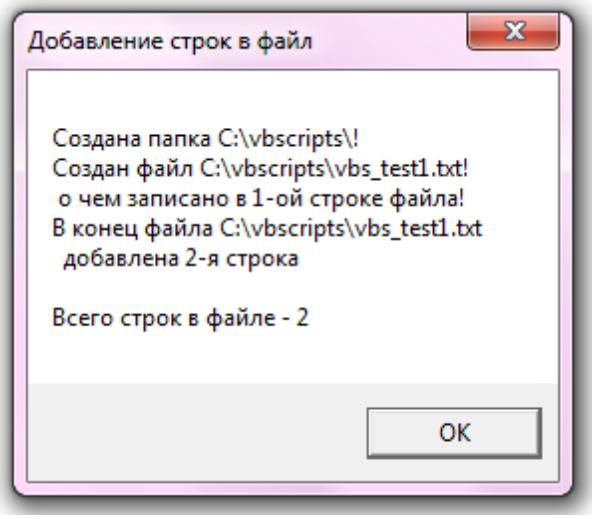


Рисунок 5.35. Пример 6

Пример 7. Копирование файла:

```
Const OverwriteExisting = TRUE
Set fso = CreateObject("Scripting.FileSystemObject")
fso.CopyFile "C:\vbscripts\vbs_test1.txt",_
     "C:\vbscripts\vbs_test2.txt ", OverwriteExisting
```

Результатом работы программы будет новый файл ***vbs_test2.txt***. Если запустить программу повторно, старый файл ***vbs_test2.txt*** будет заменен новым (т. к. параметр ***OverwriteExisting = TRUE***, если задать ***OverwriteExisting =FALSE*** появится сообщение об ошибке «File already exist» – «Файл уже существует»).

Работа с информацией об ошибках

Если написать в начале программы структуру

```
On Error Resume Next
```

автоматическое завершение работы программы при возникновении ошибки будет предотвращено и появится возможность выполнить некоторые дополнительные действия в программе с использованием объекта ***Err*** языка VBScript, который автоматически создается при работе программы и значение свойства *Number* которого в этом случае станет не равным нулю (программная обработка ошибок).

Методы объекта ***Err***: *Clear Raise.* Свойства объекта ***Err***: *Description HelpContext HelpFile Number.*

Пример программы с обработкой ошибки (дополненный пример 7):

```
On Error Resume Next
Const OverwriteExisting = False
Set fso = CreateObject("Scripting.FileSystemObject")
fso.CopyFile "C:\vbscripts\vbs_test1.txt", _
       "C:\vbscripts\vbs_test2.txt ", OverwriteExisting
IF Err.Number <> 0 Then      'файл уже существует
   MsgBox "Файл уже существует и не может быть" & _
       "заменен новым", vbCritical,"Сообщение об ошибке"
End If
```

Дополнение программы приведенного ранее Примера 5 обработкой ошибок:

```
On Error Resume Next
Dim fso, F_name, f_txt, s
Set fso = CreateObject("Scripting.FileSystemObject")
F_name = "C:\vbscripts\vbs_test1.txt"
Set f_txt = fso.OpenTextFile(F_name, 1)
```

```
If Err.Number = 0 Then          'файл существует и открыт
                                'далее – цикл чтения строк из файла
  Do While f_txt.AtEndOfStream <> True
    s = s + f_txt.ReadLine + vbLF
  Loop
  f_txt.Close
  MsgBox S,, "Чтение строк из файла"
Else
 MsgBox ("Ошибка № " & CStr(Err.Number) & vblf  _
         & Err.Description & vblf & Err.Source)
 Err.Clear     ' очистка свойств объекта Err
End If
```

Впрочем, данная ошибка может быть выявлена и другим способом: … `If IsObject(f_txt) Then` *'объект создан …*

Работа с информацией локальной сети

Для этой цели может использоваться объект ***WScript.Network***, имеющий следующие свойства: *ComputerName UserName UserDomain* и методы: `EnumNetworkDrives EnumPrinterConnections MapNetworkDrive RemoveNetworkDrive AddPrinterConnection AddWindowsPrinterConnection RemovePrinterConnection SetDefaultPrinter.`

Пример использования объекта ***WScript.Network***:

```
On Error Resume Next
Set Wnet = WScript.CreateObject("WScript.Network")
S = "Домен = " & Wnet.UserDomain & vbLF
S = S & "Имя компьютера = " & Wnet.ComputerName & vbLF
S = S & "Пользователь = " & Wnet.UserName
Net_f = "\\lks\asg"
Wnet.MapNetworkDrive "I:", Net_f,"True"
If Err.Number <> 0 Then      'сетевая папка не найдена
 MsgBox S & vbLF & "Сетевая папка " & Net_f & _
   " не найдена",vbCritical,"Ошибка..."
End If
Set NetDrives = Wnet.EnumNetworkDrives
If not isObject(NetDrives) Then
  MsgBox "Нет сетевых дисков",vbExclamation,_
        "Нет данных..."
Else
 i = 0
 S = S & vbLF & "Количество сетевых дисков = " _
       & NetDrives.Count/2 & vbLF
 While i <= NetDrives.Count-1
    S = S & NetDrives.Item(i) & " - " & _
        NetDrives.Item(i+1) & vbLF
```

```
    i = i+2
 Wend
 MsgBox S, vbInformation, "Информация о компьютере"
End If
```

Пример работы программы показан на рисунке 5.36.

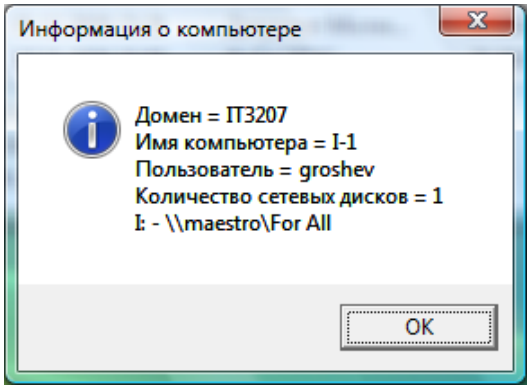


Рисунок 5.36. Пример работы с объектом ***WScript.Network***

Работа с объектами Windows OLE Automation *(Microsoft ActiveX)*

Данная возможность используется в целях администрирования серверов, компьютеров и их сетей, имеет неограниченные возможности, и здесь подробно рассматриваться не будет. Большое количество примеров использования можно найти на сайте корпорации Microsoft (*http://www.microsoft.com/technet/scriptcenter/*).

Приведем лишь примеры использования для объектов системы Microsoft Office.

Пример 1. Программа открывает существующий файл системы Microsoft Office Word (c:\vbscripts\test_prg.doc) и добавляет в него новые строки текста:

```
Const END_OF_STORY = 6
Const MOVE_SELECTION = 0
Set objWord = CreateObject("Word.Application")
objWord.Visible = True
Set objDoc = _
  objWord.Documents.Open("c:\vbscripts\test_prg.doc")
Set objSelection = objWord.Selection
objSelection.EndKey END_OF_STORY, MOVE_SELECTION
objSelection.TypeParagraph()
objSelection.Font.Size = "14"
objSelection.TypeText "Это первая строка, которую" & _
                      " VBScript добавила в документ"
```

```
objSelection.TypeParagraph()
objSelection.TypeText "Это еще одна строка, " & _
             " которую VBScript добавила в документ"
objSelection.TypeParagraph()
objSelection.TypeText "Сегодняшняя дата " & Date()
objSelection.TypeParagraph()
```

Пример работы программы показан на рисунке 5.37.

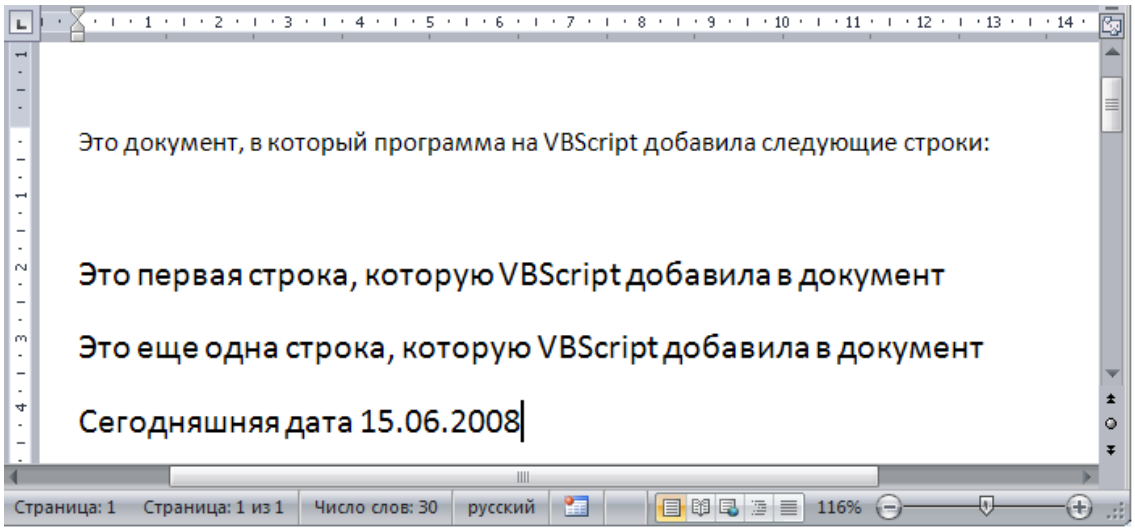


Рисунок 5.37. Пример работы с объектом ***Word.Application***

Пример 2. Программа открывает новый файл системы Microsoft Office Excel и добавляет в его ячейки, текст, числа и расчетную формулу:

```
Set objExcel = CreateObject("Excel.Application")
With objExcel
  .Visible = True
  .Workbooks.Add
  .Cells(1, 1).Value = "Тест работы VBScript с Excel"
  .Cells(2, 1).Value = 1.111
  .Cells(3, 1).Value = 2.222
  .Cells(4, 1).Value = "=A2+A3"
End With
```

Результат работы программы показан на рисунке 5.38.

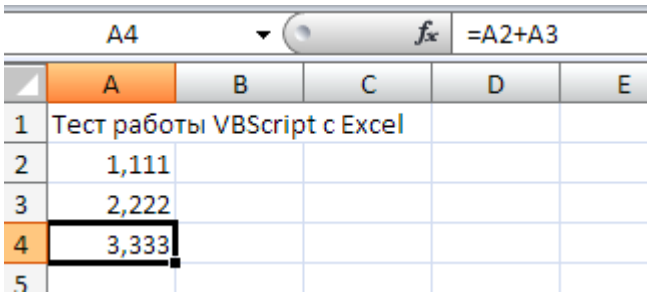


Рисунок 5.38. Пример работы с объектом ***Excel.Application***

Пример 3. Программа устанавливает связь с базой данных системы Microsoft Office Access (используется база данных, описанная в разделе 3.3.4), выполняет запрос на отбор дан-

ных и показывает их в окне, приведенном на рисунке 5.39:

```
Dim Conn, rs, S
Set Conn = CreateObject("ADODB.Connection")
Conn.Open "PROVIDER=Microsoft.ACE.OLEDB.12.0;" & _
          "DATA SOURCE= D:\ASG\Контингент.accdb"
Set rs = CreateObject("ADODB.Recordset")
Set rs = Conn.Execute("SELECT NZ, FIO FROM Список" & _
                      " WHERE N_GRUP=""591""")
S="№ зач.   Фамилия, имя, отчество" & vbLF
S = S & "----------------------------------------" & vbLF
Do Until rs.EOF
    S = S & rs("NZ") & "   " & rs("fio") & vbLF
    rs.MoveNext
Loop
msgbox S,,"Данные таблицы Список"
If rs.State = adStateOpen then rs.Close
If Conn.State = adStateOpen then Conn.Close
```

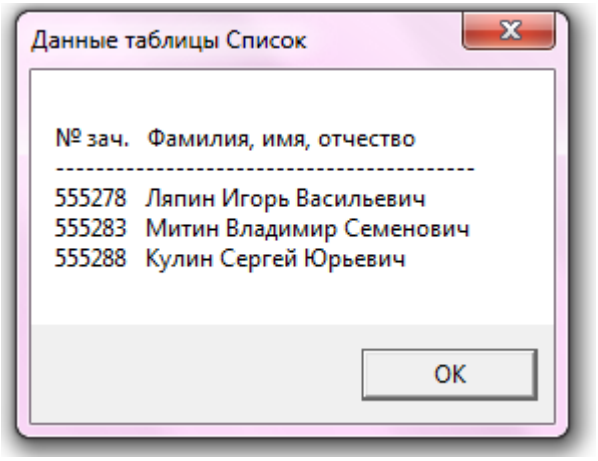


Рисунок 5.39. Пример работы с объектами ***ADODB.Connection*** и ***ADODB.Recordset***

5.3.1.13 Использование скриптов на HTML-страницах

В разделе 4.12 данного учебного пособия приведены некоторые сведения о языке гипертекстовой разметки HTML, который позволяет создавать сложные структурированные красиво оформленные web-страницы. Указывалось также, что на этих страницах могут использоваться программные компоненты – скрипты на языках сценариев, таких, как Java Script, VBScript и TCL для создания динамических страниц и использования HTML как средства создания сетевых приложений.

Сценарий на странице должен располагаться внутри пары тэгов ***<SCRIPT> и </SCRIPT>.*** По умолчанию языком сценария считается JavaScript, при использовании языка VBScript следует использовать указание в тэге ***<SCRIPT LANGUAGE="VBSCRIPT">.***

Любая программа на языке приведенная выше, может исполняться и вызовом ее через Интернет-браузер, если файл программы сохранять не с расширением *.vbs, а как *.htm или *.html.

Простейший пример скрипта, который будет выполнен программой Internet Explorer, если его написать в Блокноте и сохранить с именем *My_script1.html*:

```
<SCRIPT LANGUAGE="VBScript">
  MsgBox ("Привет, WWW!")
</SCRIPT>
```

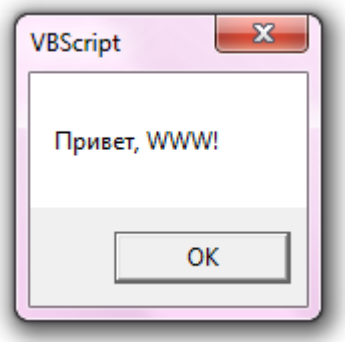


Рисунок 5.40. Простейший скрипт

Открыв файл *My_script1.html* двойным щелчком мыши (можно также открыть его, как и файлы *.vbs, из текстовых редакторов *EmEditor* и *Aditor*), мы увидим пустую web-страницу и созданную ей Windows-форму, показанную на рисунке 5.40. При этом необходимо разрешить исполнение скриптов на web-странице.

Сценарий может быть расположен в любом месте HTML-документа, но обычно он располагается в начале или в конце документа.

Пример HTML-страницы с математическими вычислениями на ней, выполняемыми скриптом (результат работы программы показан на рисунке 5.41):

```
<HTML>
  <HEAD>
    <TITLE>Работа с VBScript</TITLE>
    <SCRIPT LANGUAGE="VBScript">
      Sub Calc_OnClick()
        document.frm.stoimost.value  = _
          Round(document.frm.Kolich.value _
          * Document.frm.cena.value,2)
      End Sub
    </SCRIPT>
  </HEAD>
  <BODY>
    <H3>Пример использования VBScript на HTML-странице</H3>
    <P>Задайте количество и цену
       и нажмите кнопку "Расчет стоимости" </P>
    <FORM NAME="frm">
```

```
        <TABLE>
        <TR>
          <TD> Количество</TD>
          <TD> Цена</TD>
          <TD> Стоимость</TD>
        </TR>
        <TR>
         <TD><INPUT TYPE="Text" NAME="Kolich" SIZE=12></TD>
         <TD><INPUT TYPE="Text" NAME="cena" SIZE=12></TD>
         <TD><INPUT TYPE="Text" NAME="stoimost" SIZE=12></TD>
        </TR>
        </TABLE>
      <BR>
<INPUT TYPE="Button" NAME="Calc" VALUE="Расчет стоимости">
</FORM>
</BODY>
</HTML>
```

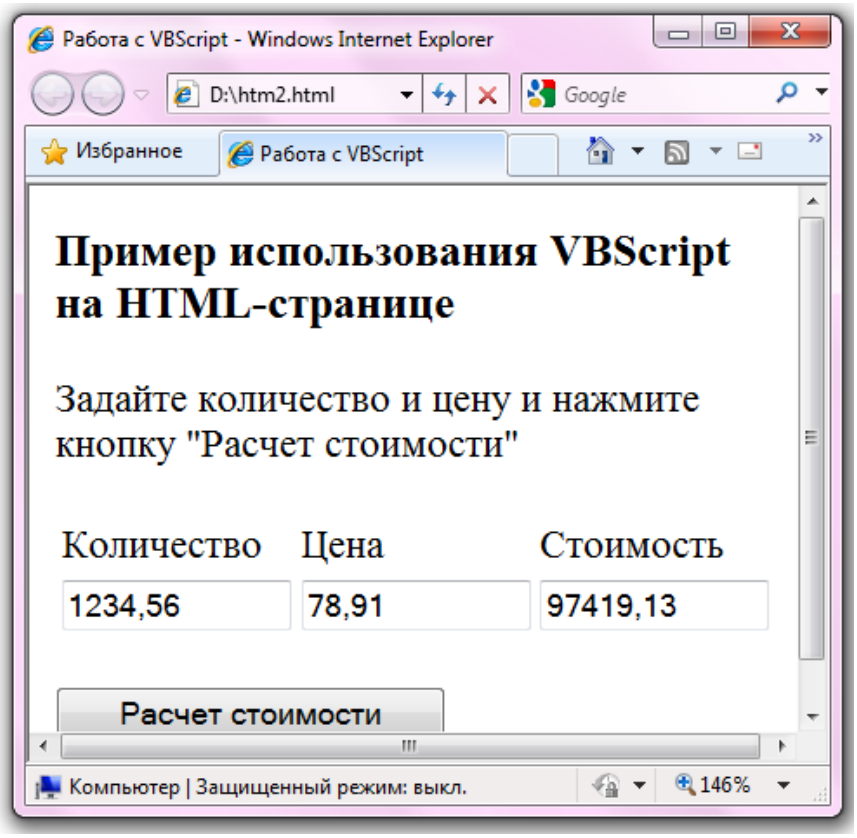


Рисунок 5.41. Использование *VBScript* на HTML-странице

5.3.2 Язык Microsoft Visual Basic for Applications

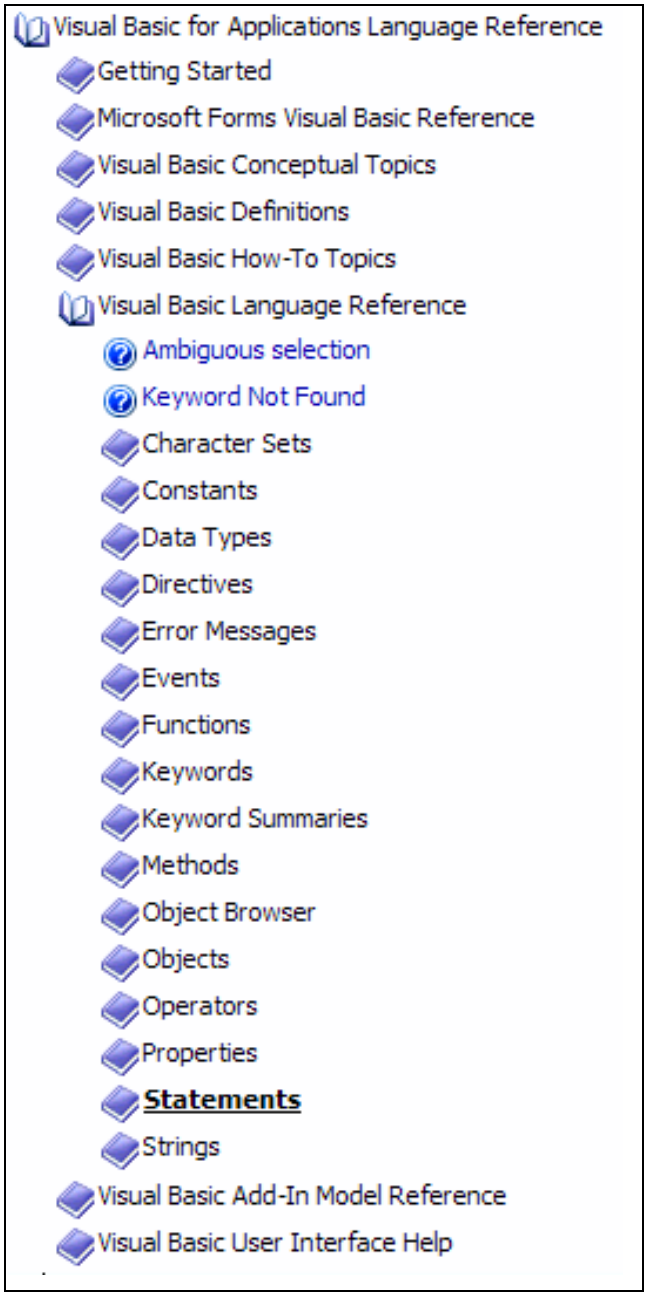


Рисунок 5.42. Оглавление справки по языку VBA Microsoft Office Word

Microsoft Visual Basic for Applications (VBA) является встроенным языком программирования для системы Microsoft Office (может использоваться в Word, Excel, Access, PowerPoint, Outlook, FrontPage, InfoPath), а также и в некоторых других системах (Microsoft Visio и Project, CorelDRAW, CorelWordPerfect Office 2000, AutoCAD).

VBA имеет очень много общего с VBScript. Различия можно найти в файле документации *…\Microsoft Office\Office12\1049\ Vbscrip5.chm*, раздел «Visual Basic for Applications Features not in VBScript» – особенности VBA, которых нет в VBScript. Текст этого раздела в переводе на русский язык показан в таблице 5.6.

Существенные отличия – в правилах написания имен и в работе с типами данных:

1) В именах (идентификаторах) в языке VBA в русской версии Microsoft Office могут использоваться русские буквы. Нельзя использовать пробел, точку, символы !, @, &, $, #. Первый символ в имени – обязательно буква.
2) Переменные и массивы могут быть объявлены с указанием их типа, как показано в описании синтаксиса массивов в таблице 5.6.

Типы данных – те же, что и подтипы для VBScript (см. таблицу 5.4). Кроме того, добавлен тип ***Decimal*** (14 байт в памяти, целое число в диапазоне +/-79 228 162 514 264 337 593 543 950 335, или действительное число в диапазоне +/-7.9228162514264337593543950335 (28 знаков в дробной части), или наименьшее ненулевое число +/-0.0000000000000000000000000001. Особенность этого типа – его нельзя использовать в описании данных, в этот тип можно преобразовать тип ***Variant*** с использованием функции ***Cdec()***.

Полную справку по языку VBA (на английском языке даже в русских версиях Office) можно получить, открыв окно редактора Visual Basic из приложений Microsoft Office (например, работая в системе Word, нажать клавиши Alt+F11, или, найдя эту команду на вкладке ленты *Разработчик*), и нажав клавишу F1. Раздел справки по VBA показан на рисунке 5.42.

Эта же справочная система содержит описание общих объектов самой системы – *Object Model Reference for the 2007 Microsoft Office System* и специфичных объектов той системы, в которой используется редактор Visual Basic.

Так, для редактора Word присутствует раздел *Word Object Model Reference (справочник по объектам Word)*, для Excel – *Excel Object Model Reference*, для Access – *Access Object Model Reference*, *Microsoft Data Access Objects (DAO) Reference*, *Microsoft ActiveX Data Objects (ADO) Reference* и *Microsoft Access SQL Reference*, для PowerPoint – *PowerPoint Object Model Reference* и т. д.

Таблица 5.6. Отличия языка VBA от VBScript

Категория	Ключевые отличия
Select Case	Выражения могут использовать ключевой слово ***Is*** Выражения могут использовать ключевой слово ***To*** для задания диапазона значений
Традиционный файловый ввод/вывод Basic	Открыть файл: ***Open*** *<pathname>* ***For*** *mode [**Access** access] [lock]* ***As*** *[**#**]filenumber [**Len**=reclength]*, где *mode* - ***Append, Binary, Input, Output,*** *or* ***Random;*** *access* - ***Read, Write,*** *or* ***Read Write;*** Чтение из файла: ***Input #****filenumber, varlist* Запись в файл: ***Print #****filenumber, [outputlist]*
Date и ***Time***	Задают текущую системную дату и время
Динамический обмен данными (DDE)	***LinkExecute, LinkPoke, LinkRequest, LinkSend***

Продолжение таблицы 5.6

Категория	Ключевые отличия
Доступ к элементам объектов с !	Например, MyCollection!Count
Перечисления ***Enum***, в отличии от коллекций каждый элемент перечисления имеет имя	***[Public | Private] Enum*** *<name>* <membername> [= <constantexpression>] ... **End Enum** Имеют методы Add, Remove и свойства Count, Item. Обращение к методам и свойствам с использованием символа **!**: *<name>!*<membername>
Массивы	***Индекс с любой заданной нижней границей:*** ***Dim [WithEvents] <varname>[([subscripts])] [As [New] <type>] [, [WithEvents] <varname>[([subscripts])][As [New] <type>]]...*** ***Где subscripts имеет синтаксис:*** ***[<lower> To] <upper> [, [<lower> To] <upper>]...***
Обработка ошибок	***Erl, Error, Resume, Resume Next***
Объекты	***Clipboard, Collection***
Объявления	***Declare*** (для библиотек DLL) ***Optional*** ***ParamArray*** ***Static***
Операции	***Like***
Отладка	***Debug.Print*** ***End, Stop***
Параметры	***Option Base, Option Compare, Option Private Module***
Задание типов для имен, начинающихся с заданных букв	***Deftype*** Например, ***DefStr P-T*** ***DefInt I-N*** ***DefDec A-F***
TypeOf для сравнения типов объектов	***If TypeOf MyControl Is CommandButton Then*** …
Type используется для описания типа пользователя	***[Private | Public] Type*** *<varname>* *<elementname>* ***[([subscripts])] As*** *<type1>* ***[<****elementname>* ***[([subscripts])] As*** *<type2>]* *...* ***End Type***

Продолжение таблицы 5.6

Категория	Ключевые отличия
Строки	Фиксированной длины ***LSet, Rset, StrConv*** Оператор ***Mid***
Управление потоками	***DoEvents*** ***GoSub...Return, GoTo*** ***On Error GoTo*** ***On...GoSub, On...GoTo*** ***Line numbers, Line labels***
Условная компиляция	***#Const*** ***#If...Then...#Else***

Далее на примерах будет показано использование языка VBA в различных компонентах *Microsoft Office System*.

Для понимания смысла свойств и методов объектов необходимо знание английского языка.

5.3.2.1 Программирование на языке VBA в Microsoft Office Word

На странице документа Word можно организовать работу с достаточно сложной программой, даже не создавая для этого отдельных Windows-форм, запускающихся, например, при открытии документа или при нажатии на кнопку на странице документа, а просто разместив на нем поля ввода данных и показывая программно на той же странице результаты после изменения данных.

Однако, многие математические расчеты проще выполнить в системе Excel, при хранении информации в базах данных очень большие возможности обработки этой информации имеет Access. При работе с базами данных в крупных информационных системах может быть предусмотрено формирование отчетов с выводом их в Word или Excel, при этом разработку таких программ может выполнить лишь специалист, хорошо знакомый как с архитектурой базы данных информационной системы, так и с языком VBA.

Некоторые принципы работы с объектами Word можно освоить, если записать некоторую последовательность своих действий с помощью средства Word ***Запись макроса***, которое присутствует на вкладке ленты ***Разработчик***. Данная вкладка присутствует, если в ***Параметрах Word*** на странице ***Основные*** стоит галочка у пункта ***Показывать вкладку Разработчик на ленте***.

Выполним следующие действия:

1) нажмем на кнопку ***Запись макроса***, ни кнопку, ни клавиши макросу можно не назначать, зададим, где будет сохраняться текст макроса: ***Макрос доступен*** для текущего документа;
2) переместимся в конец документа с текущей позиции (была не в конце), нажав клавиши *Ctrl+End*;
3) нажмем клавишу *Enter* и напишем слово «Привет!»;
4) еще раз повторим операцию пункта 3;
5) выделим последнее слово, нажав Shift+Home и нажмем клавишу *Delete*;
6) перейдем в начало первой строки «Привет!», нажмем Shift+End и затем клавишу *Delete*;
7) на вкладке **Разработчик** сначала нажмем кнопку ***Остановить запись***, затем кнопку ***Visual Basic***;
8) в открывшемся окне редактора **Visual Basic** слева на панели **Project** сделаем двойной щелчок мышкой на разделе *Progect*(<текущий файл>) – *Modules* – *NewMacros* и увидим справа окно с текстом программы на языке VBA, которое показано на рисунке 5.43.

При сохранении документа, в котором присутствуют программные модули, в системе Microsoft Office 2007 следует выбрать тип файла *Документ Word с поддержкой макросов*, иначе текст программ не будет сохранен.

В тексте макроса на рисунке 5.43 ***Selection*** – один из основных объектов, которые содержит главный объект ***Application*** – приложение Word.

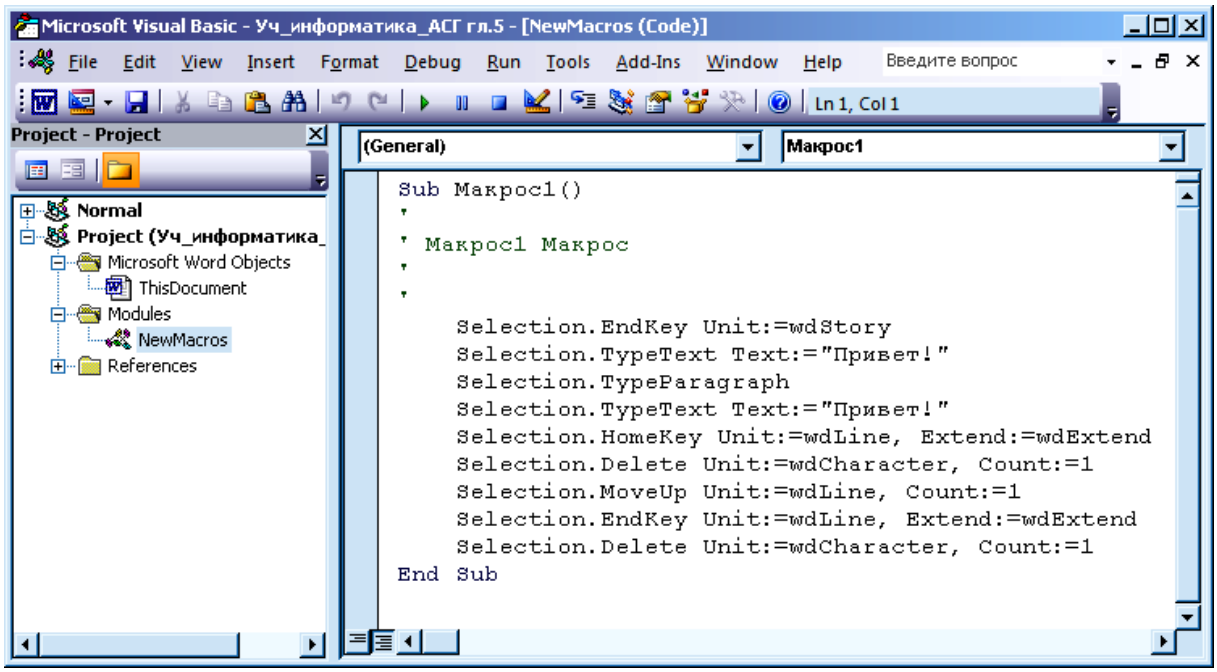


Рисунок 5.43. Текст макроса на языке VBA

При написании текста программ можно использовать всплывающий список свойств и методов, который появляется для стандартных объектов в редакторе **Visual Basic** после написания имени объекта и нажатия после неё точки (см. рисунок 5.44)**,** полное описание свойств и методов объектов присутствует в справочной системе.

Всплывающий список свойств и методов можно получить также, нажав комбинацию клавиш Ctrl+пробел.

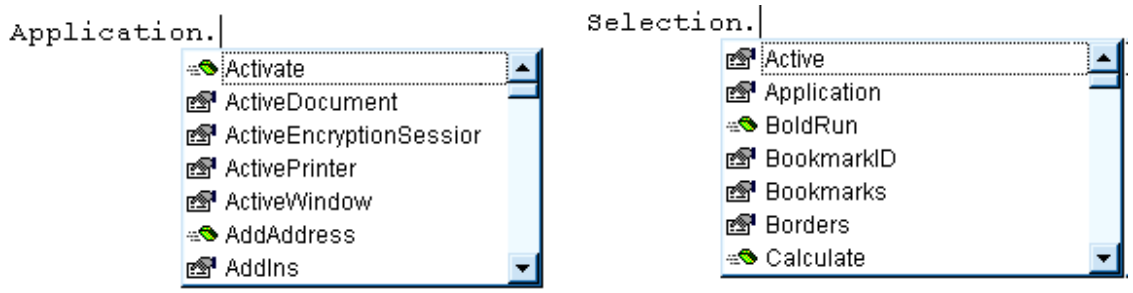


Рисунок 5.44. Использование всплывающих списков свойств и методов объектов (появляются после нажатия точки после имени объекта)

Кроме того, при написании текста программы для стандартный процедур и функций появляются всплывающие подсказки по их синтаксису (см. рисунок 5.45).

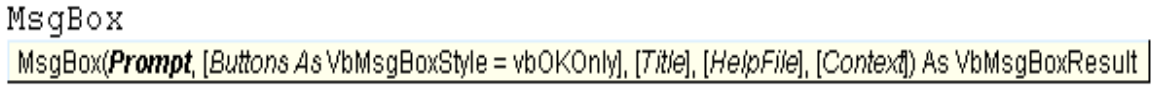


Рисунок 5.45. Всплывающая подсказка в редакторе VBA

В тексте Макрос1 используется объект ***Selection*** – место, где находится курсор в текущем документе объекта ***Application*** или выделенный фрагмент документа. Полный синтаксис обращения к этому объекту: ***Application.Selection***…, но имя родительского объекта можно опустить, так как при работе с документом в системе Word объект ***Application*** всегда является текущим активным объектом.

Selection.EndKey – метод объекта, полный синтаксис которого следующий:

```
Selection.EndKey(Unit, Extend)
```

где:

Unit – фрагмент документа, в конец которого будет перемещен курсор; по умолчанию имеет значение **wdLine** (строка документа), могут быть заданы другие значения:

wdCell	– ячейка таблицы
wdCharacter	– символ текста
wdCharacterFormatting	– символ форматирования
wdColumn	– колонка таблицы
wdItem	– выбранный компонент документа
wdParagraph	– абзац
wdParagraphFormatting	– форматирование абзаца
wdRow	– столбец таблицы
wdScreen	– размеры экрана
wdSection	– секция
wdSentence	– предложение
wdStory	– основной текст документа
wdTable	– таблица
wdWindow	– окно
wdWord	– слово

Extend – стиль перемещения курсора, по умолчанию имеет значение **wdMove** – простое перемещение без выделения фрагмента, может быть задано другое значение **wdExtend** – перемещение с выделением фрагмента текста.

Текст программы *Макрос1* с комментариями:

```
Selection.EndKey Unit:=wdStory
    ' переместить курсор в конец всего документа
Selection.TypeText Text:="Привет!"
    ' написать текст
Selection.TypeParagraph
    ' перейти к новому абзацу
Selection.TypeText Text:="Привет!"
    ' написать текст
Selection.HomeKey Unit:=wdLine, Extend:=wdExtend
' выделить фрагмент текста от текущей позиции до начала строки
Selection.Delete Unit:=wdCharacter, Count:=1
    ' удалить выделенный фрагмент
Selection.MoveUp Unit:=wdLine, Count:=1
    ' переместить курсор вверх на одну строку
Selection.EndKey Unit:=wdLine, Extend:=wdExtend
    ' выделить строку
Selection.Delete Unit:=wdCharacter, Count:=1
    ' удалить строку
```

Выполнить макрос можно, если на вкладке ленты Word **Разработчик** выбрать команду **Макросы**, затем в списке выбрать Макрос1 и нажать кнопку **Выполнить**. Две строки текста очень быстро появятся и исчезнут.

Работу макросов можно проследить с использованием отладочных режимов в окне Visual Basic (рисунок 5.47).

В пункте меню **Debug** (см. рисунок 5.46) и на панели инструментов присутствуют команды:

Step Into – выполнить построчно с заходом из текущей процедуры во все вызываемые процедуры;

Step Over – выполнить построчно без захода из текущей процедуры в вызываемые процедуры;

Step Out – выполнить остающиеся строки с заходом из текущей процедуры во все вызываемые процедуры;

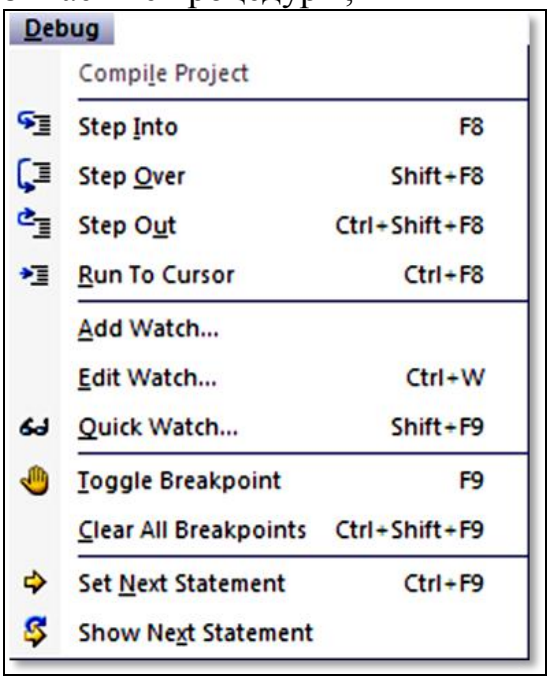


Рисунок 5.46. Пункты меню
Debug системы VBA

Run To Cursor – выполнить до позиции курсора в текущей процедуре;

Add Watch – добавить объект или переменную в окно просмотра **Watches**.

Toggle Breakpoints – задать точку останова в программе. Точку останова можно также задать щелчком мыши на левой вертикальной рамке окна программы.

Так, задав точку останова на четвертой строке Макроса1 и запустив макрос, после остановки выполнения программы на этой строке в окне ***Watches*** можно просмотреть все свойства текущего документа (см. рисунок 5.49). Во время останова можно просмотреть значения свойств и пере-

менных во всплывающих подсказках, появляющихся после наведения курсора мыши на имя свойства или переменной. Далее программу можно выполнить по одной строке и следить за ее работой. Текст Макроса1 был несколько изменен: добавлено описание константы S и задан вывод значений командой ***Debug.Print*** (вывод идет в окно ***Immediate***).

Синтаксические ошибки в тексте программы автоматически выделяются красным цветом с показом окна сообщения об ошибке. Если в меню окна Visual Basic в пункте *Tools-Options* убрать галочку у строки *Auto Syntax Check*, выделение ошибок красным цветом останется, но окно сообщений об ошибке появляться не будет.

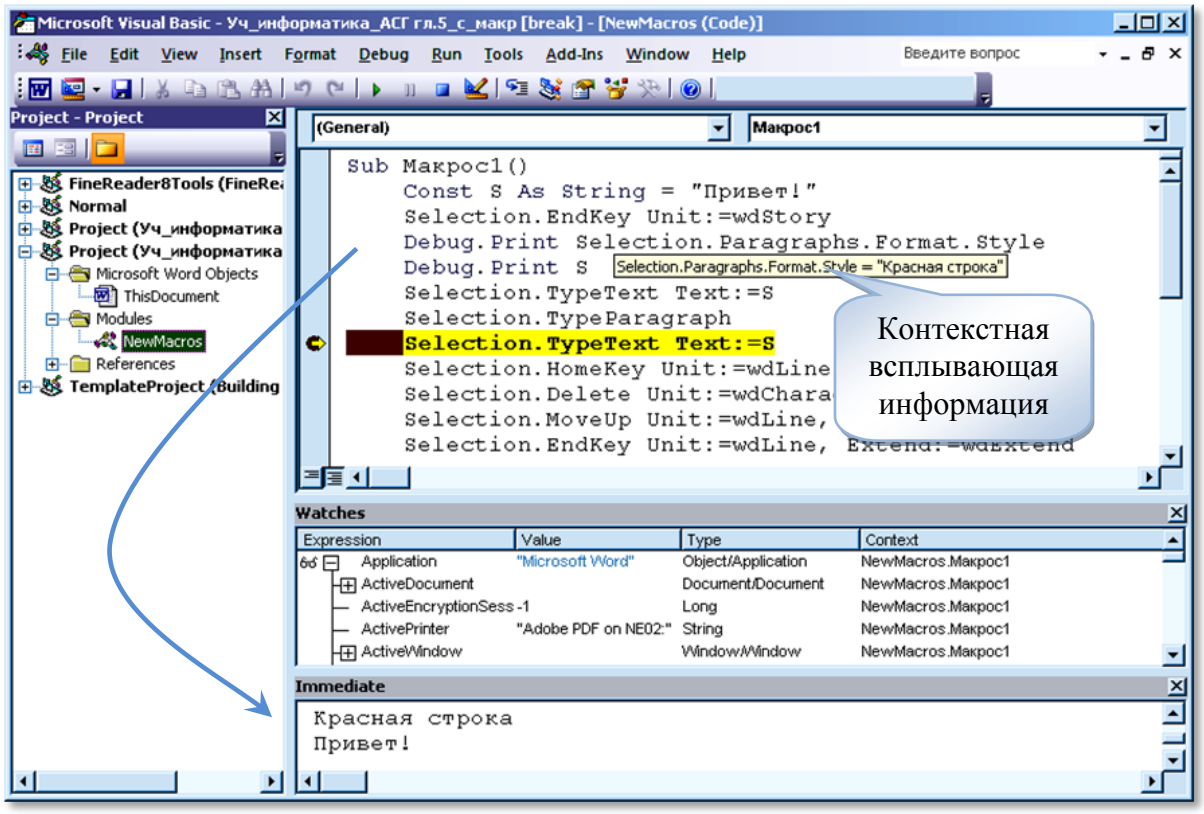


Рисунок 5.47. Отладочные режимы в окне Microsoft Visual Basic

На странице документа могут быть размещены стандартные объекты Word и объекты ActiveX, присутствующие на вкладке **Разработчик** – **Элементы управления** (показаны на рисунке 5.48).

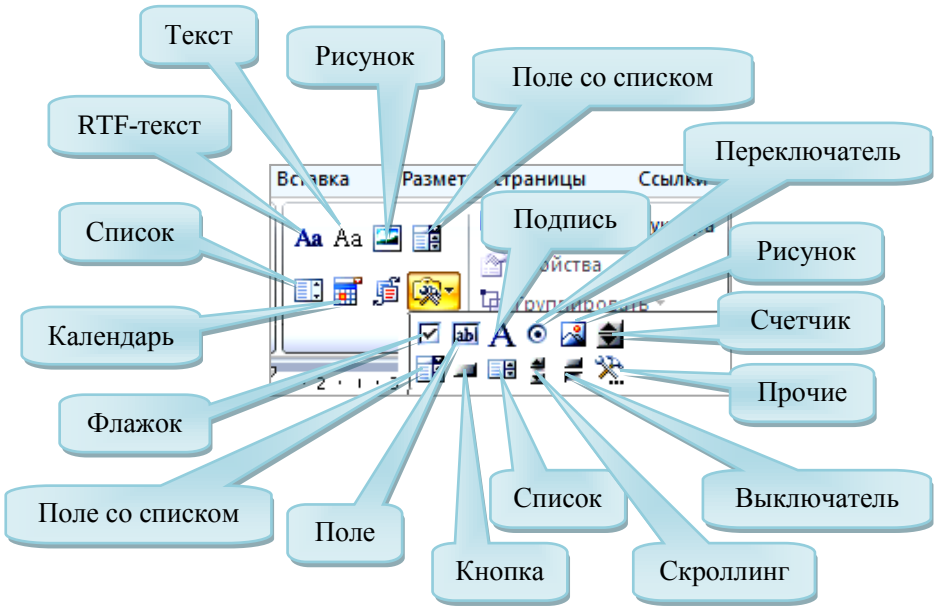


Рисунок 5.48. Элементы управления для документа Word

Простой пример организации вычислений на странице документа – три объекта типа **Поле:**

11,1235 + 12,3456 = 23,4691

Двойной щелчок на поле в **Режиме конструктора** (в котором мы оказываемся, когда выбираем операцию *Добавить поле* из **Элементов управления**) перемещает в редактор Visual Basic, в нем создается пустая процедура ***Private Sub TextBox1_Change()***, аналогично для второго поля. Напишем в этих процедурах имя процедуры ***Calc1*** и создадим процедуру расчета. Текст процедур будет выглядеть следующим образом:

```
Private Sub TextBox1_Change()
    Calc1
End Sub
```

```
Private Sub TextBox2_Change()
    Calc1
End Sub
```

```
Private Sub Calc1()
      On Error Resume Next
   TextBox3 = CSng(TextBox1) + CSng(TextBox2)
```

```
    If Err <> 0 Then
     MsgBox "Задайте число в формате, " & _
       "определенном в установках Windows", _
     vbCritical, "Ошибка ввода"
    End If
End Sub
```

Для третьего поля в окне **Properties** зададим свойство Enabled равным False, как показано на рисунке 5.49.

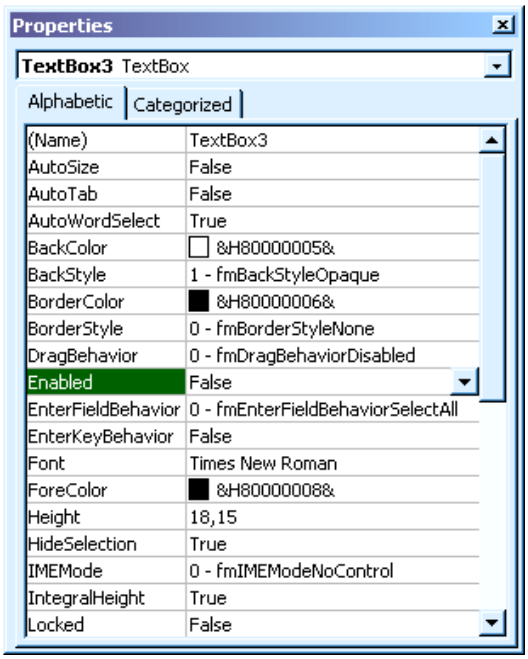


Рисунок 5.49. Окно свойств объекта

В результате после написания или изменения чисел в объектах ***TextBox1*** и ***TextBox2*** в поле ***TextBox3*** будет отображаться математическая сумма двух чисел.

Если в процедуре ***Calc1*** написать расчет без преобразования данных в числовое значение (***CSng(TextBox1)*** или ***CSng(TextBox1.Value)***), т. е. ***TextBox3 = TextBox1 + TextBox2,*** то вместо математической суммы получится сложение двух строковых значений.

Пример с использованием объекта **Кнопка** – расчет количества таблиц в текущем документе – выглядит следующим образом:

С объектом ***CommandButton1*** связана следующая процедура:

```
Private Sub CommandButton1_Click()
   TextBox5 = Tables.Count
End Sub
```

Далее показаны прочие свойства текущего документа, вписанные в данную страницу процедурой Prop_doc():

```
Приложение: Microsoft Word
Имя файла: Уч_информатика_АСГ_2009.docm
путь к файлу: D:\ASG\Мой учебник по информатике
Количество абзацев: 9980
Количество слов: 137200
Количество символов: 720834
Количество таблиц: 197
Количество объектов: 278
```

```
Количество полей: 963
Количество стилей: 367
```

Текст процедуры:

```
Sub Prop_doc()
  With ActiveDocument
   .Range(Start:=Selection.End, _
             End:=Selection.End).Style = "Программа"
    S = S & "Приложение: " &.Application & vbLf
    S = S & "Имя файла: " &.Name & vbLf
    S = S & "путь к файлу: " &.Path & vbLf
    S = S & "Количество абзацев: " _
          &.Paragraphs.Count & vbLf
    S = S & "Количество слов: " &.Words.Count & vbLf
    S = S & "Количество символов: " _
          &.Characters.Count & vbLf
    S = S & "Количество таблиц: " &.Tables.Count & vbLf
    S = S & "Количество объектов: " _
          &.Shapes.Count & vbLf
    S = S & "Количество полей: " &.Fields.Count & vbLf
    S = S & "Количество стилей: " _
          &.Styles.Count & vbLf
    Selection.TypeText (S)
  End With
End Sub
```

Пример более сложной программы в Word, рассчитывающей значения функции Y = F(X) для X, изменяющегося от $X_{нач.}$ до $X_{кои.}$ и для заданного числа точек, показывающей таблицу и диаграмму, приведен далее (текст скрипта для отдельного документа Word, в котором таблица 1 – исходные данные графика, таблица 2 – результаты расчета точек графика):

```
Private Sub CommandButton1_Click()
    Add_Table
    Tables(2).Select
    N = Shapes.Count
    MsgBox N
    chart1 = WordBasic.InsertChart
End Sub
```

```
Private Sub N_Change()
    Calc_h
End Sub
```

```
Private Sub XK_Change()
    Calc_h
```

```
End Sub
```

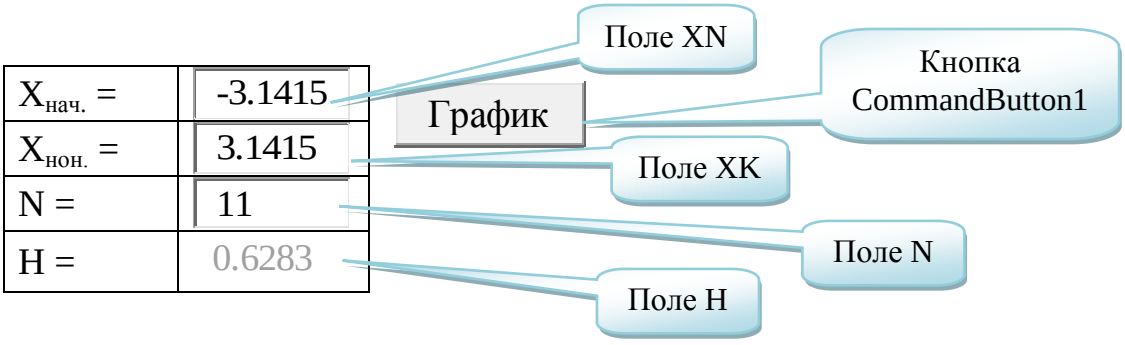
```
Private Sub XN_Change()
    Calc_h
End Sub
```

```
Sub Calc_h()
  With ActiveDocument
    x1 = Val(.XN)
    x2 = Val(.XK)
    n1 = Val(.N)
    If n1 <> 1 Then
      dx = (x2 - x1) / (n1 - 1)
    End If
   .H = dx
  End With
End Sub
```

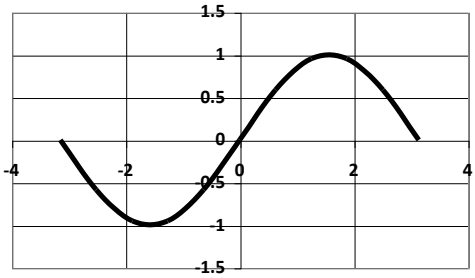
```
Sub Add_Table()
Dim X As Currency, Y As Currency
Selection.EndKey Unit:=wdStory
 With ActiveDocument
  If.Tables.Count = 2 Then
    .Tables(2).Delete
  Else
      Selection.TypeText vbNewLine
  End If
  Dim Table2 As Table
  Set Rng1 = Selection.Range
  Set Table2 =.Tables.Add(Rng1, n1+1, 2)
   ' добавлена таблица в заданном месте Range
   ' с количеством строк n1+1 и двумя столбцами
  Table2.Columns.Width = 50
   ' задана ширина колонок в таблице
  Table2.Borders.OutsideLineStyle = wdLineStyleSingle
  Table2.Borders.InsideLineStyle = wdLineStyleSingle
   ' заданы линии для внешних и внутренних границ
  Table2.Cell(1, 1).Range.Text = "X"
   ' задан текст для ячейки таблицы (1 ряд 1 столбец)
  Table2.Cell(1, 2).Range.Text = "Y"
  Table2.Cell(2, 1).Range.Text = Format(x1, "##0.00")
  Table2.Cell(2, 2).Range.Text = Format(Sin(x1), _
        "#0.000")
```

```
    X = x1
    For i = 3 To n1 + 1
       X = X + dx
       Table2.Cell(i, 1).Range.Text = Format(X, "##0.00")
       Y = Sin(X)
       Table2.Cell(i, 2).Range.Text = Format(Y, "#0.000")
    Next i
    Table2.Select
    Selection.Paragraphs.Alignment = wdAlignParagraphRight
     ' задано выравнивание по правому краю в абзацах таблицы
    Table2.Cell(1, 1).Range.ParagraphFormat.Alignment = _
          wdAlignRowCenter
     ' задано выравнивание в ячейке 1 ряд 1 столбец таблицы
    Table2.Cell(1, 2).Range.ParagraphFormat.Alignment = _
          wdAlignRowCenter
  End With
End Sub
```

Внешний вид таблиц и графика, с которыми связаны приведенные выше скрипты кнопки *CommandButton1* следующий:



X	Y
-3,14	0,000
-2,51	-0,588
-1,88	-0,951
-1,26	-0,951
-0,63	-0,588
0,00	0,000
0,63	0,588
1,26	0,951
1,88	0,951
2,51	0,588
3,14	0,000



5.3.2.2 Программирование на языке VBA в Microsoft Office Excel

Главный объект системы – **Excel.Application**, он имеет множество свойств и методов, отличных от **Word.Application**, некоторые из них показаны на рисунке 5.50.

Рисунок 5.50. Некоторые свойства и методы объекта Excel.Application (справка VBA Excel)

Объект Application имеет важнейшие свойства:

- **Cells** – ячейка активного листа,
- **Range** – одна ячейка или группа ячеек активного листа,
- **Columns** – вертикальная колонка ячеек активного листа,
- **Rows** – горизонтальный ряд ячеек активного листа,

- **Sheets** – свойство, которое возвращает ссылку на коллекцию, состоящую из объектов **Sheet** – множество листов книги со всеми их внедренными объектами (например, диаграммами);
- **Worksheets** – свойство, которое возвращает ссылку на коллекцию, состоящую из объектов **Worksheet** – листов книги без внедренных объектов,
- **Selection** – ссылка на активный выделенный объект (лист, диапазон ячеек, диаграмму (***Chart***), ряд, столбец и пр.

При написании программы в редакторе Microsoft Visual Basic системы Access для проекта открытого файла (который в этом случае должен сохраняться, как «Книга Excel с поддержкой макросов» с расширением *.xlsm) при использовании методов и свойств можно опускать название объекта **Application**. Методы программирования с использованием этих объектов рассмотрены далее на примерах.

Так же, как и в системе Microsoft Word, основы программирования в VBA для системы Excel можно освоить, воспользовавшись методом записи последовательности своих действий – **Запись макроса** на вкладке **Разработчик**, причем записываются не только нажатия клавиш, но и операции, выполняемые мышкой. Текст макроса позволяет лучше понять те операции, которые начинающий пользователь выполняет иногда не вполне осознанно, например, щелчок мышкой на ячейке A2 – команда Range("A2").Select и т. п. Конечно же, предполагается, что пользователь знает некоторые английские слова (например, Range – диапазон, Select – выбрать, Selection – выбор, Font – шрифт, Size – размер, Sheet – лист, Cell – ячейка, Characters – символы, Active – активный, Formula – формула, Center – центр, Border – граница, Edge – грань, Alignment – выравнивание …) и основы работы с объектами, тогда текст программы легко читается. Макрос будет содержать те приемы работы с объектами Excel, которые рекомендует к использованию разработчик данной системы, хотя возможны и другие варианты.

Выполним расчет значений функции и построение диаграммы Y = F(X), аналогичный приведенному ранее для Word в разделе 5.3.2.1.

Вид листа Excel, который мы получим в результате всей работы, заранее показан на рисунке 5.51, чтобы далее было легче понимать текст VBA-программ этого примера.

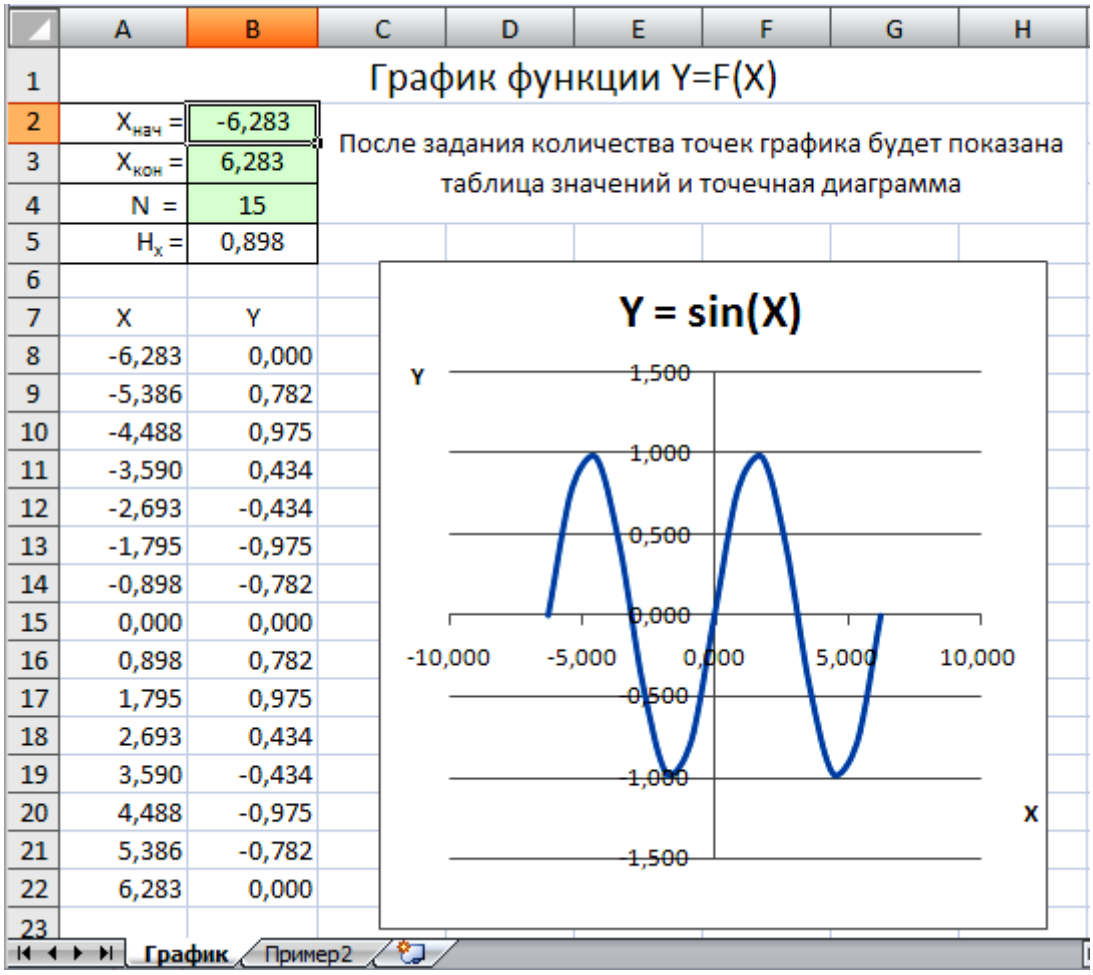


Рисунок 5.51. Построение графика функции

Запишем макрос для написания программой названия «График функции Y=F(X)» в группе объединенных ячеек A1: H1 и таблицы исходных данных в ячейках A2:B5. Текст макроса (после значительного его сокращения – убраны некоторые значения, принимаемые по умолчанию, добавлены комментарии) будет следующий:

```
Sub Табл_исх_данных()
 ' ---- Таблица исходных данных ----
    Range("A1:H1").Select
    Selection.Merge ' объединить ячейки
    Selection.HorizontalAlignment = xlCenter
    ActiveCell.FormulaR1C1 = "График функции Y=F(X)"
    Selection.Font.Size = 16
    Range("A2").Select
    ActiveCell.FormulaR1C1 = "Хнач = "
    ActiveCell.Characters(Start:=2, Length:=3) _
    .Font.Subscript = True ' подстрочные символы
    Range("A3").Select
```

```
    ActiveCell.Formula = "Xкон = "
    ActiveCell.Characters(Start:=2, Length:=3) _
     .Font.Subscript = True
    Range("A4").Formula = "N  = "
    Range("A5").Select
    Selection.Formula = "Hx = "
    ActiveCell.Characters(Start:=2, Length:=1) _
      .Font.Subscript = True
    Range("A2:A5").Select
    Selection.HorizontalAlignment = xlRight
  ' ----  Исходные данные ----
    Range("B2").Formula = "= -3*PI()"
    Range("B3").Formula = "=  3*PI()"
    Range("B2:B3").Select
    Selection.NumberFormat = "0.000"
    Range("B4").Formula = "= 15"
    Range("B5").Formula = "=(B3-B2)/(B4-1)"
    Range("B5").NumberFormat = "0.000"
    Range("B2:B5").Select
    Selection.HorizontalAlignment = xlCenter
    Range("A2:B5").Borders(xlEdgeLeft).LineStyle = _
        xlContinuous    ' линия по левой границе
     Range("A2:B5").Borders(xlEdgeTop).LineStyle = _
        xlContinuous
     Range("A2:B5").Borders(xlEdgeBottom).LineStyle = _
        xlContinuous
     Range("A2:B5").Borders(xlEdgeRight).LineStyle = _
        xlContinuous
     Range("A2:B5").Borders(xlInsideVertical). _
        LineStyle = xlContinuous
     Range("A2:B5").Borders(xlInsideHorizontal). _
        LineStyle = xlContinuous
    Range("B2:B4").Select
    Selection.Interior.Color = 14220508
    Selection.Locked = False    'снята зашита с ячеек
    Selection.FormulaHidden = False
End Sub
```

Более сложная задача – создание таблицы значений Y=F(x) и точечной диаграммы для исходных данных, описанных в первой части. При обычном оформлении такого расчета можно изменять $X_{нач}$ и $X_{кон}$, данные таблицы и диаграмма будут соответственно пересчитываться. Если же изменить N

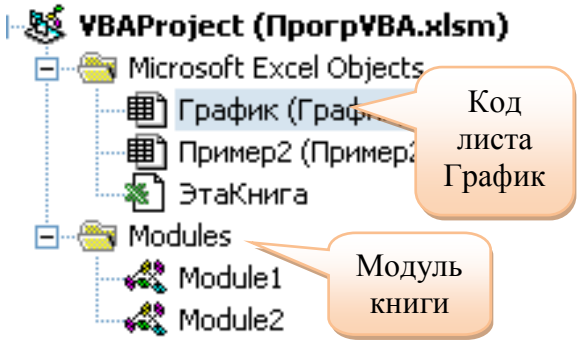


Рисунок 5.52. Проект VBA

– количество расчетных точек, необходимо изменять количество строк с формулами в таблице и диапазон данных для диаграммы. Вариант решения этой задачи программным путем показан далее. Текст программ должен располагаться в разделе кода для листа с именем График (текст предыдущей процедуры ***Табл_исх_данных*** мог располагаться здесь же или в программном модуле книги, см. рисунок 5.52):

```
Dim NLTG As Single      ' в разделе General Declarations
                        ' кода листа График
```

```
Private Sub Worksheet_SelectionChange _
            (ByVal Target As Range)
    ' процедура для листа, автоматически выполняется
    ' при выборе нового объекта на листе
     NLTG = Cells(4, 2)       ' запоминание старого
                              ' количества строк в таблице
End Sub
```

```
Private Sub Worksheet_Calculate()
    ' процедура для листа, автоматически выполняется
    ' при выполнении расчета по формулам на листе
If Cells(4, 2) <> NLTG Then
    NLT = NLTG  ' запоминаем старое значение
    NLTG = Cells(4, 2) ' запоминаем новое значение
     ' если этого не сделать, процедура зациклится
     ' т. к. она имеет рекурсивный характер
    ActiveSheet.Unprotect
    Range(Cells(7, 1), Cells(7 + NLT, 4)).Delete
     ' удаляем старую таблицу, в которой NLT строк
    If Shapes.Count > 0 Then ' удаляем старый график
       ActiveSheet.ChartObjects("Grafik").Activate
       ActiveChart.Parent.Delete
    End If
    Расчет   'выполняем процедуру Расчет
    График_функции   'выполняем процедуру
    ActiveSheet.Protect DrawingObjects:=True, _
              Contents:=True, Scenarios:=True
    ActiveSheet.EnableSelection = xlUnlockedCells
         'устанавливаем защиту на лист кроме ячеек B2:B4
 End If
End Sub
```

```
Sub Расчет()
```

```
' расчет таблицы Y=F(x)
   Range("A7").Formula = "X"
   Range("B7").Formula = "Y"
   Range("A7:B7").HorizontalAlignment = xlCenter
   Range("A8").Formula = "= B2"
   For i = 2 To Range("B4").Value
      Cells(i + 7, 1).Value = "=R[-1]C+R5C2"
   Next
   For i = 1 To Range("B4").Value
    Cells(i + 7, 2).Value = "=SIN(RC[-1])"
   Next
   Range(Cells(8, 2), Cells(8 + _
      Range("B4").Value, 2)).NumberFormat = "0.000"
End Sub
```

```
Sub График_функции()
' построение диаграммы
  ActiveSheet.Shapes.AddChart.Select
  ActiveChart.Parent.Name = "Grafik"
  ActiveChart.Parent.Height = 250
  ActiveChart.Parent.Width = 250
  ActiveChart.Parent.Left = 120
  ActiveChart.Parent.Top = 80
  rng = "A7:B" + LTrim(Str(7 + Range("B4").Value))
  ActiveChart.SetSourceData Source:=Range(rng)
      'задаем диапазон исходных данных("A7:B22")
  ActiveChart.ChartType = xlXYScatterSmoothNoMarkers
  ActiveChart.PlotBy = xlRows
  ActiveChart.PlotBy = xlColumns
  ActiveChart.ChartTitle.Text = "Y = sin(X)"
  ActiveChart.Axes(xlValue).HasTitle = True
  ActiveChart.Axes(xlValue).AxisTitle.Select
  ActiveChart.Axes(xlValue, xlPrimary). _
        AxisTitle.Text = "X"
  ActiveChart.SetElement _
       (msoElementPrimaryValueAxisTitleHorizontal)
  ActiveChart.Axes(xlCategory).HasTitle = True
  ActiveChart.Axes(xlCategory).AxisTitle.Select
  ActiveChart.Axes(xlCategory, xlPrimary). _
       AxisTitle.Text = "Y"
  ActiveChart.Axes(xlCategory).AxisTitle.Select
  Selection.Left = 5
  Selection.Top = 31.333
```

```
    ActiveChart.Axes(xlValue).AxisTitle.Select
    Selection.Left = 325.011
    Selection.Top = 194.794
    ActiveChart.Legend.Select
    Selection.Delete
End Sub
```

Процедуры обработки событий для листа: ***Private Sub Worksheet_SelectionChange (ByVal Target As Range)*** и ***Private Sub Worksheet_Calculate()*** не могут быть сгенерированы с помощью **Записи макросов**, их должен писать программист. Основа процедур ***Sub Расчет()*** и ***Sub График_функции()*** была создана с помощью режима **Запись макроса**, затем они были модифицированы: организована возможность их выполнения для различного количества точек графика, добавлено название ***ActiveChart.Parent.Name = "Grafik"*** для реализации возможности его удаления в процедуре ***Worksheet_Calculate()***, добавлены координаты и размер графика, заданы координаты для надписей осей графика, и т. д.

Важная задача, которую можно решить в Excel – подключение к базе данных и отображение одной из ее таблиц на рабочем листе. На вкладке ленты *Данные* кнопка *Получить внешние данные* (см. рисунок 5.55) позволяет выполнить импорт данных из текстовых файлов, баз данных системы Access, серверов баз данных Microsoft SQL Server, Oracle и других, выполнить отбор данных из этих источников с использованием запросов Microsoft Query. При этом используются OLE DB провайдеры (библиотеки **COM** интерфейсов), драйверы доступа к данным, информация о которых хранится в средстве Windows ODBC (Open Database Connectivity), прикладные задачи работы с информацией базы данных могут решаться с использованием библиотек *Microsoft ActiveX Data Objects (ADO)* и *Microsoft Data Access Objects (DAO)*.

При выполнении операции открытия таблицы реляционной базы данных с использованием пункта ленты Excel *Получить внешние данные* программный код сохраняется в профиле пользователя в папке *Мои источники данных* в виде файла с расширением *.odc. В последующем этим подключением можно воспользоваться повторно из пункта *Существующие подключения* (см. рисунок 5.53) и свойства его просмотреть в группе *Подключения*. Другой вариант использования программного кода подключения и извлечения информации из баз данных – записать макрос подключения и в дальнейшем использовать его, как основу для написания программы работы с этой информацией.

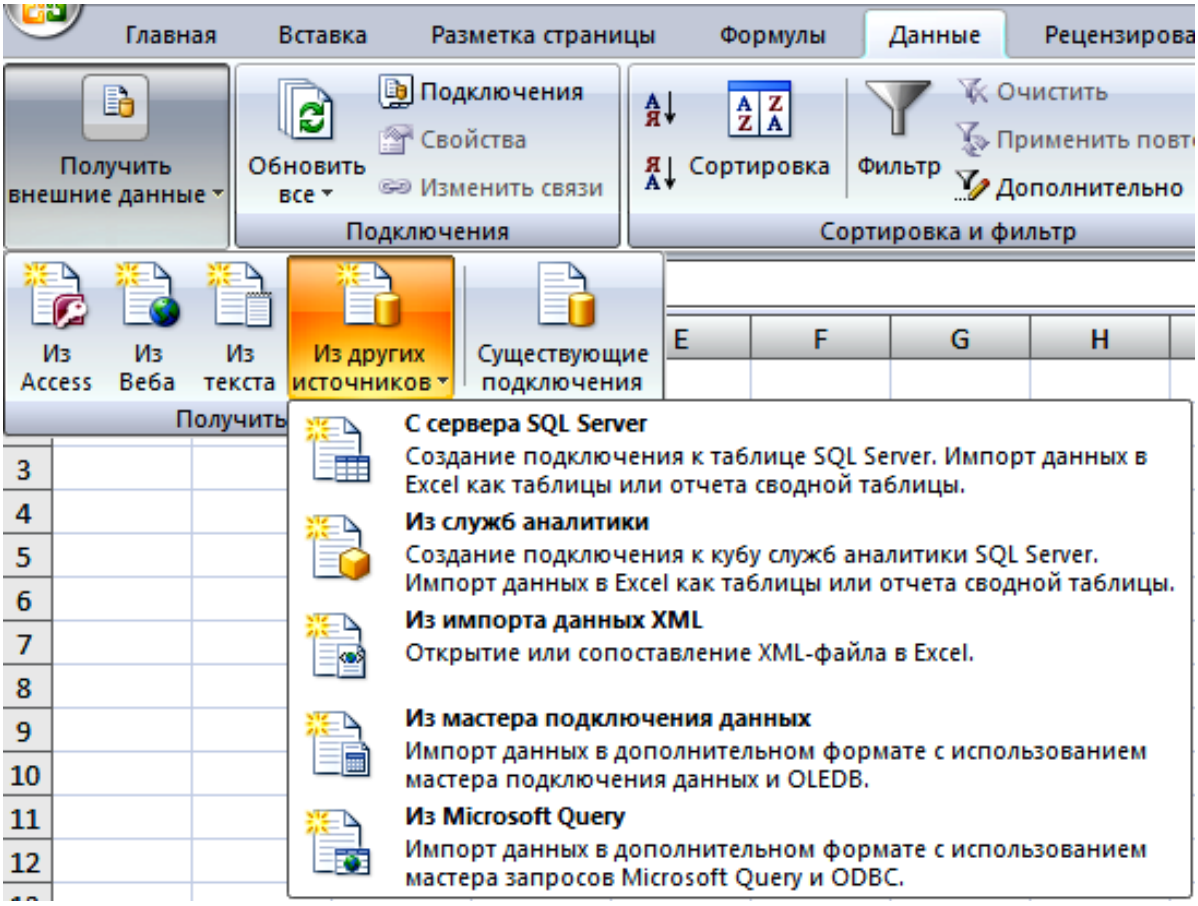


Рисунок 5.53. Средства работы с внешними данными в Excel

К сожалению, стандартные средства, используемые в системе Excel не всегда правильно отображают тип данных реляционной таблицы, возможны также проблемы при работе с текстовой информацией национальных кодовых таблиц.

Например, код загрузки данных из Access может выглядеть следующим образом (из текста убраны параметры, используемые по умолчанию):

```
Sub access_Список()
    'данные таблицы Список базы Контингент.accdb
    With ActiveSheet.ListObjects.Add(SourceType:=0, _
Source:=Array("OLEDB;Provider=Microsoft.ACE.OLEDB.12.0;" _
        & "Data Source=C:\AGTU\Контингент.accdb;" _
        & "Mode=Share Deny Write;" _
        & "Jet OLEDB:Engine Type=6;" _
        & "Jet OLEDB:Database Locking Mode=0;" _
        & "Jet OLEDB:Global Partial Bulk Ops=2;" _
        & "Jet OLEDB:Global Bulk Transactions=1;"), _
        Destination:=Range("$A$1")).QueryTable
```

```
        .CommandType = xlCmdTable
        .CommandText = Array("Список")
        .PreserveFormatting = True
        .BackgroundQuery = True
        .RefreshStyle = xlInsertDeleteCells
        .SaveData = True
        .AdjustColumnWidth = True
        .RefreshPeriod = 0
        .PreserveColumnInfo = True
        .SourceDataFile = "C:\AGTU\Контингент.accdb"
        .ListObject.DisplayName = "Табл_Контингент.accdb"
        .Refresh BackgroundQuery:=False
    End With
End Sub
```

Результат выполнения данной программы – таблица данных (***ListObject***) на рабочем листе (см. рисунок 5.54), работа с которой выполняется с использованием контекстной вкладки *Работа с таблицами.*

	A	B	C	D	E	F	G	H
1	NZ	FIO	DATA_P	N_FCLT	N_SPECT	KURS	N_GRUP	N_PASP
2	123976	Шварц Ирана Олеговна	25.07.2007	3	240406.65	1	572	34 3297
3	123977	Савин Сергей Олегович	16.07.2006	2	240406.65	1	572	369721
4	239648	Андреев Сергей Сергеевич	12.07.2005	2	240406.65	1	572	236971
5	239764	Проданова Нина Сергеевна	19.07.2005	2	240406.65	1	572	139764
6	269731	Сергеева Ирина Анатольевна	19.07.2005	2	240406.65	1	572	136975
7	459871	Антонова Ирина Сергеевна	17.08.2005	1	280201.65	1	571	116897
8	555337	Таратина Юлия Васильевна	01.08.2006	5	200501.65	1	573	115689
9	555354	Мосеева Ольга Юрьевна	14.08.2005	4	200501.65	1	573	116978
10	555671	Яхина Лилия Тахировна	17.08.2005	5	200501.65	1	573	112398
11	555978	Зиновьева Елена Сергеевна	18.08.2005	4	200501.65	1	573	112596

Рисунок 5.54. Результат переноса данных из базы Access в Excel

Синтаксис команды добавления таблицы на рабочий лист следующий:

```
ListObjects.Add(SourceType, Source, LinkSource, _
                HasHeaders, Destination)
```

где ***SourceType*** – тип данных, равен 0 для внешних данных;

Source – данные, представлены в данном примере массивом строк таблицы типа ***QueryTable***, которые определяются строкой установления связи с базой данных Access с использованием провайдера ***Microsoft Jet OLEDB.*** Параметры строки определяются используемым типом связи, их синтаксис и описание можно найти в справке языка

VBA системы Microsoft Access;

LinkSource и HasHeaders – по умолчанию имеют значение True;

Destination – объект Range, специфицирующий адрес ячейки левого верхнего угла таблицы.

Другой вариант программного кода получения данных таблицы *Список* базы данных *Контингент*, расположенной на Microsoft SQL Server'е:

```
Sub MSSQLServer_Контингент()
' Данные с сервера, база Контингент
str_connect = "Provider=SQLOLEDB.1;" _
    & "Integrated Security=SSPI;" _
    & "Auto Translate=True;" _
    & "Initial Catalog=Контингент;" _
    & "Data Source=.;" _
    & "Workstation ID=LKSN"
 Set myList = ActiveSheet.ListObjects.Add(SourceType:=0, _
     Source:=Array("OLEDB;" & str_connect & ";" _
    & "Tag with column collation when possible=False"), _
     Destination:=Range("$A$1")).QueryTable
 myList.CommandType = xlCmdTable
 myList.PreserveFormatting = True
 myList.ListObject.DisplayName ="База_Контингент_Список "
 myList.CommandText = _
               Array("""Контингент"".""dbo"".""Список""")
 myList.Refresh
End Sub
```

Результат работы будет выглядеть аналогично рисунку 5.54, однако для колонки с типом данных *Дата* необходимо будет задать этот тип в Excel'е вручную. Пример с использованием библиотеки доступа к данным ADO:

```
Sub ADO_MS_SQL_Server_Контингент()
Dim cnnConnect As ADODB.Connection
Set cnnConnect = New ADODB.Connection
cnnConnect.Open "Provider=SQLOLEDB;" & _
    "Data Source=(LOCAL);" & _
    "Integrated Security=SSPI;" & _
    "Workstation ID=LKSN;" & _
    "Initial Catalog=Контингент"
```

```
Set rstRecordset = New ADODB.Recordset
rstRecordset.Open _
    Source:="Select * From Список", _
    ActiveConnection:=cnnConnect, _
    CursorType:=adOpenDynamic, _
    LockType:=adLockReadOnly, _
    Options:=adCmdText
With ActiveSheet.QueryTables.Add( _
        Connection:=rstRecordset, _
        Destination:=Range("A1"))
   .Name = "Список"
   .FieldNames = True
   .RowNumbers = False
   .FillAdjacentFormulas = False
   .PreserveFormatting = True
   .RefreshOnFileOpen = False
   .BackgroundQuery = True
   .RefreshStyle = xlInsertDeleteCells
   .SavePassword = True
   .SaveData = True
   .AdjustColumnWidth = True
   .RefreshPeriod = 0
   .PreserveColumnInfo = True
   .Refresh BackgroundQuery:=False
End With
```

В этом случае в программе используется создание двух объектов библиотеки Microsoft ADO:

1) `Set cnnConnect = New ADODB.Connection`
   – строка связи с базой данных;
2) `Set rstRecordset = New ADODB.Recordset`
   – набор записей таблицы базы данных.

Библиотеку ADO нужно подключить в системе VBA в пункте меню **Tools – References…**, как показано на рисунке 5.55.

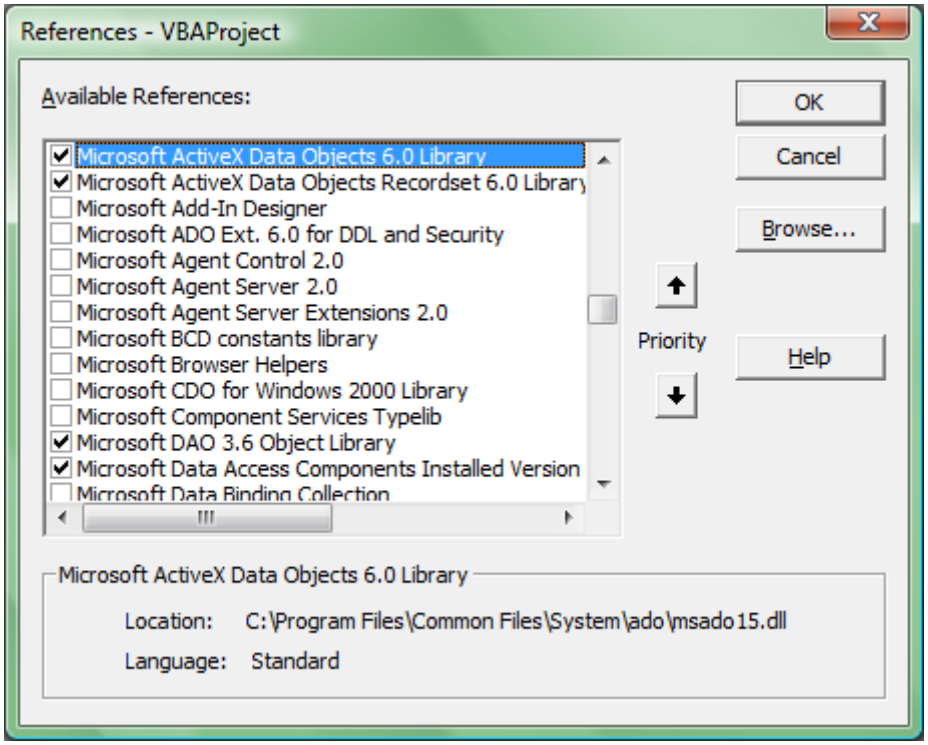


Рисунок 5.55. Подключение дополнительных библиотек **ADO** и **DAO** в системе VBA

Результат работы программы показан на рисунке 5.56. В этом случае данные представлены, как обычная информация в ячейках Excel с соблюдением формата *Дата-время*.

	A	B	C	D	E	F	G	H
1	NZ	FIO	DATA_P	N_FCLT	N_SPECT	KURS	N_GRUP	N_PASP
2	123976	Шварц Ирана Олеговна	25.07.2007 0:00	3	240406.65	1	572	34 3297
3	123977	Савин Сергей Олегович	16.07.2006 0:00	2	240406.65	1	572	369721
4	239648	Андреев Сергей Сергеевич	12.07.2005 0:00	2	240406.65	1	572	236971
5	239764	Проданова Нина Сергеевна	19.07.2005 0:00	2	240406.65	1	572	139764
6	269731	Сергеева Ирина Анатольевна	19.07.2005 0:00	2	240406.65	1	572	136975
7	459871	Антонова Ирина Сергеевна	17.08.2005 0:00	1	280201.65	1	571	116897
8	555337	Таратина Юлия Васильевна	01.08.2006 0:00	5	200501.65	1	573	115689
9	555354	Мосеева Ольга Юрьевна	14.08.2005 0:00	4	200501.65	1	573	116978
10	555671	Яхина Лилия Тахировна	17.08.2005 0:00	5	200501.65	1	573	112398
11	555978	Зиновьева Елена Сергеевна	18.08.2005 0:00	4	200501.65	1	573	112596

Рисунок 5.56. Данные Microsoft SQL Server'а с использованием ADO

Еще один пример – получение данных с сервера баз данных Oracle с использованием драйвера ODBC и запроса:

```
Sub Запрос_Ora_TTCCOM130203()
' Запрос_Ora_TTCCOM130203
```

```
With ActiveSheet.ListObjects.Add(SourceType:=0, _
  Source:=Array(Array("ODBC;DSN=Oracle;UID=BAAN;;" & _
  "DBQ=BAAN5 ;DBA=W;APA=T;EXC=T;" & _
  "XSM=Owner;FEN=T;QTO=T;FRC=10;FDL=10;LOB=T;" & _
  "RST=T;GDE=T;FRL=F;BAM=IfAllSuccessful;MTS=F;" & _
  "MDI=F;CSR=F;FWC=F;PFC=10;TLO=0;")), _
  Destination:=Range("$A$1")).QueryTable
  .CommandText = Array("SELECT TTCCOM130203.T$NAMA," & _
  "TTCCOM130203.T$NAMC, TTCCOM130203.T$NAME, " & _
  "TTCCOM130203.T$PSTC, TTCCOM130203.T$DTLM " & _
  Chr(13) & Chr(10) & "FROM BAANDB.TTCCOM130203" & _
  TTCCOM130203 & Chr(13) & "" & Chr(10) & _
  "WHERE (TTCCOM130203.T$DTLM > " & _
  "{ts '2005-04-01 23:59:59'})" & _
  "AND (Not TTCCOM130203.T$PSTC=' ')")
  .FillAdjacentFormulas = False
  .PreserveFormatting = True
  .BackgroundQuery = True
  .RefreshStyle = xlInsertDeleteCells
  .SaveData = True
  .AdjustColumnWidth = True
  .RefreshPeriod = 0
  .PreserveColumnInfo = True
  .ListObject.DisplayName = "Query_from_Oracle"
  .Refresh BackgroundQuery:=False
 End With
End Sub
```

Результаты выполнения запроса к базе данных Baan5 на сервере Oracle показаны на рисунке 5.57.

	A	B	C	D	E
1	T$NAMA	T$NAMC	T$NAME	T$PSTC	T$DTLM
2	Производственное предприятие	Ленинградскй пр.	Архангельск	163000	11.04.2005 10:36
3	Клиент - покупатель	ул. Металлистов, д.1	Северодвинск	164000	11.04.2005 6:10
4	Производственное предприятие № 100	ул. Смольный Буян	Архангельск	163000	05.04.2005 12:54
5	Склад 810 предприятия № 100	ул. Смольный Буян	Архангельск	163000	05.04.2005 9:17
6	Клиет	ул. Советская	Пенза	173000	05.04.2005 6:35
7	Пставщик 2002	ул. Советская	Уфа	183000	05.04.2005 12:59
8	Иванов Иван Иванович	ул. Северодвинская	Архангельск	163000	05.04.2005 6:40

Рисунок 5.57. Данные с сервера Oracle

5.3.2.3 Программирование на языке VBA в Microsoft Office Access

В системе Access также существует объект **Access.Application**, как и в двух предыдущих. С использованием его можно запустить в работу Access, его свойства и методы позволяют настроить параметры пользовательского интерфейса, получить доступ к некоторым групповым операциям с таблицами, к компонентам форм и отчетов, к программным модулям и пр.

При разработке клиентских приложений для работы с базой данных в виде экранных форм системы Access могут использоваться методы и свойства этого объекта и всех вложенных в него объектов (прежде всего полей) с использованием библиотеки *Microsoft Access Object Library.*

При написании программ на языке VBA используется библиотека *Visual Basic For Application.*

Для выполнения сложных расчетных операций с данными таблиц базы данных в системе Access необходимо использовать библиотеки **ADO** (*Microsoft ActiveX Data Objects)* или **DAO** (*Microsoft Data Access Objects)*, о которых упоминалось ранее, как о библиотеках, обеспечивающих подключение к базам данных и выполнение операций с их данными из самых разных программных систем (см. раздел *5.3.1.12 Работа с объектами*). Библиотека **ADO** более современная, чем **DAO**, считается, что она содержит более развитые возможности работы с данными. Основы работы в этих двух системах очень близки, главным объектом в них на уровне базы данных является ***Recordset*** (набор записей).

Recordset – временная таблица (*cursor*), созданная из записей таблицы базы данных или в результате выполнения SQL-запроса

Синтаксис SQL-запросов рассматривался ранее в разделе *3.3.4.6 Использование запросов*.

Просмотреть список всех используемых библитотек и подключить при необходимости другие библиотеки можно в окне *References* окна *Microsoft Visual Basic* системы Access (см. ранее рисунок 5.55).

Для объекта ***Recordset*** как в **ADO**, так и в **DAO** определено большое количество свойств и коллекций, методов и событий, краткое описание их для **ADO** приведено в **Приложении 11.** Например, определить, находимся ли мы в конце ***Recordset***, можно с использованием метода ***EOF*** этого объекта, узнать количество записей позволяет метод ***RecordCount***, добавить новую запись в ***Recordset*** позволяет метод ***AddNew***, переместиться на следующую запись – ***MoveNext***, создать временный набор записей (т. е. сам объект ***Recordset***) – метод ***Open*** и т. д.

При создании объекта ***Recordset*** могут быть определены динамические или статические его типы (см. таблицу 5.7).

Таблица 5.7. Типы объекта ***Recordset*** в библиотеках **ADO** и **DAO**

ADO		**DAO**	
Тип	Краткое описание	Тип	Краткое описание
Dynamic cursor **(adOpen-Dynamic)**	Позволяет видеть добавление, изменение, удаление записей, выполняемые другими пользователями. Метод <**Recordset**>.**Update** позволяет обновить записи в таблице базы данных.	**Dynamic-type**	Результат выполнения запроса для одной или нескольких таблиц базы данных, в котором можно добавлять, изменять или удалять записи. Метод <**Recordset**>.**Update** позволяет обновить запись в таблице базы данных, если до редактирования записи был выполнен метод **Recordset>.Edit**.
Keyset cursor **(adOpen-Keyset)**	Позволяет видеть изменение записей, не позволяет видеть добавленные записи, не позволяет редактировать удаленные записи, при выполнении этих операций другими пользователями.	**Dynaset-type**	Результат выполнения запроса, который может обновлять записи в таблицах базы данных. Можно добавлять, изменять, или удалять записи одной таблицы или нескольких таблиц.
Static cursor **(adOpen-Static)**	Статическая копию ряда записей, которую можно использовать, чтобы найти данные или генерировать отчеты.	**Snapshot-type**	Статическая копия ряда записей, которую можно использовать, чтобы найти данные или генерировать отчеты
Forward-only cursor **(adOpen-Forward-Only)**	Позволяет перемещаться только вперед в объекте **Recordset**. Добавления, изменения или удаления записей другими пользователями не видимы. Используется, когда необходимо однократное прохождение по записям объекта **Recordset**.	**Forward-only-type**	Идентичен типу **Snapshot** за исключением того, что курсор не создается. Можно только выполнить перемещение вперед через записи. Можно использовать, когда необходимо сделать единственный проход через записи.
		Table-type	Позволяет добавлять, изменять или удалять записи в одной из таблиц базы данных.

Для создания объекта ***Recordset*** в процедурах базы данных Access используется следующий синтаксис:

recordset*.Open *<источник>*, *<соединение>*, *<тип курсора>*, *<тип блокировки>*, *<опции>

где ***источник*** – SQL-запрос или объект *Command* ;

соединение – информация о связи с данными (*ConnectionString)* ;

тип курсора – см. таблицу 5.7;

тип блокировки – оптимистическая (**adLockOptimistic**, запись блокируется только в момент сохранения), пессимистическая (**adLockPessimistic**, запись блокируется в начале ее редактирования) и пр.;

опции – используется, если ***источник*** – *Command*.

Некоторые основы работы с объектами библиотеки Access и использования возможностей библиотек **ADO** и **DAO** рассмотрены далее на примере расчета сдельного наряда (данные, приведенные в примере, взяты из системы «Зарплата» функционировавшей на одном из предприятий, разработчиком которой в системе FoxPro был автор данного учебника).

Схема фрагмента базы данных, связанная с данной задачей, показана на рисунке 5.58.

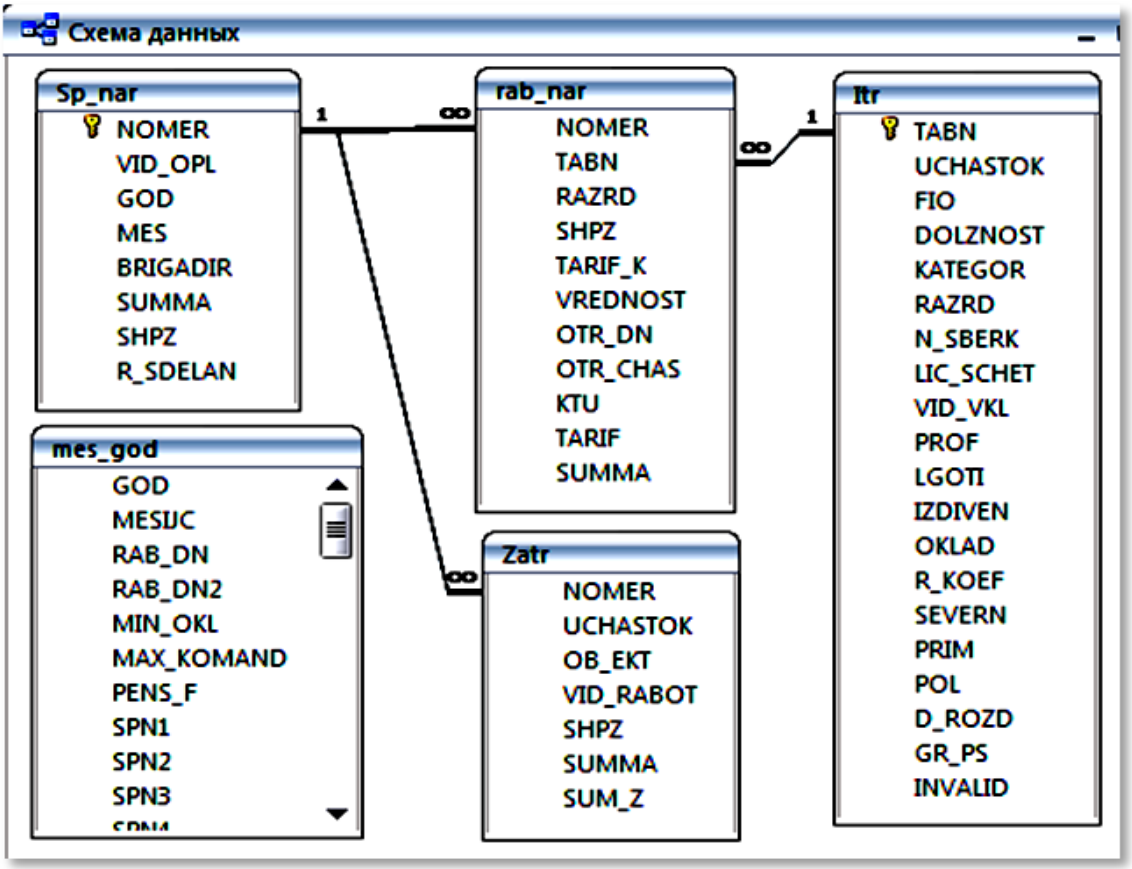


Рисунок 5.58. Схема базы данных сдельного наряда

Внешний вид экранной формы для работы со сдельными нарядами показан на рисунке 5.59.

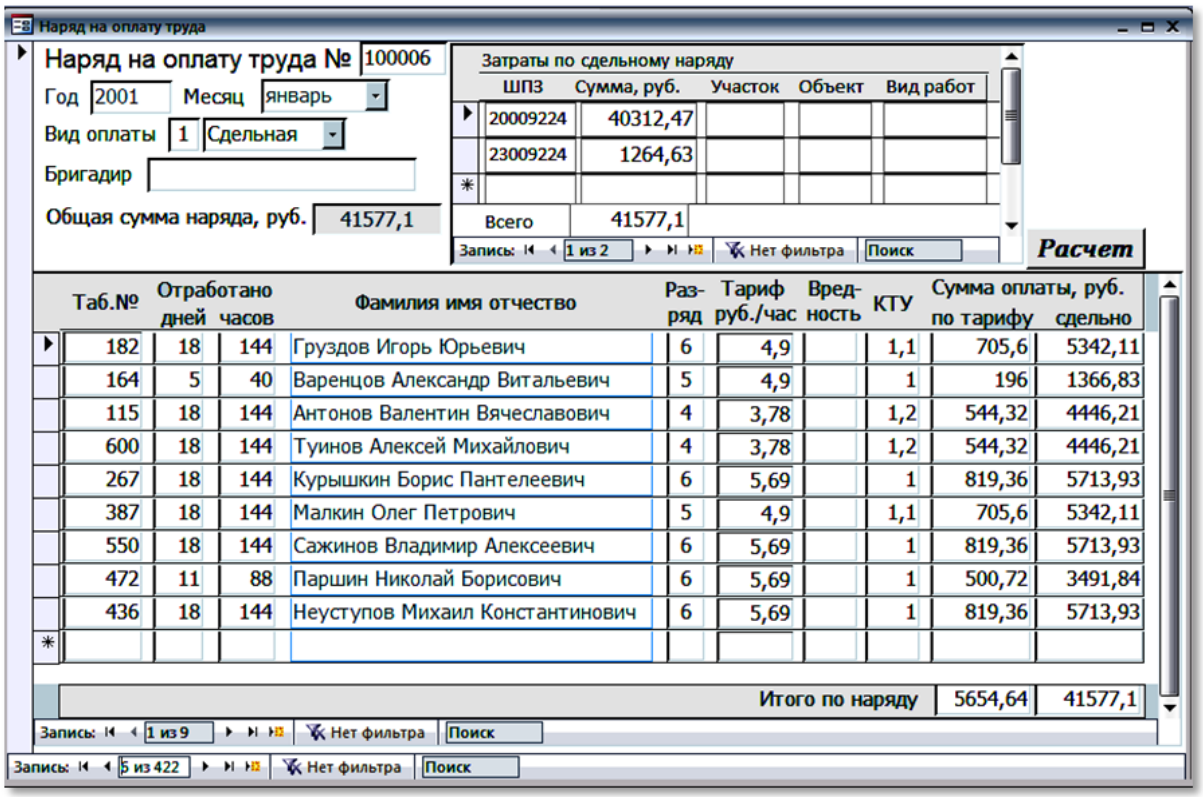


Рисунок 5.59. Сдельный наряд в Microsoft Office Access

На данной форме присутствуют данные четырех таблиц базы ZRPL:

- SP_NAR – список нарядов и общие данные наряда;
- RAB_NAR – данные работников в наряде;
- ZATR – суммы затрат по различным ШПЗ в наряде;
- ITR – справочник работников.

С объектами формы связано достаточно много процедур. Для главной формы заданы процедуры обработки события Load (выполняется при загрузке формы в память) и Current (выполняется при перемещении по записям и при обновлении формы), а также в разделе General-Declaration описана глобальная переменная NN (см. рисунок 5.60).

В процедуре *Form_Load* формы *Список нарядов* одна строка:

`DoCmd.GoToRecord,, acLast` – перейти на последнюю запись данных формы.

В процедуре *Form_Current* две строки:

Form_refr – выполнение процедуры для задания нередактируемых полей год и месяц, если они ранее были определены для наряда (редактирование возможно только для новой записи).

Find_TK – выполнение процедуры для определения тарифных коэффициентов текущего месяца; процедура Find_TK описана в том же окне кода формы Список нарядов.

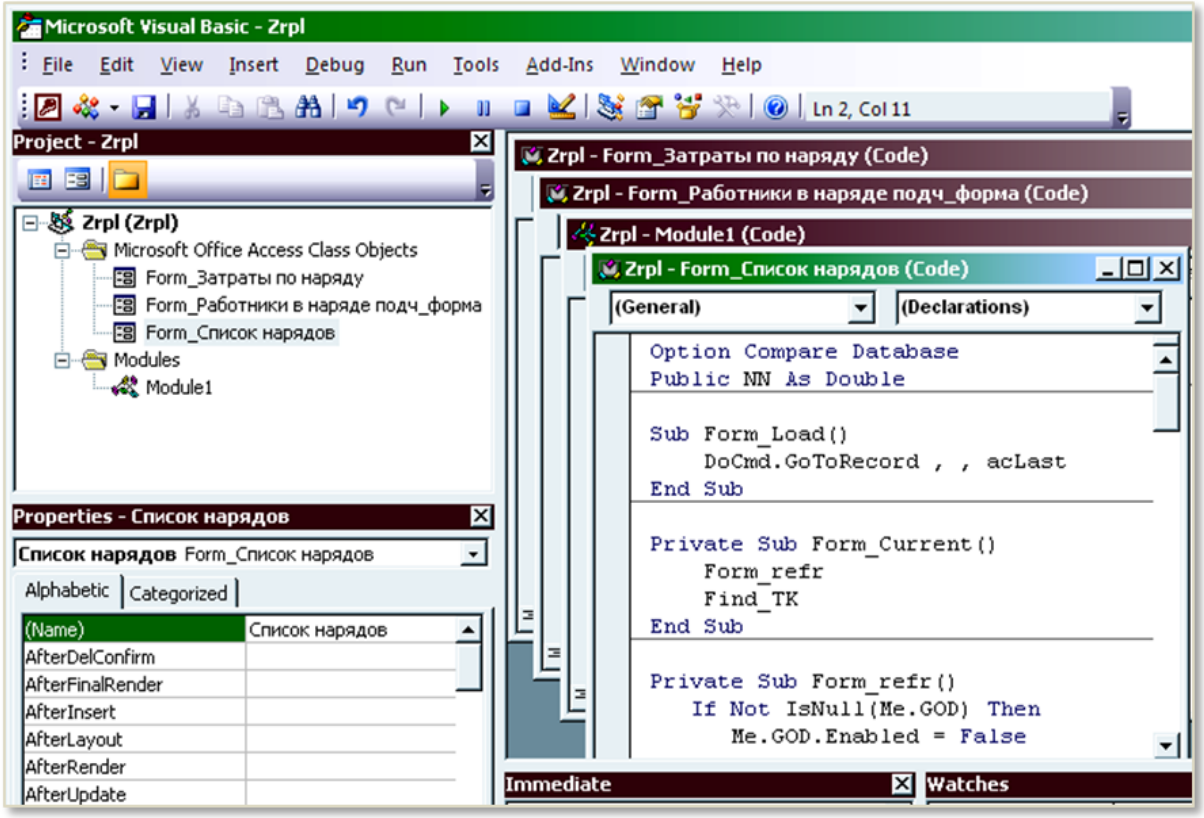


Рисунок 5.60. Процедуры событий Load и Current формы Список нарядов базы данных Zrpl в окне редактора Visual Basic

Текст этих двух процедур:

```
Private Sub Form_refr()
If Not IsNull(Me.GOD) Then          ' Me.GOD – поле с именем GOD
    Me.GOD.Enabled = False          ' текущей формы
Else                                ' к значению поля текущей
    Me.GOD.Enabled = True           ' формы можно обращаться
End If                              ' и без Me, а только по имени
If Not IsNull(Me.MES) Then
   Me.ПолеСоСписком21.Enabled = False
Else
    Me.ПолеСоСписком21.Enabled = True
End If
End Sub
```

```
Sub Find_TK()
'поиск тарифных коэффициентов для текущего месяца и года
'в таблице mes_god, содержащей эти значения
 Dim rs_mg As New ADODB.Recordset
 Dim s_sql As String
   's_sql – строка запроса на языке SQL
 s_sql = "SELECT mes_god.* FROM mes_god " & _
     " WHERE mes_god.GOD = " & Me.GOD & _
     " and mes_god.MESIJC = " & Me.MES & ""
        'текст SQL-запроса – источник данных Recordset
  'Set rs_rn = CurrentDb.OpenRecordset(s_sql)   'для DAO
 rs_mg.Open s_sql, CurrentProject.Connection, _
           adOpenStatic, adLockReadOnly
     ' открыть RecordSet - 2-й вариант - ADO
     ' для текущего проекта Access,динамический курсор,
     ' оптимистическая блокировка записей
  With rs_mg        'далее к методам и свойствам объекта rs_mg
                    ' можно обращаться, начиная с точки
   ts(1) = !R1
   ts(2) = !R2
   ts(3) = !R3
   ts(4) = !R4
   ts(5) = !R5
   ts(6) = !R6
  .Close
  End With
End Sub
```

Если при исполнении данной программы или последующих появляется сообщение об ошибке компиляции *"User-defined type not defined"*, значит Вы не подключили библиотеку **ADO** (см. рисунок 5.55).

В окне кода формы *Список нарядов* присутствуют также и другие процедуры для различных объектов данной формы.

- Для поля *GOD* процедура события AfterUpdate, которая выполняется при изменении значения этого поля:

```
Private Sub GOD_AfterUpdate()
    Find_TK
End Sub
```

- Для поля со списком *MES* процедура события AfterUpdate, которая выполняется при изменении значения этого поля:

```
Private Sub ПолеСоСписком21_AfterUpdate()
    Find_TK
End Sub
```

- Для кнопки *Расчет* процедура события Click, которая выполняется при нажатии на эту кнопку:

```
Private Sub Расчет_Click()
  ' *** расчет сдельной оплаты по наряду ***
  'Dim rs_rn As DAO.Recordset      ' 1-й вариант - DAO
  Dim rs_rn As New ADODB.Recordset ' 2-й вариант - ADO
  Dim NN As Double
  Dim s_sql As String
  NN = Me.NOMER  ' NOMER - имя поля на форме
                 ' Me - текущий объект - форма Список нарядов
  s_sql = "SELECT rab_nar.*, rab_nar.NOMER " & _
          "FROM rab_nar WHERE rab_nar.NOMER = " & NN & ""
  'Set rs_rn = CurrentDb.OpenRecordset(s_sql)
     ' открыть RecordSet  1-й вариант - DAO
  rs_rn.Open s_sql, CurrentProject.Connection, _
        & adOpenDynamic, adLockOptimistic
     ' открыть RecordSet - 2-й вариант - ADO
     ' для текущего проекта Access,динамический курсор,
     ' оптимистическая блокировка записей
  s_t_vr_ktu = 0   'сумма тариф*вредность*КТУ
  With rs_rn    'далее к методам и свойствам объекта rs_rn
                ' можно обращаться, начиная с точки
   '.Edit 'нужно только для 1-го варианта - DAO
  .MoveFirst
   Do Until.EOF
   'далее после знака ! - обращение к полям rs_rn
      If Not IsNull(!TARIF) Then
         If IsNull(!VREDNOST) Then VRDN = 1 _
                              Else VRDN = !VREDNOST
         s_t_vr_ktu = s_t_vr_ktu + !TARIF * VRDN * !KTU
         Debug.Print !TARIF, s_t_vr_ktu
      End If
     .MoveNext
   Loop
   s_dopl = Me.SUM_nar - _
     [Form_Работники в наряде подч_форма].[Всего_тариф]
 .MoveFirst
  Do Until.EOF
      '.Edit   ' - нужно только для 1-го варианта - DAO
     If IsNull(!VREDNOST) Then VRDN = 1 _
                          Else VRDN = !VREDNOST
    !SUMMA = !TARIF + s_dopl / s_t_vr_ktu * _
                              !TARIF * VRDN * !KTU
   ' Сумма сдельной оплаты = сумме оплата по тарифу +
   ' доплата, распределяемая пропорционально тарифной
   ' оплате с учетом коэффицментов вредности и КТУ
```

```
        !SUMMA = Round(!SUMMA, 2)
        Debug.Print !SUMMA
      .Update
      .MoveNext
    Loop
   .Close
  End With
End Sub
```

Для формы *Затраты по наряду* определена одна процедура для события *AfterUpdate*, которая оперативно обновляет значения сумм затрат по наряду в этом окне и в родительском (Parent) окне (*Список нарядов*):

```
Private Sub SUM_ZATR_AfterUpdate()
  Me.Refresh
  Me.Parent.Refresh
  [Form_Список нарядов].[SUM_nar] = SUM_Z
End Sub
```

Для формы *Работники в наряде подч_форма* определены следующие процедуры:

- Для поля *OTR_CHAS* процедура события *AfterUpdate*, которая выполняется при изменении значения этого поля:

```
Private Sub OTR_CHAS_AfterUpdate()
  Calc_tarif
End Sub
```

- Для поля *OTR_DN* процедура события *AfterUpdate*, которая выполняется при изменении значения этого поля:

```
Private Sub OTR_DN_AfterUpdate()
  OTR_CHAS = 8 * OTR_DN
  Calc_tarif
End Sub
```

- процедура расчета суммы оплаты по тарифу:

```
Private Sub Calc_tarif()
    If Not IsNull(Me.RAZRD) And _
       (IsNull(Me.TARIF_K) Or Me.TARIF_K = 0) _
        Then  Me.TARIF_K = ts(Me.RAZRD)
        Me.TARIF_K = Round(Me.TARIF_K, 2)
    End If
    Me.TARIF = OTR_CHAS * TARIF_K
    Me.Refresh
End Sub
```

В окне *Module1* в разделе General-Declaration содержится описание общего для нескольких окно массива:

```
Public ts(6) As Single
```

В этом окне могут присутствовать также различные служебные программы для работы с информацией базы данных.

Еще один пример одновременной работы с Access и Word – программа чтения данных из таблицы файла *.doc или *.docx с перенесением их в таблицу базы данных. В рассмотренном примере используется таблица ***Список*** базы данных ***Контингент*** (см. раздел учебника 3.3.4.3 Создание базы данных в Microsoft Office Access 2007) и таблица Word (файл *d:\asg\Список студ.doc*), по содержанию соответствующая некоторым полям таблицы ***Список*** (см. таблицу 5.8).

Таблица 5.8. Таблица файла *d:\asg\Список студ.doc*

№ зачетки	Фамилия, имя, отчество	Дата поступления	Факультет	Специальность	Курс	Группа
555274	Логинов Василий Иванович	25.07.2006	5	2603	3	009
555275	Минин Максим Викторович	25.07.2006	5	2603	3	009
555276	Лыжников Евгений Анатольевич	25.07.2006	5	2603	3	009
555278	Ляпин Игорь Васильевич	25.07.2006	5	0701	3	131
555280	Медников Николай Витальевич	25.07.2006	5	2603	3	009
555281	Мальчук Елена Александровна	25.07.2006	5	2603	3	009
555282	Мальцев Вячеслав Викторович	25.07.2006	5	2603	3	007
555283	Митин Владимир Семенович	25.07.2006	5	0701	3	131
555284	Маслов Владимир Михайлович	25.07.2006	5	2603	3	009
555285	Мельников Виктор Петрович	25.07.2006	5	2603	3	009
555288	Кулин Сергей Юрьевич	25.07.2006	5	0701	3	131

Программу следует написать, как процедуру обработки события *Нажатие кнопки (Click)* для новой кнопки с именем ***Данные из Word***, которую можно разместить на новой или существующей форме.

Текст процедуры может быть, например, следующий:

```
Private Sub Данные_из_Word_Click()
On Error Resume Next
Dim s_sql As String
```

```
Dim rs As New ADODB.Recordset
s_sql = "SELECT Список.* FROM Список"
   'текст SQL-запроса - источник данных Recordset
rs.Open s_sql, CurrentProject.Connection, _
       adOpenDynamic, adLockOptimistic
         ' создаем динамический курсор для таблицы Список
         ' с оптимистической блокировкой записей
Set objWord = CreateObject("Word.Application")
          'создаем объект типа Word.Application
objWord.Visible = True  'до завершения отладки использовать Visible
Set objDoc = objWord.Documents.Open("d:\Список студ.doc")
               'открываем файл для объекта objWord
n = objDoc.Tables(1).Rows.Count  'определяем количество строк
                                 'в таблице 1 документа Word
no_dat = ""   ' строка для записи неудачных операций.Update
yes_dat = ""  ' строка для записи удачных операций.Update
For i = 2 To n
    s = objDoc.Tables(1).Rows(i)
        'i-я строка таблицы 1 документа
    fld = Split(s, Chr(13) + Chr(7))
        ' i-тую строку таблицы разбиваем на поля для массива fld
    With rs  ' далее к методам и свойствам объекта rs
             ' можно обращаться, начиная с точки
    .AddNew  'добавить строку в курсор
    .Fields("NZ") = fld(0)      ' присвоение полю NZ
        ' набора записей rs значения элемента массива fld(0)
    .Fields("FIO") = fld(1)            ' ФИО
    .Fields("DATA_P") = CDate(fld(2)) ' дата
    .Fields("N_FCLT") = Eval(fld(3))  ' № фак.
    .Fields("N_SPECT") = fld(4)        ' Код спец
    .Fields("KURS") = Eval(fld(5))     ' Курс
    .Fields("N_GRUP") = fld(6)         ' № группы
    .Update        'пытаемся сохранить курсор в таблице базы
      If Err.Number <> 0 Then 'если сохранить не удается
         no_dat = no_dat & fld(0) & " " & fld(1) & vbLf
      Else
        yes_dat = yes_dat & fld(0) & " " & fld(1) & vbLf
      End If
    End With
```

```
  Next
rs.Close
objDoc.Close
objWord.Quit
MsgBox "Всего в таблице Word " & n & " строк " & vbLf & _
 "Добавлены в таблицу Список записи с Таб.№ :" & vbLf & _
  yes_dat & _
 "Не добавлены из-за нарушений целостности базы::" _
  & vbLf & no_dat,, "Результаты переноса данных"
End Sub
```

Результаты выполнения программы будут показаны в окне MsgBox (см. рисунок 5.61).

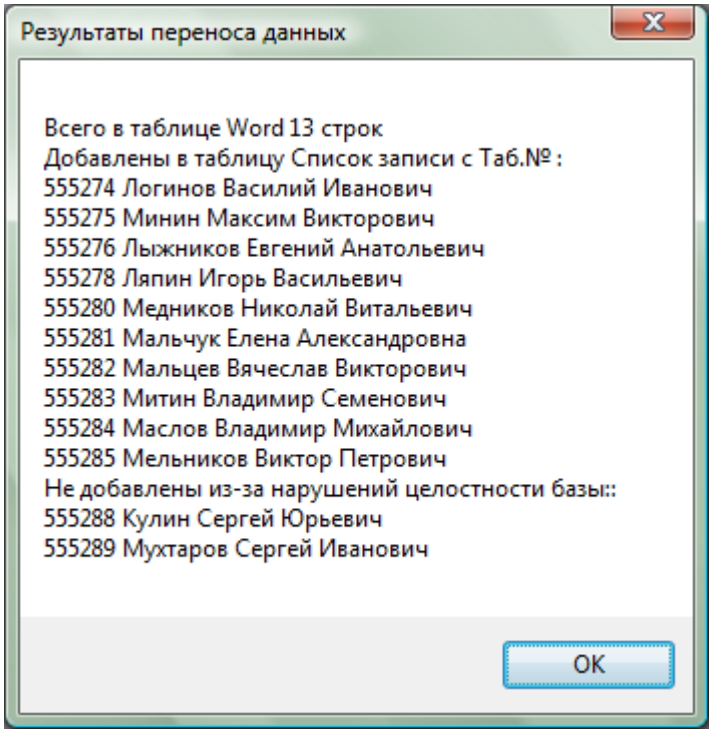


Рисунок 5.61. Сообщение после выполнения процедуры добавления записей в таблицу Access их таблицы Word

Контрольные вопросы к главе 5

1) Какие стандарты существуют в области разработки программного обеспечения?
2) Что такое язык программирования?
3) В чем разница языков программирования низкого и высокого уровня? Назовите примеры этих языков.
4) Что такое трансляция, компиляция и интерпретация при работе с программами на языках высокого уровня?
5) Какие категории алгоритмических языков существуют?
6) Что такое алгоритм, в каком виде он может существовать?
7) Какие три типовые структуры алгоритмов существуют?
8) Перечислите 5 принципов структурного программирования.
9) Что такое объект в языках программирования и его принципы инкапсуляции, наследования и полиморфизма?
10) Каков рейтинг языка Visual Basic, какие разновидности его существуют?
11) Что необходимо для разработки и исполнения на ПК программ на языке Visual Basic Scripting Edition?
12) Что такое лексемы языка программирования, каковы правила образования лексем разного типа в языке VBScript?
13) Какие типы и подтипы данных определены в языке VBScript?
14) Что такое константы, как их можно использовать в VBScript?
15) Что такое переменные и массивы? Каковы правила их обозначения и использования в языке VBScript?
16) В чем разница строчного и блочного операторов условного перехода в языке VBScript?
17) Каково назначение оператора выбора Case?
5) Как можно написать условие окончания цикла для структур ***Do... Loop*** и ***While...Wend*** языка VBScript?
6) В чем отличие оператора цикла ***For...Next*** от ***Do... Loop*** и ***While...Wend*** языка VBScript?
7) Как использовать процедуры и функции пользователя в языке VBScript?
18) Приведите примеры стандартных функций для работы с информацией математической, строковой, с датами и временем.
19) Какие библиотеки объектов может использовать язык VBScript?

20) Что такое свойства и методы объектов, как к ним можно обратиться?
21) Что такое коллекция объектов?
22) Как может использоваться язык VBScript для работы с приложениями системы Microsoft Office?
23) Как может использоваться язык VBScript для выполнения программ на HTML-страницах?
24) Компонентом какой системы является язык Microsoft Visual Basic for Applications (VBA)?
25) Каковы существенные отличия в правилах написания идентификаторов в языках VBScript и VBA?
26) В чем существенная разница в работе с переменными и массивами в языках VBScript и VBA?
27) Как выполняется отладка программ в VBA?
28) Чем является объект ***Selection*** для приложения Word?
29) Элементы управления каких типов можно разместить в документе Word?
30) Чем является объект ***Range*** для приложения Excel?
31) Какие библиотеки использует VBA для работы с базами данных Access?
32) Что такое объект ***Recordset*** при работе с библиотеками ADO и DAO?
33) Как можно использовать свойства объектов *Microsoft Access Object Library* для выполнения вычислений на формах, работающих с базой данных Access?
34) Какова последовательность использования объектов библиотек ADO и DAO для выполнения операций с записями таблиц базы данных Access?

Список литературы

1. Леонтьев В. П. Новейшая энциклопедия персонального компьютера 2010 – М.: ОЛМА Медиа Групп, 2010. – 800 с.
2. Старков В. Архитектура персонального компьютера: организация, устройство, работа. – М.: Горячая Линия – Телеком, 2009. – 536 с.
3. Леонов В. Самоучитель работы на компьютере. – М., 2009. – 352 с.
4. Пол Мак-Федрис. Microsoft Windows 7. Полное руководство. – М.:Вильямс, 2009. – 800 с.
5. Крис Фейли. Мастерская Windows, XP, Vista и Office. – М., 2009. – 608 с.
6. Грег Перри. Microsoft Office 2007. Все в одном.- М.:Диалектика, 2008. – 608 с.
7. Меженный О. А. Microsoft Office 2007. Краткое руководство. – М., 2009. – 384 с.
8. Джон Уокенбах. Microsoft Office Excel 2007. Библия пользователя. – М.:Диалектика, 2008. – 816 с.
9. Майкл Г., Джозеф С., Гэвин П. Microsoft Office Access 2007. Библия пользователя.- М.:Диалектика, 2008. – 1200 с.
10. Слепцова Л. Д. Программирование на VBA в Microsoft Office 2007. Самоучитель. – М., 2009. – 432 с.
11. Леонтьев В.П. Интернет 2010: Универсальный справочник – М.:Олма медиа групп, 2010. – 800 с.
12. Леонтьев В.П. Новейшая энциклопедия Интернета 2010.- М.:Олма медиа групп, 2010. – 640 с.
13. Г. Борн. Руководство разработчика на Microsoft Windows Script Host 2.0. – СПб.:Питер; М.:Изд.-торг.дом "Русская редакция", 2001. – 480 с.
14. *Журавлев А.В.* Персональный компьютер. Просто как дважды два. – М.:Эксмо, 2008. – 288 с.
15. *Журавлев А.В.* Microsoft Windows Vista. Просто как дважды два. – М.:Эксмо, 2007. – 352 с.
16. *Иванова Е.Н.* Microsoft Office 2007. Просто как дважды два. – М.:Эксмо, 2007. – 336 с.
17. *Кушнир А.Н.* Microsoft Office Access 2007. Просто как дважды два. – М.:Эксмо, 2009. – 288 с.
18. *Аксак В.А.* Интернет. Просто как дважды два. – М.:Эксмо, 2009. – 288 с.
19. *Борисенко А.А.* Локальная сеть. Просто как дважды два. – М.:Эксмо, 2009. – 192 с.

Сайты Интернет-ресурсов

1. *www.informika.ru* – сайт Государственного научно-исследовательского института информационных технологий и телекоммуникаций (ФГУ ГНИИ ИТТ "Информика")
2. *www.intel.ru* и *www.intel.com* – корпорация Intel
3. *www.microsoft.ru* и *www.microsoft.com* – корпорация Microsoft
4. *pcnews.ru, computer-news.ru , www.hardvision.ru, news.ferra.ru, www.ixbit.com, www.computerra.ru, www.compulenta.ru, www.comp-life.ru* – компьютерные новости
5. *www.itnews.ru, http://subscribe.ru/catalog/comp, www.studioit.ru, www.it-top.ru, www.cnews.ru, it-technologiess.ru, www.worldnewsit.ru* – новости информационных технологий
6. *www.3dnews.ru* – Daily Digital Digest, все о компьютерах – обзоры, аналитика, новости Hardware, новости Software, сети, программное обеспечение, энциклопедия и пр., тематические рассылки
7. *www.intuit.ru* – Интернет-Университет Информационных Технологий, бесплатные курсы (более 400), обучение, видеокурсы и пр.
8. *www.planet-it.ru* – портал предназначен для аккумулирования различных образовательных мероприятий в области информационных технологий: олимпиад, конкурсов, тестов, удаленного обучения и т.д.
9. *www.citforum.ru* – новости, рассылки и форумы по темам: IT-консалтинг, Software Engineering, программирование, СУБД, безопасность, Internet, сети, операционные системы, Hardware
10. *www.ict.edu.ru* – информационно-коммуникационные технологии в образовании
11. *www.forum.softweb.ru* – форум с разделами «Компьютер для начинающих», «Программы», «Программирование», «Интернет и сети», «Я и компьютер», группы разделов «Образование и работа», библиотека с книгами для скачивания и пр.
12. *http://forum.oszone.net* – форум с различными группами тем в области информационных технологий
13. *www.cyberforum.ru* – форум программистов
14. *http://vbsbook.ru* – справочник по языку программирования VBScript

Приложение 1. Национальные кодовые страницы (CP) Windows

Локализация	ANSI CP	Mac CP	DOS Primary CP (OEM)	DOS Secondary CP (OEM)	Другие CPs
Afrikaans	1252	10000	850	437	
Albania	1250			852	
Arabic	1256		864	710	709, 708,720
Basque	1252	10000	850	437	
Byelorussian	1251	10007	866	855	
Bulgarian	1251	10007	866	855	
Catalan	1252	10000	850	437	
Chinese (Taiwan)	950		950		
Chinese (People's Republic of China)	936		936		
Chinese	950		950	936	
Chinese (Singapore)	936		936		
Croatia	1250			852	
Czech	1250	10029	852		
Danish	1252	10000	850	865	437
Dutch	1252	10000	850	437	
English (United States)	1252	10000	437	850	
English (Britain, Canada и др.)	1252	10000	850	437	
Estonian	1257		775		
Faeroese	1250			850	
Farsi	1256				
Finnish	1252	10000	850	865	
French (Standard)	1252	10000	850	437	
French (Canada)	1252	10000	850	863	437
German (Standard)	1252	10000	850	437	
Greek	1253	10006	737		869
Hebrew	1255		862		

Локализация	**ANSI CP**	**Mac CP**	**DOS Primary CP (OEM)**	**DOS Secondary CP (OEM)**	**Другие CPs**
Hungarian	1250	10029	852		
Icelandic	1252	10079	850	861	865
Indonesian	1252	10000	437	850	
Italian	1252	10000	850	437	
Japanese	932	10001	932	437	942
Korean	949		949	1361	
Korean (Johab)	1361			1361	
Latvian	1257	775			
Lithuanian	1257	775			
Norwegian	1252	10000	850	865	437
Norwegian (Nynorsk)	1252	10000	850	865	437
Polish	1250	10029	852		
Portuguese (Standard)	1252	10000	850	860	437
Romanian	1250	10029	852		
Russian	***1251***	***10007***	***866***	**855**	
Serbian (Latin)	1250			852	
Slovak	1250	10029	852		
Slovenian	1250	10029	852		
Spanish	1252	10000	850	437	
Swedish	1252	10000	850	865	437
Thai	874		874		
Turkish	1254	10081	857	852	850
Ukrainian	1251	10007	866	855	

Приложение 2. Некоторые многоядерные процессоры Intel для настольных ПК и их цены

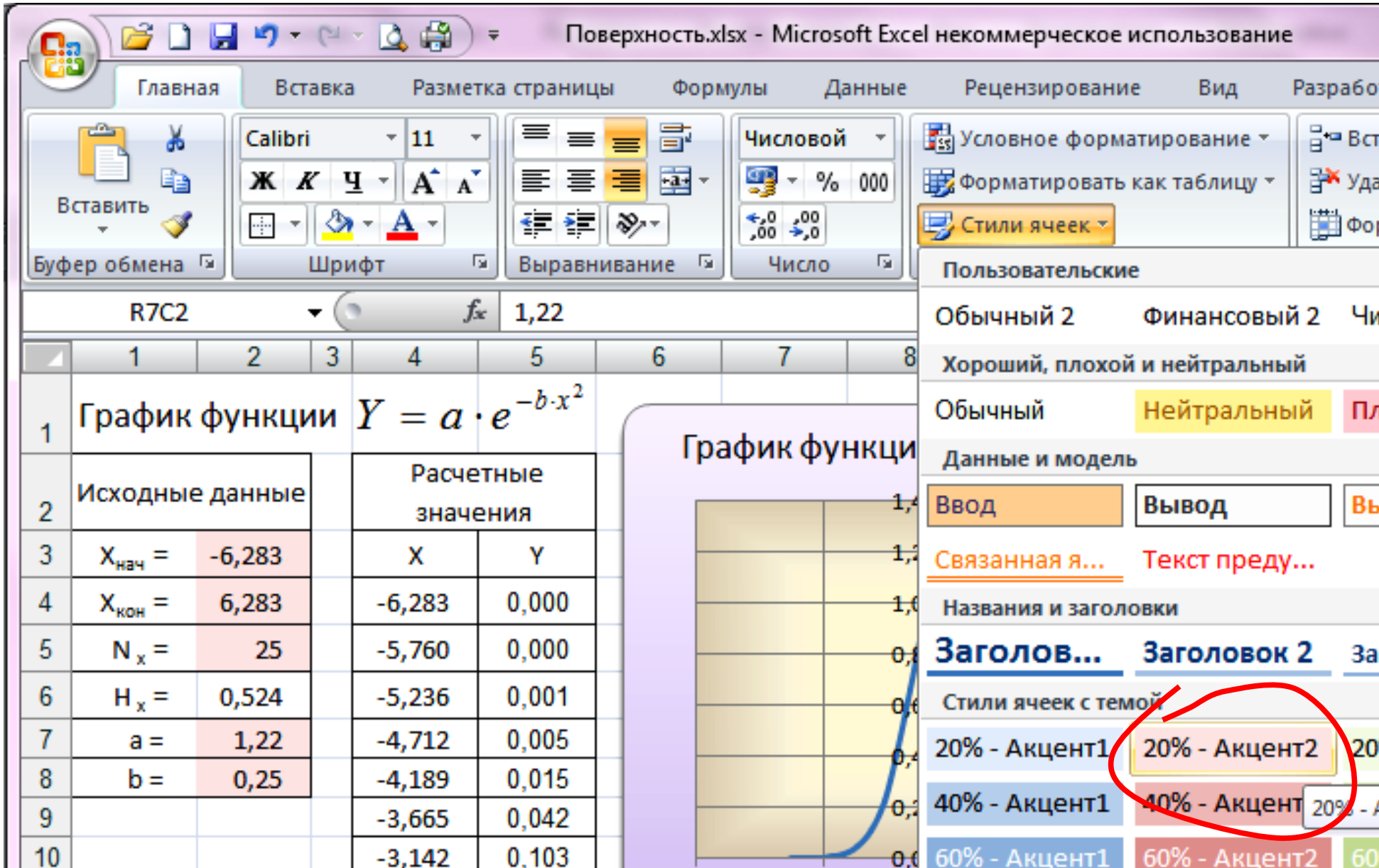
(http://price.ru/, январь 2010 г.)

Марка процессора Intel®	Номер процессора	Количество ядер/ потоков	Тактовая частота, ГГц	Частота системной шины, МГц	Кэш-память второго уровня, Мб	Цена, руб.
Pentium™	G6950	2	2.80	-	3	3048
Core™ i3	530	2/4	2,93	-	4	3933
	540	2/4	3,06	-	4	4794
Core™ i5	650	2/4	3,20-3,46	-	4	6188
	660	2/4	3,33-3,60	-	4	6798
	750	4	2,66	-	8	6680
Core™ i7	960	4/8	3,20-3,46	-	8	19548
	950	4/8	3,06-3,33	-	8	19278
	920	4/8	2,66-3,93	-	8	8928
	860	4/8	2,8-3,46	-	8	9206
Core™2 Extreme	QX9755	4	3,20	1600	12	30560
	QX9650	4	3,00	1333	12	33258
Core™2 Quad	Q9550	4	2,83	1333	12	8760
	Q8400	4	2,66	1333	4	4950
Core™2 Duo	E8300	2	2,83	1333	6	4286
	E8200	2	2,66	1333	6	4271
	E7500	2	2,93	1066	3	3441
	E7400	2	2,80	1066	3	3260
Pentium® Dual-Core	E6300	2	2,80	800	2	2294
	E5300	2	2,60	800	2	1932
	E2160	2	1,80	800	1	1811
	E2140	2	1,60	800	1	1626

Приложение 3. Различия выпусков ОС Windows 7

Возможности	Home Basic Домашняя базовая	Home Premium Домашняя расширенная	Professional Профессио-нальная	Ultimate Максималь-ная
Список последних файлов для недавно использовавшихся программ в меню Пуск и для значков панели задач (Jump Lists)	✓	✓	✓	✓
Новый браузер Internet Explorer 8	✓	✓	✓	✓
Windows Media Center и Internet TV		✓	✓	✓
Интерфейс Aero Glass и улучшенная навигация на Рабочем столе		✓	✓	✓
Создание домашней группы для общего доступа к файлам и принтерам в домашней сети		✓	✓	✓
Запуск программ для Windows XP в режиме Windows XP			✓	✓
Упрощенное и более безопасное подключение к корпоративным сетям с помощью функции присоединения к домену			✓	✓
Возможность архивации данных в домашней или рабочей сети			✓	✓
Защита данных на компьютере и портативных устройствах хранения от потери и кражи с помощью шифрования диска BitLocker				✓
Выбор нужного языка интерфейса и возможность переключения между 35 доступными языками				✓

Приложение 4. Использование галереи стилей в Excel 2007



Приложение 5. Использование пакета анализа в Excel

Обработка результатов наблюдений (ГОСТ 8.207 - 76)

Исходные данные - результаты параллельных наблюдений (выборочная дисперсия)

x_i =	1,00	1,20	0,85	0,99	1,45	0,97	1,25	0,86	1,90
$(x_i - x_{ср})^2$ =	0,027	0,001	0,098	0,030	0,082	0,037	0,008	0,092	0,543

1) Среднее арифметическое результатов наблюдений

$\bar{x}$ = *СРЗНАЧ(B4:J4)* = **1,1633**

2) Среднее квадратическое отклонение (стандартное отклонение)

(дисперсия выборки в таблице равна S^2)

Можно рассчитать по формулам $S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}} = \sqrt{\frac{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}{n(n-1)}} =$ **0,3387**

или использовать статистическую функцию Excel *СТАНДОТКЛОН(B4:J4)* = **0,3387**

Сервис - Анализ данных - Описательная статистика

Среднее	**1,1633**
Стандартная ошибка	0,1129
Медиана	1
Мода	#Н/Д
Стандартное отклонение	**0,3387**
Дисперсия выборки	0,1147
Эксцесс	1,9874
Асимметричность	1,4412
Интервал	1,05
Минимум	**0,85**
Максимум	**1,9**
Сумма	10,47
Счет (количество точек)	**9**
Уровень надежности(95.0%)	**0,2604**

3) Исключение грубых погрешностей по ГОСТ 11.002-73

X_{min} = *МИН(B4:J4)* = **0,85**

X_{max} = *МАКС(B4:J4)* = **1,90**

$$U_1 = \frac{X_{max} - \bar{X}}{S} = \frac{1,9 - 1,16333}{0,338748} = 2,17468$$

$$U_2 = \frac{\bar{X} - X_{min}}{S} = 0,92498$$

Для сравнения с h-критерием берем большее значение U_1 = 2,17468

Таблица h - критерия

n	3	4	5	6	7	8	9	10
$h_{0,05}$	1,15	1,46	1,67	1,82	1,94	2,03	2,11	2,18

n	25	30	40	50	100	250	300
$h_{0,05}$	2,87	2,928	3,015	3,082	3,285	3,534	3,703

$h_{0,05}(9)$ = 2,11 **Если U >= h, результат можно исключить, в противном случае его исключать нельзя**

Для ряда 1 1.2 0.85 0.99 1.45 0.97 1.25 0.86 1.9 U_1 = 2.174677 > $h_{0.05}(9)$ = 2.11

Следует исключить максимальный результат = 1.9

Для ряда 1 1.2 0.85 0.99 1.45 0.97 1.25 0.86

S = *СТАНДОТКЛОН(B4:I4)* = 0,2096 U_1 = 1,8071 U_2 = 1,05565

$h_{0,05}(8)$ = 2,03

Все остальные результаты, кроме исключенного 1.9 можно использовать для расчета погрешности.

Сервис - Анализ данных - Описательная статистика

Среднее	1,0713
Стандартная ошибка	0,0741
Медиана	0,995
Мода	#Н/Д
Стандартное отклонение	**0,2096**
Дисперсия выборки	0,0439
Эксцесс	-0,203
Асимметричность	0,8272
Интервал	0,6
Минимум	0,85
Максимум	1,45
Сумма	8,57
Счет	**8**
Уровень надежности(95.0%)	**0,1752**

4) Оценка среднего квадратического отклонения результата наблюдения

$\bar{x}$ = **1,0713**

$$S(\bar{x}) = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n(n-1)}} = 0,0741 \quad \text{или} \quad S(\bar{x}) = \frac{S}{\sqrt{n}} = 0,0741$$

$(x_i - x_{ср})^2$ =	0,005	0,017	0,049	0,007	0,143	0,010	0,032	0,045

Для вероятности P = 0.95 (α=0.05) и n - 1 = 7 находим критерий Стьюдента

$t_{0,05}(7)$ = *СТЬЮДРАСПОБР(0.05;7)* = **2,36462**

5) Случайная погрешность результата измерения

$\xi = t \cdot S(\bar{x}) =$ **2.364623 * 0.0741 = 0,175219** (т.о., случайная погрешность представлена в "Описательной статистике", как. строка "Уровень надежности"- см. табл.)

6) Результаты измерения представляют в форме $\bar{x} \pm \Delta, P$

где Δ - погрешность (равна случайной погрешности ε при отсутствии систематической погрешности)

P - доверительная вероятность, %

Результат измерения следует записать, как 1.1 ± 0.2, P = 95%

Однофакторный дисперсионный анализ

Метод позволяет установить, существенно или нет влияние различных факторов (например, температуры, концентрации веществ или катализаторов, времени) на результат эксперимента в исследуемом диапазоне их изменения.

Исходные данные для анализа - значения результатов эксперимента при различных значениях некоторого фактора.

Если известна дисперсия результатов эксперимента (внутригрупповая дисперсия), достаточно по одному результату для каждого значения фактора.

Если неизвестна - необходимо иметь несколько параллельных результатов для каждого значения фактора.

Вариант 1. Внутригрупповая дисперсия известна.

Из предварительных 5-ти парралельных определений при одном из значений параметра эксперимента - температуры (выполненных также для для определения погрешности метода) известна дисперсия результата измерения - выхода продукта S^2

При различных значения температуры получили различный выход продукта x :

x_i	35,2	37,8	42,4	44,7	48,0	52,1

Необходимо установить, существенно или нет влияние температуры на выход продукта.

Для этого рассчитывается межгрупповая дисперсия и сравнивается с внутригрупповой.

$$S_1^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1} = \frac{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}{n(n-1)} = \quad \mathbf{39,63}$$

Сервис - Анализ данных - Описателная статистика	
Среднее	43,36667
Стандартная ошибка	2,569911
Медиана	43,55
Мода	#Н/Д
Стандартное отклонение	6,294972
Дисперсия выборки	**39,627**
Эксцесс	-1,03855
Асимметричность	0,059971
Интервал	16,9
Минимум	35,2
Максимум	52,1
Сумма	260,2
Счет	6
Уровень надежности(95.0%)	6,606157

Для сравнения дисперсий вычисляется значение критерия Фишера и сравнивается с табличным значением:

$$F = \frac{S_1^2}{S_2^2} = \frac{39,62667}{2,5} = \mathbf{15,851}$$

$k_1 = m - 1 = 6 - 1 = 5$

$k_2 = m(n - 1) = 6(5 - 1) = 24$

Табличное значения критерия Фишера $F(0.05, k_1, k_2)$ = *FРАСПОБР(0,05;5;24)* = 2,6207

Влияние исследуемого фактора существенно, т.к. $F > F_{табл.}$

Вариант 2. Внутригрупповая дисперсия неизвестна.

Расчет дисперсий проводится по формулам:

Внутригрупповая дисперсия

$$S_2^2 = \frac{1}{m(n-1)} \sum_{i=1}^{m} \sum_{j=1}^{n} (x_{ij} - x_{i\,cp})^2$$

Межгрупповая дисперсия

$$S_1^2 = \frac{1}{m-1} \sum_{i=1}^{m} (x_{i\,cp} - x_{cp})^2$$

где m – количество уровней фактора;

n – число параллельных определений на каждом уровне;

x_{ij} – результат наблюдения на i - том уровне в j - том определении;

$x_{i\,cp}$ – групповое средгнее арифметическое для I - того уровня;

x_{cp} – общее среднее арифметическое значение.

$$x_{i\,cp} = \frac{1}{n}\sum_{j=1}^{n} x_i x_{cp} = \frac{1}{n}\sum_{i=1}^{m} x_{i\,cp} = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}$$

Пример расчета 1.

Были получены по 3 параллельных результата для шести значений изучаемого фактора (температуры):

m	t, °C	Паралельные результаты эксперимента n			$x_{i\,cp}$	$(x_{cp} - x_{cp})^2$	$\frac{1}{(n-1)}\sum_{j=1}^{n}(y_j - y_{i\,cp})^2$
		1	2	3			
1	120	33,2	33,8	34,5	33,8333	67,3311	0,4233
2	130	36,5	35,8	36,7	36,3333	32,5534	0,2233
3	140	40,6	40,2	41,8	40,8667	1,3741	0,6933
4	150	42,3	43,1	43,7	43,0333	0,9889	0,4933
5	160	45,8	46,3	47,5	46,5333	20,2000	0,7633
6	170	50,9	51,8	52,2	51,6333	92,0534	0,4433
					x_{cp} 42,0389	42,9002	0,5067

S_1^2

S_2^2

Описательная статистика для $x_{i\,cp}$	
Среднее	42,0389
Стандартная ошибка	2,6740
Медиана	41,95
Мода	#Н/Д
Стандартное отклоне	6,5498
Дисперсия выборки	42,9002
Эксцесс	-0,74337
Асимметричность	0,2510
Интервал	17,8
Минимум	33,8333
Максимум	51,6333
Сумма	252,2333
Счет	6
Уровень надежности(	6,8736

В MS Excel для представленной таблизы результатов рассчитаны итоги дисперсионного анализа **(Сервис - Анализ данных - Однофакторный дисперсионнвй анализ):**

Однофакторный дисперсионный анализ

ИТОГИ

Группы	*Счет*	*Сумма*	*Среднее*	*Дисперсия*
Строка 1	3	101,5	33,8333	0,4233
Строка 2	3	109	36,3333	0,2233
Строка 3	3	122,6	40,8667	0,6933
Строка 4	3	129,1	43,0333	0,4933
Строка 5	3	139,6	46,5333	0,7633
Строка 6	3	154,9	51,6333	0,4433

$\frac{128,701}{3} = 42,9002$

Дисперсионный анализ

Источник вариации	*SS*	*df*	*MS*	*F*	*P-Значение*	*критическое*
Между группами	643,5028	5	**128,701**	254,0143	9,741E-12	**3,1059**
Внутри групп	6,08	12	**0,50667**			
Итого	649,5828	17				

$$F = \frac{S_1^2}{S_2^2} = \frac{42,9002}{0,5067} = \mathbf{84,6714}$$

Таким образом, при использовании пакета анализа MS Excel, следует сравнивать **F деленное на число параллельных определений (254.01/3=84.671) и F-критическое (3.1059)**

Табличнје значения критерия Фишера $F(0.05, k_1, k_2)$ = ***FРАСПОБР(0,05;5;12)*** **3,1059**

k1 = m - 1 = 6 - 1 = 5

k2 = m(n - 1) = 6(3 - 1) = 12

Влияние исследуемого фактора существенно, т.к. $F > F_{табл.}$

Приложение 6. Спецификации системы Microsoft Office Access

Атрибут	Максимальное значение
База данных	
Размер файла базы данных	2 Гбайт
Число объектов в базе данных	32 768
Модули	1 000
Число знаков в имени объекта	64
Число знаков в пароле	14
Число знаков в имени пользователя или имени группы	20
Число одновременно работающих пользователей	255
Таблица	
Число знаков в имени таблицы	64
Число знаков в имени поля	64
Число полей в таблице	255
Число открытых таблиц	2048
Размер таблицы	2 Гбайт
Число знаков в текстовом поле	255
Число знаков в поле MEMO	65 535 при вводе данных через интерфейс пользователя; 1 Гбайт для хранения знаков при программном вводе данных.
Размер поля объекта OLE	1 Гбайт
Число индексов в таблице	32
Число полей в индексе	10
Число знаков в сообщении об ошибке	255
Число знаков в условии на значение записи	2 048
Число знаков в описании таблицы или поля	255
Число знаков в записи (кроме полей MEMO и полей объектов OLE)	2 000
Число знаков в значении свойства поля	255
Запрос	
Число установленных связей	32 на одну таблицу за вычетом числа индексов, не участвующих в связях
Число таблиц в запросе	32
Число полей в наборе записей	255
Размер набора записей	1 Гбайт
Предел сортировки	255 знаков
Число уровней вложения запросов	50

Атрибут	Максимальное значение
Число знаков в ячейке на бланке запроса	1 024
Число знаков для параметра в запросе с параметрами	255
Число операторов AND в предложении WHERE или HAVING	99
Число знаков в инструкции SQL	приблизительно 64 000
Форма и отчет	
Число знаков в надписи	2 048
Число знаков в поле	65 535
Ширина формы или отчета	22 дюйма (55,87 см)
Высота раздела	22 дюйма (55,87 см)
Высота всех разделов плюс заголовки разделов (в режиме конструктора)	200 дюймов (508 см)
Число уровней вложения форм или отчетов	7
Число полей или выражений, которые можно отсортировать или сгруппировать в отчете	10
Число заголовков и примечаний в отчете	1 заголовок/примечание отчета; 1 заголовок/ примечание страницы; 10 заголовков/примечаний групп
Число печатных страниц в отчете	65 536
Число элементов управления и разделов, которые можно добавить за время существования формы или отчета	754
Число знаков в инструкции SQL, работающей в качестве свойства **Источник записей (RecordSource)** или **Источник строк (RowSource)** формы, отчета или элемента управления (оба .mdb и .adp)	32 750
Макрос	
Число макрокоманд в макросе	999
Число знаков в условии	255
Число знаков в комментарии	255
Число знаков в аргументе макрокоманды	255

Приложение 7. Математические функции языка VBScript

Синтаксис функции	Описание
Abs(*числ.выраж.)*	Абсолютная величина числового выражения
Array(сп.эл.массива)	Создание массива из списка
Atn(*числ.выраж.)*	Арктангенс, числовое выражение в радианах
Cbool(*числ.выраж.)*	Возвращает True для ненулевого числового выражения, False для 0, ошибка для нечислового значения
CByte(*выраж.)*	Преобразование в субтип **Byte**
CCur(*выраж.)*	Преобразование в субтип **Currency**
CDbl(*выраж.)*	Преобразование в субтип **Double**
CInt(*выраж.)*	Преобразование в субтип **Integer**
CLng(*выраж.)*	Преобразование в субтип **Long**
Cos(*выраж.)*	Косинус, числовое выражение в радианах
CSng(*выраж.)*	Преобразование в субтип **Single**
CStr(*выраж.)*	Преобразование в субтип **Строка**
Exp(*числ.выраж.)*	Функция экспонента Exp(X) или e^x
Eval(*выражение***)**	Вычисляет выражение и возвращает его результат
Fix(*числ.выраж.)*	Целая часть числа
Int(*числ.выраж.)*	Целая часть числа
FormatCurrency(*числ.выраж.* [,дес.зн. [**,***вед.ноль* [**,***исп.() для отр.чисел* [**,***исп.разд.*]]]])	Форматирование *числ.выраж*. в заданный денежный формат с указанием денежной единицы, заданной в Windows
FormatNumber(*числ.выраж.* [,дес.зн. [**,***вед.ноль* [**,***исп.() для отр.чисел* [**,***исп.разд.*]]]])	Форматирование *числ.выраж*. в заданный формат
FormatPercent(*числ.выраж.* [,дес.зн. [**,***вед.ноль* [**,***исп.() для отр.чисел* [**,***исп.разд.*]]]])	Форматирование *числ.выраж*. в заданный процентный формат (/100) с символом %
Hex(*числ.выраж.)*	Шестнадцатеричное число десятичного *числового выражения*
(*числ.выраж.)* **** (*числ.выраж.)*	Деление двух числовых выражения, результатом которого является целое число (целочисленное деление). До деления результаты округляются до целого числа (<0,5 равно 0, >=0.5 равно 1), у результата операции отбрасывается дробная часть
Lbound(*имя_массива* [,размер])	Наименьшее возможное значение индекса в массиве.. Если размер не указан, он равен 1
Log(*числ.выраж.)*	Натуральный логарифм выражения. Если нужно вычислить логарифм с основанием ***n***, следует использовать выражение: ***Logn(x) = Log(x) / Log(n)***
(*числ.выраж.)* **Mod** (*числ.выраж.)*	Остаток целочисленного деления. Перед делением результаты *численных выражений* округ-

Синтаксис функции	Описание
	ляются до целых чисел
Oct(*число)*	Строка, представляющая восьмеричную величину числа
Randomize	Инициализация генератора случайных чисел
Rnd[(*числ.выраж.)]*	Случайное число большее или равное нулю, но меньшее 1.
RGB(*red*, *green*, *blue)*	Преобразование цветового значения, где каждый из трех цветов в диапазоне от 0 до 255 в одно число, рассчитанное по формуле: ***CLng(red + (green * 256*) *+ (blue* 65536*)*)
Round(*числ.выраж.* [, дес.зн.])	Число, округленное до заданного количества десятичных знаков. Если дес.зн. не указано, его значение равно 0
Sgn(*числ.выраж.)*	Знак числа (>0 – 1, 0 – 0, <0 – -1)
Sin(*числ.выраж.)*	Синус угла, выраженного в радианах
Sqr(*числ.выраж.)*	Квадратный корень числа
Tan(*числ.выраж.)*	Тангенс угла, выраженного в радианах
TypeName (*varИмя)*	Название типа переменной (см. далее для **VarType**)
Ubound(имя_*массива* [, размер])	Наибольшее возможное значение индекса в массиве.. Если размер не указан, он равен 1
VarType(*varИмя)*	Числовое обозначение типа переменной: **vbEmpty** 0 **vbNull** 1 **vbInteger** 2 **vbLong** 3 **vbSingle** 4 **vbDouble** 5 **vbCurrency** 6 **vbDate** 7 **vbСтрока** 8 **vbObject** 9 **vbError** 10 **vbBoolean** 11 **vbVariant** 12 **vbDataObject** 13 **vbByte** 17 **vbArray** 8192

Приложение 8. Строковые функции языка VBScript

Синтаксис функции	Описание
Asc(*строка*)	ANSI-код первого символа в строке
Chr(*число***)**	Символ заданного ANSI-кода
Eval(*выражение***)**	Вычисляет выражение и возвращает его результат
InStr([*нач.поз.***,**] *строка1***,** *строка2* [**,** *тип сравн.*])	Позиция строки2 в строке1 начиная с нач.поз. поиска для заданного типа сравнения (vbBinaryCompare или vbTextCompare, если не указано, то первый)
InStrRev(*строка1***,** *строка2* [**,** *нач.* [**,** *тип сравн.*]])	То же, что и **InStr**, но номер позиции с конца строки
Join(имя_массива [, *разделитель*])	Строка, созданная из элементов массива
LTrim(*строка***), RTrim(***строка***), Trim(***строка***)**	Строка без пробелов слева (**LTrim**), справа (**RTrim**), или без тех и других (**Trim**).
LCase(*строка***)**	Преобразует все символы строки в строчные
Left(*строка***,** *длина*)	Возвращает заданное количество символов с начала строки
Len(*строка* | *имя_пер.*)	Число символов в строке или строковой переменной
Mid(*строка***,** *нач.*[**,** *длина*])	Возвращает заданное количество символов с заданной позиции **нач**. в строке
Replace(*исх_стр, стр_поиска, стр_замены [, нач. [, колич. [, тип сравн.]]]*)	Замена в исходной строке *строки поиска* на *строку замены,* начиная с позиции ***нач.,*** заданное количество раз
Right(*строка***,** *длина*)	Возвращает заданное количество символов с конца строки
Space(*количество***)**	Строка из заданного количества пробелов
Split(*исх_стр.*[**,** *разделитель* [**,** *количество*[**,***.[, тип сравн.*]]]**)**	Возвращает одномерный массив строк, полученный разбиением *исх_стр.* по *разделителям* на заданное *количество* частей. Если *разделитель* не указан, за него принимается знак пробела.
StrComp(*строка1***,** *строка2* [**,***тип сравн.*]**)**	Сравнение строк. Если строка1<строка2, возвращается -1, если строка1=строка2, возвращается 0, если строка1>строка2, возвращается 1.
String(*количество***,** *символ*)	Создает строку из заданного количества заданных символов
StrReverse(*строка*)	Переворачивает строку задом-наперед
UCase(*строка*)	Преобразует все символы строки в прописные

Приложение 9. Функции работы с датой и временем языка VBScript

Синтаксис функции	Описание
CDate(*стр.выр.*)	Преобразование строкового выражения в подтип Дата-время
DateAdd(*формат*, количество, "*дата-время*")	Дата-время, к которому добавлен заданное количество времени для заданного формата: "*yyyy*" Год "*q*" Квартал "*m*" Месяц "*y*" День года "*d*" День "*w*" день недели "*ww*" Week of year "*h*" Час "*n*" Минута "*s*" Секунда
DateDiff(*формат*, *дата1*, дата*2* [,*первый_день_нед*[, первая_нед_года]])	Интервал между двумя временными интервалами в заданном формате
DatePart(*формат*, *дата* [,*первый_день_нед*[, первая_нед_года]])	Часть даты в заданном формате
DateSerial(*год*, *месяц*, *день*)	Дата для заданных *года*, *месяца и дня*
DateValue("*дата-время*")	Подтип Дата-время для заданного строкового выражения
Day("*дата-время*")	День месяца – целое число лт 1 до 31
FormatDateTime(*дата*[, *формат*)	Форматирование *даты* в заданный формат: **vbGeneralDate, vbLongDate, vbShortDate, vbLongTime, vbShortTime**
Hour(*время*)	Час дня, целое число от 0 до 23
Minute(*время*)	Минуты часа, целое число от 0 до 59
Month(*дата*)	День месяца, целое число от 1 до 12
MonthИмя(*месяц*[, сокр_назв])	Строковое обозначение месяца, сокращенное название (True) или полное (False). Если не указано, False – полное название
Now	Текущая дата и время системных часов
Second(*время*)	Секунды в минуте, целое число от 0 до 59
Time	Текущее время системных часов
TimeSerial(*час*, *минута*, *секунда*)	Время для заданных значений часа, минуты и секунды
TimeValue("*время*")	Преобразование строкового значения времени в подтип Время
Weekday(*дата,* [*первый_день_нед.*])	Целое число – день недели
WeekdayИмя(*день_нед.*, *сокр.*, *первый_день_нед.*)	Строка, показывающая название заданного дня недели
Year(*дата*)	Целое число – год для заданной даты

Приложение 10. Логические функции и операторы языка VBScript

Синтаксис функции	Описание
IsArray(*переменная*)	***True***, если переменная = массив
IsDate(*выражение*)	***True***, если *выражение может быть преобразовано в дату*
IsEmpty(*выражение*)	**True**, если *выражение пустое*
IsNull(*выражение*)	***True***, если *выражение* не содержит данных (значение ***Null***)
IsNumeric(*выражение*)	***True***, если значение *выражения – число*
IsObject(*выражение***)**	***True***, если *выражение* - объект
Результат **=** *выражение1* ***And*** *выражение2*	Логическая конъюнкция. Если выражения слева и справа от него истинны, результат ***True,*** иначе ***False*** или ***Null***
Результат **=** *выражение1* **Eqv** *выражение2*	Проверка эквивалентности двух выражений. Возвращает ***True***, если они имеют одинаковое значение
Результат **=** *выражение1* ***Imp*** *выражение2*	Логическая импликация. Выражение ***E1 Imp E2*** возвращает ***False***, если ***E1 = True*** и ***E2 = False***, во всех остальных случаях – ***True***
Результат **=** *object1* ***Is*** *object2*	Проверка эквивалентности двух объектов
Результат **=** ***Not*** *выражение*	Логическое отрицание. Возвращает ***True***, если условие ложно и наоборот
Результат **=** *выражение1* ***Or*** *выражение2*	Логическая дизъюнкция. Должно быть истинным хотя бы одно из выражений
Результат **=** *выражение1* ***Xor*** *выражение2*	Логическое исключение. Выражение ***E1 Xor E2*** возвращает ***True***, если только ***E1 = True*** или только ***E2 = True***, иначе – ***False***

Приложение 11. Методы и свойства объекта ADO.Recordset

Свойства/Коллекции

Имя	Краткое описание
AbsolutePage	Указывает, на какой странице находится текущая запись
AbsolutePosition	Указывает порядковую позицию текущей записи в объекте Recordset
ActiveCommand	Указывает объект *Command*, созданный связанным объектом Recordset
ActiveConnection	Указывает, какому объекту *Connection* в настоящее время принадлежат указанная команда, Recordset, или запись
BOF, EOF	**BOF** – указывает, что текущая позиция – перед первой записью в объекте Recordset. **EOF** – указывает, что текущая позиция – после последней записи
Bookmark	Указывает закладку, которая однозначно определяет текущую запись в объекте Recordset или устанавливает текущую позицию в объекте Recordset на запись, идентифицированную значением закладки *Bookmark*
CacheSize	Указывает число записей объекта Recordset, которые кэшируются в памяти
CursorLocation	Указывает местоположение курсора
CursorType	Указывает тип курсора, используемого в объекте Recordset
DataMember	Указывает Имя компонента данных, который будет найден для объекта, на который ссылается свойство DataSource
DataSource	Указывает объект, содержащий данные, которые представлены как объект Recordset
EditMode	Указывает статус редактирования текущей записи
Fields *(Collection)*	Коллекция полей для записи *Recordset*
Filter	Указывает фильтр данных для *Recordset*
Index	Указывает имя индекса, заданного для объекта *Recordset*
LockType	Указывает тип блокировки
MarshalOptions	Указывает, какие записи должны быть возвращены назад на сервер
MaxRecords	Указывает максимальное количество записей, которое возвращает Recordset для запроса
PageCount	Указывает сколько страниц данных содержит объект *Recordset*
PageSize	Указывает из сколько записей состоит одна логическая страница данных в *Recordset*.
Properties *(Collection)*	Коллекция свойств объекта
RecordCount	Указывает количество записей в объекте *Recordset*
Sort	Указывает одно имя поля или несколько по которым объект *Recordset* сортируется в порядке возрастания или убывания

Имя	Краткое описание
Source	Указывает источник данных для объекта Recordset
State	Состояние объекта: открыт, закрыт; соединяется, исполняется, выполняется
Status	Статус текущей записи
StayInSync	Указывает, в иерархическом объекте Recordset, выполняются или нет изменения соответствующих дочерних записей, если позиция родительской строки изменяется

Методы

Имя	Краткое описание
AddNew	Создание новой записи в обновляемом объекте Recordset
Cancel	Отменить выполнение
CancelBatch	Отменить выполнение пакетного обновления
CancelUpdate	Отменить обновление для объекта *Recordset* или коллекции полей или объекта *Record* до вызова метода *Update*
Clone	Создать дубликат объекта *Recordset* (только для чтения)
Close	Закрыть объект и все подчиненные объекты
CompareBookmarks	Сравнивает две закладки и возвращает их относительное положение (CompareEnum)
Delete	Удаляет текущую запись или группу записей
Find	Поиск записи в объекте *Recordset* для заданных условий
GetRows	Переносит множество записей объекта *Recordset* в массив
GetString	Возвращает *Recordset* как строку с разделителями
Move	Перемещает текущую позицию в объекте *Recordset*
MoveFirst, MoveLast, MoveNext, MovePrevious	Перемещает текущую позицию в объекте *Recordset* в начало, в конец, на следующую или предыдущую запись
NextRecordset	Очищает текущий объект Recordset и возвращает следующий Recordset при продвижении через ряд команд
Open	Открыть курсор
Requery	Обновляет данные в объекте Recordset перезапуском запроса, на котором базируется этот объект
Resync	Обновляет данные в текущем объекте Recordset, или коллекции полей объекта из основной базы данных
Save	Сохраняет Recordset в файле, или объекте *Stream*
Seek	Ищет индекс Recordset, чтобы быстро определить запись, которая соответствует указанным значениям, и изменяет текущую позицию строки в объекте Recordsetё
Supports	Определяет, поддерживает ли указанный объект Recordset специфический тип функциональности
Update	Сохраняет любые изменения, которые Вы делаете в текущей строке объекта Recordset, или коллекции полей объекта *Record*
UpdateBatch	Записывает все пакетные обновления на диск

События (events)

Имя	Краткое описание
EndOfRecordset	Происходит, когда выполняется попытка переместиться за последнюю запись объекта Recordset
FetchComplete	Происходит после того, как все записи в длинной асинхронной операции были найдены в Recordset
FetchProgress	Происходит периодически в течение длинной асинхронной операции, чтобы сообщить, сколько записей было найдено в операции выборки
WillChangeField, FieldChangeComplete	Событие WillChangeField происходит прежде, чем изменяется значение одного или более полей в Recordset. Событие FieldChangeComplete происходит после того, как значение одного или более полей изменилось
WillChangeRecord, RecordChangeComplete	Событие WillChangeRecord происходит перед изменением одной или более записей в Recordset. Событие RecordChangeComplete происходит после одного или более изменений записей
WillChangeRecordset, RecordsetChangeComplete	Событие WillChangeRecordset происходит прежде, чем изменяется Recordset. Событие RecordsetChangeComplete происходит после того, как Recordset изменился
WillMove, MoveComplete	Событие WillMove происходит прежде, чем изменяется текущая позиция в Recordset. Событие MoveComplete происходит после изменения текущей позиции в Recordset