

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА  
И ПРОДОВОЛЬСТВИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

ГЛАВНОЕ УПРАВЛЕНИЕ ОБРАЗОВАНИЯ,  
НАУКИ И КАДРОВОЙ ПОЛИТИКИ

Учреждение образования  
«БЕЛОРУССКАЯ ГОСУДАРСТВЕННАЯ  
ОРДЕНОВ ОКТЯБРЬСКОЙ РЕВОЛЮЦИИ  
И ТРУДОВОГО КРАСНОГО ЗНАМЕНИ  
СЕЛЬСКОХОЗЯЙСТВЕННАЯ АКАДЕМИЯ»

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

## КУРС ЛЕКЦИЙ

*Рекомендовано учебно-методическим объединением  
по образованию в области сельского хозяйства  
в качестве учебно-методического пособия для студентов  
учреждений образования, обеспечивающих получение общего  
высшего образования по специальностям  
6-05-0811-02 Производство продукции животного происхождения,  
6-05-0831-01 Водные биоресурсы и аквакультура*

Горки  
БГСХА  
2023

УДК 004.9(075.8)  
ББК 32.973.202я73  
И74

*Рекомендовано методической комиссией  
факультета биотехнологии и аквакультуры 28.11.2022 (протокол № 3)  
и Научно-методическим советом БГСХА 30.11.2022 (протокол № 3)*

**Авторы:**

кандидат сельскохозяйственных наук, доцент *О. Н. Азаренко*;  
кандидат сельскохозяйственных наук, доцент *Л. И. Ковалевская*;  
кандидат экономических наук, доцент *В. Г. Ракутин*;  
кандидат физико-математических наук, доцент *Д. В. Шаршунов*;  
старший преподаватель *И. П. Барчук*;  
старший преподаватель *Д. В. Воробьев*;  
старший преподаватель *И. К. Мирончиков*;  
старший преподаватель *И. В. Шараева*

**Рецензенты:**

кандидат физико-математических наук, доцент *М. Н. Борисевич*;  
кандидат экономических наук, доцент *М. Н. Садовская*

**Информационные технологии. Курс лекций:** учебно-  
И74 методическое пособие / О. Н. Азаренко [и др.]. – Горки :  
БГСХА, 2023. – 249 с.  
ISBN 978-985-882-392-4.

Рассмотрены теоретические вопросы, терминологический аппарат и другие сведения в области современных информационных технологий.

Для студентов учреждений образования, обеспечивающих получение общего высшего образования по специальностям 6-05-0811-02 Производство продукции животного происхождения, 6-05-0831-01 Водные биоресурсы и аквакультура.

**УДК 004.9(075.8)  
ББК 32.973.202я73**

**ISBN 978-985-882-392-4**

© УО «Белорусская государственная  
сельскохозяйственная академия», 2023

## ВВЕДЕНИЕ

Характерной чертой настоящего времени являются интенсивно развивающиеся процессы информатизации практически во всех сферах человеческой деятельности. Они привели к формированию новой информационной инфраструктуры, которая связана с новым типом общественных отношений, с новой реальностью, с новыми информационными технологиями различных видов деятельности. Современному специалисту необходимо знать информационные технологии, уметь грамотно работать с информацией, анализировать и оценивать собранные данные, готовить доклады, материалы к презентациям и др.

Данный курс лекций поможет студентам получить знания в области современных информационных технологий с учетом требований к уровню подготовки согласно образовательному стандарту Республики Беларусь. Он ориентирован на формирование у студентов компетенций, связанных с пониманием возможностей использования информационных технологий в профессиональной деятельности.

Курс лекций составлен коллективом кафедры экономического анализа и прикладной информатики. Его содержание состоит из пяти лекций. В первой лекции «Базовые понятия информационных технологий», подготовленной доцентом В. Г. Ракутиным, дана системная характеристика, свойства, основные направления развития и компонентная структура информационных технологий; представлены различные их классификации. Над второй лекцией работали доценты О. Н. Азаренко и Л. И. Ковалевская и старшие преподаватели В. Д. Воробьев, И. П. Барчук. В ней описаны программные и методические средства информационных технологий. В третьей лекции, подготовленной старшим преподавателем И. В. Шараевой, изложены главные принципы построения и функционирования компьютерных (вычислительных) сетей, сервисы, предоставляемые глобальной сетью Интернет. Четвертая лекция, автором которой является доцент Д. В. Шаршунов, посвящена базам данных и СУБД Access. Последняя пятая лекция подготовлена старшим преподавателем И. К. Мирончиковым, и в ней рассматриваются принципы организации корпоративных информационных систем, их структура, международные стандарты в области КИС.

Курс лекций предназначен для студентов первого курса, обучающихся по специальностям общего высшего образования 6-05-0811-02 Производство продукции животного происхождения, 6-05-0831-01 Водные биоресурсы и аквакультура, для подготовки к теоретическим и лабораторным занятиям, сдаче зачета по дисциплине «Информационные технологии».

## **Тема 1. ОСНОВНЫЕ ПОНЯТИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

1.1. Информатизация. Информационное общество.

1.2. Понятия информации и информационной системы. Информационные процессы. Информационные технологии.

### **1.1. Информатизация. Информационное общество**

Стремительное развитие цивилизации привело во второй половине XX в. к новому этапу развития общества – переходу от индустриального общества к информационному. Процесс, который обеспечивает этот переход, получил название информатизации.

Информатизация – это организационный, социально-экономический и научно-технический процесс обеспечения потребности органов государственной власти, юридических и физических лиц в получении сведений о лицах, предметах, фактах, событиях, явлениях и процессах на базе информационных систем и сетей, осуществляющих формирование и обработку информационных ресурсов и выдачу пользователю документированной информации.

Информатизация, основанная на развитии информационных технологий, национальных и глобальных телекоммуникационных сетей, открывает принципиально новые возможности организации экономической жизни и трудовой деятельности людей, развития здравоохранения и образования, сохранения и защиты окружающей среды, предупреждения природных, социально-экономических и политических чрезвычайных ситуаций и кризисов, создания системы эффективного государственного управления и сбалансированного развития международных отношений. Информатизация, включающая бурный рост глобальных телекоммуникационных сетей, прежде всего сети Интернет, коренным образом меняет социально-политическую и культурную жизнь миллионов людей на всех континентах, ведет к формированию единого мирового информационного пространства. Республика Беларусь как государство, обладающее достаточно развитой информационной и телекоммуникационной инфраструктурой, развитой экономикой и высоким интеллектуальным потенциалом, может и должна быть составной и неотъемлемой частью этого пространства.

Информационное общество – общество, в котором большинство работающих занято производством, хранением, переработкой и реализацией информации, особенно высшей ее формой – знаний. Движущей

силой развития общества должно стать производство не материального, а информационного продукта. Материальный продукт станет более информационно емким, что означает увеличение доли инноваций, дизайна и маркетинга в его стоимости.

На уровне программных документов данный приоритет закреплён в Национальной стратегии устойчивого социально-экономического развития Республики Беларусь на период до 2030 г., одобренной Президиумом Совета Министров Республики Беларусь 10 февраля 2015 г.

К основным факторам, способствующим развитию информатизации в Республике Беларусь, относятся:

- устойчивая и эффективная политическая система;
- достаточно высокий уровень валового внутреннего продукта на душу населения;
- признание информатизации в качестве одного из национальных приоритетов устойчивого развития и совершенствование правового регулирования ее процессов;
- развитая собственная информационная индустрия, стимулируемая государством;
- высокий образовательный уровень населения.

Развитие информатизации в Республике Беларусь в течение 2011–2015 гг. осуществлялось в соответствии со Стратегией развития информационного общества на период до 2015 г., утвержденной постановлением Совета Министров Республики Беларусь от 9 августа 2010 г. № 1074, и разработанными для ее выполнения Национальной программой ускоренного развития услуг в сфере информационно-коммуникационных технологий (ИКТ) на 2011–2015 гг., отраслевыми и региональными программами информатизации.

Стратегической целью дальнейшего развития информатизации в Республике Беларусь является совершенствование условий, содействующих трансформации сфер человеческой деятельности под воздействием ИКТ, включая формирование цифровой экономики, развитие информационного общества и совершенствование электронного правительства Республики Беларусь.

В 2016–2022 гг. в вопросе развития информатизации Республика Беларусь ставит своими задачами:

- развитие эффективной и прозрачной системы государственного управления посредством внедрения передовых ИКТ во все сферы человеческой жизнедеятельности;
- совершенствование системы управления и правового регулирования процессами информатизации;

- дальнейшее совершенствование национальной информационно-коммуникационной инфраструктуры;
- обеспечение прозрачности и удобства коммуникаций между гражданами, бизнесом и государством путем повсеместного перевода данных коммуникаций в электронную форму;
- создание и внедрение государственной системы идентификации субъектов информационных отношений;
- дальнейшее формирование единого информационного пространства для оказания электронных услуг на основе интеграции информационных систем;
- создание условий для использования электронных услуг, стимулирующих их востребованность;
- увеличение объема производства и безопасного потребления высокотехнологичных и наукоемких ИКТ товаров и услуг;
- модернизация традиционных отраслей промышленности на основе внедрения мировых стандартов качества, технологий цифрового маркетинга и производства;
- обеспечение непрерывности, безотказности, безопасности информационных потоков.

*При решении данных задач следует руководствоваться следующими принципами:*

- определяющая роль государства в формировании политики информатизации и стимулировании применения ИКТ во всех сферах жизнедеятельности современного общества;
- открытость государственного управления, свобода доступа к информации и знаниям;
- развитие государственно-частного партнерства в сфере информатизации;
- вовлечение в процесс информатизации всех слоев и социальных групп населения, ликвидация цифрового неравенства;
- обеспечение нового уровня цифровой грамотности населения;
- содействие обеспечению национального суверенитета в информационной сфере и национальной безопасности;
- гармонизация направлений информатизации с государствами-партнерами по ЕАЭС, другими государствами Европы и Азии.

Критерием успешности реализации данных направлений является нахождение Республики Беларусь к 2022 г. в рейтинге по индексу готовности к электронному правительству среди развитых стран мира (в соответствии с классификацией ООН).

## 1.2. Понятия информации и информационной системы. Информационные процессы. Информационные технологии

Термин «информация» происходит от латинского *information* – разъяснение, изложение, осведомление о каком-либо факте или событии. В последнее время информацию чаще относят к разделу общенаучных понятий, так как она выходит за рамки какой-то одной отрасли знаний и используется многими науками. Под **информацией** понимают совокупность фактов, явлений, событий, представляющих интерес и подлежащих регистрации и обработке.

В настоящее время имеется несколько взглядов на понятие информации.

С точки зрения философии информация – это категория, представляющая собой отражение объективного мира, его причинных и следственных связей.

Известен также технологический (прикладной) подход к понятию информации. В этом случае при любой обработке сведения на входе процедуры обработки не являются еще информацией, а играют роль информационного «сырья». Сведения, получаемые на выходе процедуры (при условии, что в результате обработки достигается поставленная цель) – это и есть информация, т. е. «готовая продукция». Сущность обработки состоит в том, что из «сырого информационного ресурса» производится извлечение нужных получателю сведений – информации.

В теории информации под термином «информация» понимается такое сообщение, которое содержит факты, неизвестные ранее потребителю и дополняющие его представление об изучаемом и анализируемом объекте (процессе, явлении). Иначе говоря, по К. Э. Шеннону, **информация** – это снятая неопределенность, т. е. с точки зрения теории информации информацией могут быть лишь те сведения, которые позволяют устранить меру неопределенности в системе. И лишь получатель этих сведений может установить, представляют ли они собой информацию.

В теории информации термин «информация» идет рядом с таким понятием как «данные». Под данными понимают сведения о состоянии любого объекта. **Данные** – это информация, представленная в виде, удобном для передачи, интерпретации и обработки. А обработка данных – это некоторая систематизированная последовательность операций, приводящая данные к виду, удобному для получения из них ин-

формации. Кроме того, информация из данных получается только в результате воздействия на данные каких-либо методов, т. е. имеет место выражение:

**Информация = Данные + Методы.**

В результате одни и те же данные при обработке различными методами могут привести к различной информации.

Следует также отметить, что нет однозначной связи между формой данных и формой получаемой из них информации, т. е. данные могут быть, например звуковые (или речевые), а информацию они могут дать не только звуковую, но и текстовую (если ее записать словами) или графическую (если озвученные образы нарисовать). Таким образом, **информацию** можно рассматривать как содержательную часть данных, интерпретированных человеком.

Следующим основным (после информации и данных) понятием, на котором базируется информатика как наука, являются знания. **Знаниями** называют проверенный практикой результат познания действительности, ее верное отражение в сознании человека. Научное знание заключается в понимании действительности (от прошлого к настоящему и будущему), в достоверном обобщении фактов, в выявлении закономерностей и т. п. В системах искусственного интеллекта, которые в настоящее время занимают лидирующее положение среди всех компьютерных информационных систем, знания связывают с понятием логического вывода. Поэтому знания можно интерпретировать как информацию, на основе которой реализуется процесс логического вывода.

Существует три подхода к измерению информации:

I подход – неизмеряемость информации в быту (информация как новизна).

II подход – технический, или объемный (информация как сообщения в форме знаков или сигналов, хранимые, перерабатываемые и обрабатываемые с помощью технических устройств).

В вычислительной технике применяются две стандартные единицы измерения информации: **бит** и **байт**. Поскольку компьютер предназначен для обработки больших объемов информации, то используют производные единицы – *килобайт* (Кб), *мегабайт* (Мб), *гигабайт* (Гб). Обычно приставка «кило» означает тысячу, а приставка «мега» – миллион. Но в вычислительной технике осуществляется привязка к принятой двоичной системе кодирования.

В силу этого один килобайт равен не 1000 байтов, а  $2^{10} = 1024$  байта.



Аналогично  $1 \text{ Мб} = 2^{10} \text{ Кб} = 1024 \text{ Кб} = 2^{20} \text{ байтов} = 1\,048\,576 \text{ байтов}$ .

$1 \text{ Гб} = 2^{10} \text{ Мб} = 1024 \text{ Мб} = 2^{20} \text{ Кб} = 2^{30} \text{ байтов} = 1\,073\,741\,824 \text{ байта}$ .

III подход – вероятностный. Измерение информации в теории информации (информация как снятая неопределенность).

Получение информации (ее увеличение) означает увеличение знания, что, в свою очередь, означает уменьшение незнания или информационной неопределенности.

За единицу количества информации принимают выбор одного из двух равновероятных сообщений («да» или «нет», «1» или «0»). Она также названа *битом*.

Информация обладает динамическим характером. Она существует только в момент взаимодействия данных и методов. Все остальное время она пребывает в состоянии данных. Таким образом, информация существует только в момент протекания информационного процесса.

Одни и те же данные могут в момент потребления поставлять разную информацию в зависимости от степени адекватности взаимодействующих с ними методов. Например, для учащегося, не владеющего каким-либо языком, текст, написанный на этом языке, дает только ту информацию, которую можно получить методом наблюдения (количество символов, наличие незнакомых символов и т. д.). Использование же более адекватных методов даст иную информацию.

Данные являются объективными, поскольку это результат регистрации объективно существовавших сигналов, вызванных изменениями в материальных телах или полях. В то же время методы являются субъективными. В основе искусственных методов лежат алгоритмы (упорядоченные последовательности команд), составленные и подготовленные людьми (субъектами). В основе естественных методов лежат биологические свойства субъектов информационного процесса. Таким образом, «информация возникает и существует в момент диалектического взаимодействия объективных данных и субъективных методов».

Характерной особенностью информации, отличающей ее от других объектов природы и общества, является то, что на свойства информации влияют свойства данных, составляющих ее содержательную часть, и свойства методов, взаимодействующих с данными в ходе информационного процесса. Информация обладает множеством свойств. Каждая научная дисциплина рассматривает те свойства, которые ей наиболее важны. С точки зрения информатики и педагогики наиболее важ-

gufbij\_ ^klZ\eyxiky ke\_ ^mxsb\_ k\hckl\Z h[t\_dlb\ghklv iheghlZ  
 ^hklh\ \_jghklvZlZ^hklmighklv b ZdlmZevghklv bgnhjZpbb  
 H[t\_dlb\ghklv km[t\_dlb\ghklv hgnhjZpbb lhgyld\_ h[t\_  
 lb\ghklb yley\_iky hlgkbl\_evguf lZd dZd f\_lh^u yleyxiky km[t\_  
 lb\gufb ;he\_ h[t\_dlb\ghc ijbgyih kqblZlv hm bgnhjZpbx \ dhl  
 jmx f\_lh^u \ghkyl km[t\_dlb\guc we\_f\_gl  
 lheghlZgnhjZpbb lheghlZ bgnhjZpbb \h fgh]hf oZjZdl\_j  
 am\_l dZq\_kl\h bgnhjZpbb b hij\_ ^ey\_l ^hklZlhqghklv ^Zgguo ^ey  
 ijbgylby j\_r\_gbc beeb ^ey kha^Zgby gh\uo x^Zgguo gZ hkgh\ \_bf\_  
 sboky Q\_f iheg\_ ^Zgg\_ l\_f rbj\_ dhZlZa\_hgf\_lh^h\  
 gh bkihevah\Zlv  
 >hklh\ \_jghklv hgnhjZpbb >Zgg\_ \hagbdZxb \ fhf\_gl j\_  
 kljZpbb kb]gZeh\ gh g\_ \k\_ kb]gZeu y\eyx]l^kZ ©ihe\_agufb^a  
 ijbkmkl\m\_l-lZdZm]h\ \_gv ihklhjhggb kb]gZeh\ \ j\_amevlZl\_  
 q\_]h ihe\_agu\_ ^Zgg\_ khjh\h^eZxlgkf mjh\g\_f ©b  
 nhjfZpbhggh]h rmfZ^a ?keb ihe\_aguc kb]gZe aZj\_]bklbjh\Zg [he\_  
 q\_ldh q\_f ihklhjhggb\_ kb]gZeu ^hklh\ \_jghklv bgnhjZpbb fh`\_l  
 [ulv [he\_ \ukhdhc  
 :^\_d\Zlgh\ bgnhjZpbb kl\_i\_gv khhl\ \_lkl\by j\_Zevghfm  
 h[t\_dlb\ghfm khklvZgbb^Z^\_d\ZlgZy bgnhjZpby fh`\_l h[j  
 ah\l\Zlvky ijb kha^Zgbb gh\hc bgnhjZpbb gZ hkgh\ \_g\_iheguo beeb  
 g\_ ^hklh\ \_jguo ^Zgguo H^gZdh b ihegu\_ b ^hklh\ \_jgu\_ ^Zgg\_ f  
 ]ml ijb\h^blv d kha^Zgbx g\_Z^\_d\Zlghc bgnhjZpbb \ kemqZ\_ ijb  
 g\_gby d gbf g\_Z^\_d\Zlguo f\_l  
 >hklmighklv bgnhjZpbb\hafh`ghklb ihemqblv lm beeb  
 bgmx bgnhjZpbx GZ kl\_i\_gv ^hklmighklb ^bgnhjZpbb \ebyxl h  
 gh\j\_f\_ggh dZd ^hklmighklv ^Zgguo lZd b ^hklmighklv Z^\_d\Zlguo  
 f\_lh^h\ ^ey bo bgl\_jij\_lZpbb Hlkmkl\l\_b\_ ^hklmiZ d ^Zgguf beeb h  
 kmklb\_ Z^\_d\Zlguo f\_lh^h\ h[jZ[hldb ^Zgguo ijb\h^yl d h^bgZd  
 \hfm j\_amevlZlm bgnhjZpby hdZau\Z\_lky g\_ ^hklmighc  
 :dlmZevgh\ bgnhjZpbb kl\_i\_gv khhl\ \_lklZby bgnhj  
 pbb l\_dms\_fm fhf\_glm \j\_f\_gb lhkdhevdm bgnhjZpbhggu\_ ijhp\_  
 ku jZklygmlu \h \h^hklh\ \_jgZy b Z^\_d\ZlgZy gh mklZj\_\rZy  
 bgnhjZpby fh`\_l ijb\h^blv d hrb[hqguf j\_r\_gbyf  
 J\_ij\_a\_glZlb\ghklv hgnhjZpbb klyaZgZ k ijZ\bevgghklvx \_ h  
 [hjZ b nhjfbjh\Zgby \ p\_eyo Z^\_d\Zlgh]h hljZ`\_gby k\hckl\ h[t\_dlZ  
 <Z`gh\_ agZq\_gb\_ a^\_kv bf\_xl  
 xijZ\bevgghklv dhgp\_ipbb gZ [Za\_ dhlh\hc knhjfmebjh\Zgh b  
 oh^gh\_ ihgylb\_

- обоснованность отбора существенных признаков и связей отображаемого явления.

Нарушение репрезентативности информации нередко приводит к существенным ее погрешностям.

**Информационная система** – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели. Информационная система – это инструментарий информационных технологий. Информационная система немыслима без персонала, взаимодействующего с компьютером и телекоммуникациями.

По виду поддерживаемых информационных технологий, по классам решаемых задач и по областям применения можно выделить следующие информационные системы:

- системы обработки данных;
- системы, поддерживающие банки данных (фактографические базы данных);
- системы документографические, поддерживающие полнотекстовые документальные архивы;
- издательские системы;
- системы информационного обслуживания (системы научно-технической информации, информационно-поисковые системы);
- геоинформационные системы;
- системы автоматизированного проектирования;
- вычислительные системы;
- системы диагностики.

**Свойства информационных систем:**

- любая информационная система может быть подвергнута анализу, построена и управляема на основе общих принципов построения систем;
- информационная система является динамичной и развивающейся;
- при построении информационных систем необходимо использовать системный подход;
- выходной продукцией информационных систем является информация, на основе которой принимаются решения;
- информационную систему следует воспринимать как человеко-компьютерную систему обработки информации.

Общую структуру информационных систем можно рассматривать как совокупность подсистем независимо от сферы применения. Структура любой информационной системы может быть представлена совокупностью обеспечивающих подсистем: информационной, технической, математической, программной, организационной, правовой.

Также следует отметить, что информационные системы могут быть «ручными», автоматическими и автоматизированными. «Ручные» информационные системы характеризуются отсутствием современных технических средств переработки информации и выполнением всех операций человеком. Автоматические информационные системы выполняют все операции по переработке информации без участия человека. Автоматизированные информационные системы предполагают участие в процессе обработки информации и человека, и технических средств, причем основным средством является компьютер.

**Информационная технология (ИТ)** – процесс, использующий совокупность средств и методов сбора, обработки и передачи данных (первичной информации) для получения информации нового качества о состоянии объекта, процесса или явления (информационного продукта).

Цель ИТ – производство информации для ее анализа человеком и принятия на его основе решения по выполнению какого-либо действия. Она является наиболее важной составляющей процесса использования информационных ресурсов общества. В информационном обществе акцент внимания и значимости смещается с традиционных видов ресурсов (материальных, природных, трудовых, финансовых и энергетических) на информационный ресурс, который, хотя всегда существовал, не рассматривался ни как экономическая, ни как иная категория.

**Информационные ресурсы** – это знания, подготовленные людьми для социального использования в обществе и зафиксированные на материальном носителе.

К настоящему времени ИТ прошла несколько эволюционных этапов, смена которых определялась главным образом развитием научно-технического прогресса, появлением новых технических средств переработки информации.

В современном обществе основным техническим средством технологии переработки информации служит персональный компьютер, который существенно повлиял как на концепцию построения и ис-

пользования технологических процессов, так и на качество результатной информации. Внедрение персонального компьютера в информационную сферу и применение телекоммуникационных средств связи определили новый этап развития информационной технологии и, как следствие, изменение ее названия за счет присоединения одного из синонимов: компьютерная, новая или современная.

Определение компьютерная подчеркивает, что основным техническим средством ее реализации является компьютер, новая – указывает на новаторский, а не эволюционный характер этой технологии. Ее внедрение является новаторским актом в том смысле, что она существенно изменяет содержание различных видов деятельности в организациях. В понятие новой информационной технологии (НИТ) включены также коммуникационные технологии, которые обеспечивают передачу информации разными средствами, а именно – телефон, телеграф, телекоммуникации, факс и др.

**Инструментарий информационной технологии** – один или несколько взаимосвязанных программных продуктов для определенного типа компьютера, технология работы в котором позволяет достичь поставленную пользователем цель. В качестве инструментария можно использовать следующие распространенные виды программных продуктов для персонального компьютера: текстовые процессоры, электронные таблицы, системы управления базами данных, информационные системы функционального назначения (бухгалтерские, маркетинговые и пр.), экспертные системы и т. д.

Существует несколько точек зрения на развитие информационных технологий с использованием компьютеров, которые определяются различными признаками деления:

- вид задач и процессов обработки информации;
- проблемы, стоящие на пути информатизации общества;
- преимущество, которое приносит компьютерная технология;
- виды инструментария технологии.

В соответствии с последним признаком выделяют следующие этапы развития информационных технологий (табл. 1.1).

Информационные технологии могут обслуживать различные предметные области:

- бухгалтерский учет;
- управление персоналом;
- производственный менеджмент и др.

Таблица 1.1. Этапы развития информационных технологий

| Этап                                    | Вид ИТ               | Инструментарий                                                                                                                                                                                            | Основная цель технологии                                                                                         |
|-----------------------------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| 1-й этап<br>(до второй половины XIX в.) | Ручная               | Перо, чернильница, книга.<br>Ручной способ коммуникаций (переправка через почту писем, пакетов, депеш)                                                                                                    | Представление информации в нужной форме                                                                          |
| 2-й этап<br>(с конца XIX в.)            | Механическая         | Пишущая машинка, телефон, диктофон.<br>Оснащенная более совершенными средствами доставки почта                                                                                                            | Представление информации в нужной форме более удобными средствами                                                |
| 3-й этап<br>(40–60-е гг. XX в.)         | Электрическая        | Большие ЭВМ и соответствующее программное обеспечение, электрические пишущие машинки, ксероксы, портативные диктофоны                                                                                     | Акцент перемещается с формы представления информации на формирование ее содержания                               |
| 4-й этап<br>(с начала 70-х гг. XX в.)   | Электронная          | Большие ЭВМ и создаваемые на их базе автоматизированные системы управления (АСУ) и информационно-поисковые системы (ИПС), оснащенные широким спектром базовых и специализированных программных комплексов | Формирование содержательной стороны информации для управленческой среды различных сфер общественной жизни        |
| 5-й этап<br>(с середины 80-х гг. XX в.) | Компьютерная (новая) | Персональный компьютер с широким спектром стандартных программных продуктов разного назначения.<br>Телекоммуникации, глобальные и локальные компьютерные сети                                             | Персонализация АСУ, которая проявляется в создании систем поддержки принятия решений определенными специалистами |

Классификация ИТ по типу обрабатываемой информации приведена на рис. 1.1, а по другим классификационным признакам – на рис. 1.2.

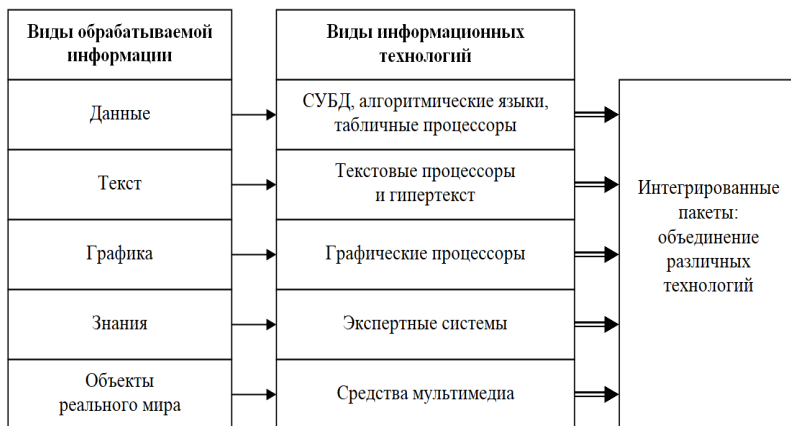


Рис. 1.1. Классификация компьютерных информационных технологий в зависимости от типа обрабатываемой информации



Рис. 1.2. Классификация компьютерных информационных технологий по способу построения сети ЭВМ, типу пользовательского интерфейса и степени автоматизации задач управления

## Контрольные вопросы

1. В чем заключается процесс информатизации?
2. Что составляет научно-техническую базу процесса информатизации общества?
3. Какое общество называют информационным?
4. Что такое информация?
5. Что понимается под информацией с точки зрения философии?
6. Что понимают под информацией с технической точки зрения?
7. Что такое данные?
8. Назовите и дайте характеристику свойствам информации.
9. Как влияет информация на уровень жизни общества?
10. Что такое информационная система? Приведите примеры.
11. Что такое информационный ресурс?
12. Назовите классические информационные процессы.
13. Что такое информационные технологии?
14. Какие этапы выделяют в развитии информационных технологий?
15. Назовите классификационные признаки информационных технологий.

## ЛИТЕРАТУРА

1. Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы: учебник / В. А. Гвоздева. – М.: ИД ФОРУМ; НИЦ ИНФРА-М, 2013. – 544 с.
2. Информационные системы и технологии в экономике и управлении: учебник для бакалавров / под ред. проф. В. В. Трофимова. – 3-е изд., перераб. и доп. – М.: Юрайт, 2012. – 521 с.
3. Информационные технологии: учебник / О. Л. Голицына [и др.]. – М.: Форум; ИНФРА-М, 2013. – 607 с.
4. Гришин, В. Н. Информационные технологии в профессиональной деятельности: учебник / В. Н. Гришин, Е. Е. Панфилова. – М.: ИД ФОРУМ; НИЦ ИНФРА-М, 2013. – 416 с.
5. Исаев, Г. Н. Информационные технологии: учеб. пособие / Г. Н. Исаев. – М.: Омега-Л, 2013. – 464 с.
6. Хлебников, А. А. Информационные технологии: учебник / А. А. Хлебников. – М.: КноРус, 2014. – 472 с.



## Тема 2. ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

- 2.1. Классификация программного обеспечения.
- 2.2. Системы обработки текстовых документов.
- 2.3. Табличные процессоры.
- 2.4. Системы подготовки презентаций.

### 2.1. Классификация программного обеспечения

В основу работы компьютеров положен программный принцип управления, состоящий в том, что компьютер выполняет действия по заранее заданной программе. Этот принцип обеспечивает универсальность использования компьютера: в определенный момент времени решается задача соответственно выбранной программе. После ее завершения в память загружается другая программа и т. д.

**Программа** – это запись алгоритма решения задачи в виде последовательности команд или операторов языком, который понимает компьютер. Конечной целью любой компьютерной программы является управление аппаратными средствами.

Под **программным обеспечением** (Software) понимается совокупность программ, выполняемых вычислительной системой. К программному обеспечению относится также вся область деятельности по проектированию и разработке ПО.

Программное обеспечение – неотъемлемая часть компьютерной системы. Оно является логическим продолжением технических средств. Сфера применения конкретного компьютера определяется используемым ПО. Сам по себе компьютер не обладает знаниями ни в одной области применения. Все эти знания сосредоточены в выполняемых на компьютерах программах.

*Классификация программного обеспечения.*

В первом приближении все программы, работающие на компьютере, можно условно разделить на три категории:

- **системные;**
- **прикладные;**
- **инструментальные.**

### Системное программное обеспечение

Системное программное обеспечение – это совокупность программ и программных комплексов, предназначенных для обеспечения работы компьютера и вычислительных сетей.

В составе системного ПО можно выделить две составляющие:

**Базовое программное обеспечение** – минимальный набор программных средств, обеспечивающих работу компьютера.

**Сервисное программное обеспечение** – программы и программные комплексы, которые расширяют возможности базового программного обеспечения и организуют более удобную среду работы пользователя. Оно может быть приобретено дополнительно.

Основу базового системного программного обеспечения составляет операционная система.

**Операционная система (ОС)** – совокупность программных средств, обеспечивающая управление аппаратной частью компьютера и прикладными программами, а также их взаимодействием между собой и пользователем.

Операционная система образует автономную среду, не связанную ни с одним из языков программирования. Любая прикладная программа связана с ОС и может эксплуатироваться только на компьютерах, где имеется аналогичная системная среда. Программа, созданная в среде одной ОС, не функционирует в среде другой ОС, если в ней не обеспечена возможность конвертации (преобразования) программ.

Основная функция всех операционных систем – посредническая. Она заключается в обеспечении нескольких видов взаимодействия – интерфейса:

- интерфейс между пользователем и программно-аппаратными средствами (интерфейс пользователя);
- интерфейс между программным и аппаратным обеспечением (аппаратно-программный интерфейс);
- интерфейс между разными видами программного обеспечения (программный интерфейс).

ОС загружается автоматически при включении компьютера. Каждая программа пользуется услугами ОС, а потому может работать только под управлением той ОС, которая обеспечивает для нее эти услуги. Таким образом, выбор ОС очень важен, так как он определяет, с какими программами вы сможете работать на своем компьютере. От выбора ОС зависят также производительность вашей работы, степень защиты ваших данных, необходимые аппаратные средства и т. д.

Сервисное системное программное обеспечение является расширением базового и представляет собой набор специальных, сервисных, дополнительно устанавливаемых программ, которые можно классифицировать по функциональному признаку следующим образом:

- программы диагностики работоспособности компьютера;
  - антивирусные программы, обеспечивающие защиту компьютера, обнаружение и восстановление зараженных файлов;
  - программы обслуживания дисков, обеспечивающие проверку качества поверхности магнитного диска, контроль сохранности файловой системы на логическом и физическом уровнях, сжатие дисков, резервное копирование данных на внешние носители и др;
  - программы архивирования данных, которые обеспечивают процесс сжатия информации с целью уменьшения объема памяти для ее хранения;
  - программы обслуживания сети.
- Эти программы часто называют утилитами.

### Прикладное программное обеспечение

Прикладное программное обеспечение предназначено для решения задач пользователя. В его состав входят **пакеты прикладных программ** (ППП) различного назначения и **прикладные программы пользователей**.

**Прикладная программа пользователя** – это любая программа, способствующая решению какой-либо задачи в пределах данной проблемной области. Такие программы могут использоваться либо автономно, либо в составе программных комплексов.

**Пакеты прикладных программ** (ППП) – это специальным образом организованные программные комплексы, рассчитанные на применение в определенной проблемной области и дополненные соответствующей технической документацией. Классификация пакетов прикладных программ (ППП) приведена на рис. 2.1.



Рис. 2.1. Классификация прикладного программного обеспечения

Различают следующие типы ППП:

**ППП общего назначения** – универсальные программные продукты, предназначенные для автоматизации широкого класса задач пользователя. К ним относятся: *текстовые процессоры* (MS Word, Word Perfect, Лексикон); *табличные процессоры* (MS Excel, Lotus 1-2-3, Quattro Pro); *системы динамических презентаций* (MS Power Point, Freelance Graphics, Harward Graphics); *системы управления базами данных* (Microsoft Access, Oracle, Microsoft SQL Server, Informix); *графические редакторы* (Corel Draw, Adobe Photoshop); *издательские системы* (Page Maker, Venture Publisher); *системы автоматизации проектирования* (AllFusion Process Modeler r7, ERWin Data Modeler r7); *электронные словари и системы перевода* (Prompt, Сократ, Лингво, Контекст); *системы распознавания текста* (Fine Reader, Cunei Form).

Системы общего назначения часто интегрируются в многокомпонентные пакеты для автоматизации офисной деятельности – **офисные пакеты** – Microsoft Office, OpenOffice и др. Различные редакции офисного пакета могут включать разные компоненты, но обычно в его состав входит минимальный набор программных продуктов, отвечающий реализации наиболее распространенных функций в деятельности офиса: текстовый и табличный процессоры, графический редактор, программа-органайзер, программы-переводчики, средства проверки орфографии, распознавание текста, браузеры, средства создания web-страниц, средства электронной почты.

**Методо-ориентированные ППП**, в их основе лежит реализация определенных методов решения задач. К ним относятся системы математической обработки данных (Mathematica, MathCad, Maple), системы статистической обработки данных (Statistica, Stat, SPSS).

Основные функциональные возможности пакетов математической обработки данных: естественная запись математических выражений; построение разнообразных диаграмм и графиков; наличие встроенных вычислительных средств; реализация методов решения разнообразных математических задач (матричная алгебра, решение дифференциальных и интегральных уравнений, решение систем линейных и нелинейных уравнений и пр.); работа с размерными величинами; наличие средств аналитического решения задач.

Основные функциональные возможности пакетов по статистической обработке данных: реализация основных методов статистической обработки данных; построение диаграмм и графиков; реализация линейных и нелинейных методов анализа; реализация многомерных раз-

ведочных технологий анализа; расширение функциональных возможностей за счет установки дополнительных модулей (нейронные сети, прогнозные модели и пр.).

**Проблемно-ориентированные ППП** предназначены для решения определенных задач в конкретной предметной области. Например, информационно-правовые системы (ЮрЭксперт, ЮрИнформ); пакеты бухгалтерского учета и контроля (1С: Бухгалтерия, Анжелика, БЭСТ, Парус); в области маркетинга (Касатка, Marketing Expert, Олимп: Маркетинг); автоматизации банковской деятельности (СТБанк.ИТ, SC-Bank NT, Ва-Банк), интегрированные информационные системы (Галактика, 1С: Предприятие, SAP R3, БОСС-Корпорация, БЭСТ-ПРО).

**ППП автоматизированного проектирования** предназначены для поддержки работы конструкторов и технологов, связанных с разработкой чертежей, схем, графическим моделированием и конструированием. Отличительной особенностью этого класса ППП являются высокие требования к аппаратному обеспечению, наличие библиотек встроенных функций, объектов, интерфейсов с графическими системами и БД (AutoCAD).

**Программные средства мультимедиа.** Основное значение данных программных средств – создание и использование аудио- и видеoinформации для расширения информационного пространства пользователя (различные БД компьютерных произведений искусства, библиотеки звуковых записей и т. д.).

**Настольные издательские системы.** Данный класс программ включает: PageMaker, CorelDraw, PhotoShop for Windows и др., обеспечивающие информационную технологию компьютерной издательской деятельности:

- форматирование и редактирование текстов;
- автоматическую разбивку текста на страницы;
- компьютерную верстку печатной страницы;
- монтирование графики;
- подготовку иллюстраций и т. п.

**Интеллектуальные системы.** Под интеллектуальными системами (ИС) можно понимать автоматические и автоматизированные системы с элементами искусственного интеллекта (ИИ).

*Основными направлениями ИИ являются:*

- представление знаний и разработка систем, основанных на знаниях;
- игры и творчество (шахматы, шашки, го);

- разработка естественно-языковых интерфейсов и машинный перевод;
- распознавание образов (каждому объекту ставится в соответствие матрица признаков, по которой происходит его распознавание);
- новые архитектуры компьютеров (нейрокомпьютеры);
- интеллектуальные роботы;
- специальное ПО (языки ЛИСП, ПРОЛОГ);
- обучение и самообучение (включает модели, методы и алгоритмы, ориентированные на автоматическое накопление знаний на основе анализа и обобщения данных).

*Экспертные системы* – программные комплексы, содержащие знания экспертов в конкретных предметных областях и тиражирующие этот опыт для консультаций менее квалифицированных пользователей.

*Системы искусственного интеллекта:*

- программы-оболочки для создания экспертных систем путем наполнения баз знаний и правил логического вывода;
- готовые экспертные системы для принятия решений в рамках определенных предметных областей;
- системы анализа и распознавания речи, текста и т. п.

Примеры систем искусственного интеллекта: FIDE, MYSIN, Guru и др.

## **Инструментальное программное обеспечение**

Инструментальное программное обеспечение (системы программирования) предназначено для разработки новых программ. Включает специальные программы, которые являются инструментальными средствами для разработки программного обеспечения. Поддерживает все технологические этапы процесса проектирования, программирования, отладки и тестирования создаваемых программ. Пользователями являются системные и прикладные программисты.

Инструментарий технологии программирования – это программные продукты, предназначенные для поддержки технологии программирования (рис.2.2).

***Средства для создания приложений*** – совокупность языков и систем программирования, инструментальные среды пользователя, а также различные программные компоненты для отладки и поддержки создаваемых программ.

**Язык программирования** – это формализованный язык для описания алгоритма решения задач на компьютере. Языки программирования можно условно разделить на следующие классы:

- машинные языки – это языки, воспринимаемые аппаратной частью компьютера (машинные коды);
- машинно-ориентированные языки, отражающие структуру конкретного типа компьютера (ассемблер);
- процедурно-ориентированные языки – это языки, в которых имеется возможность описания программы как совокупности процедур, или подпрограмм (Си, Паскаль и др.);
- проблемно-ориентированные языки, предназначенные для решения задач определенного класса (ЛИСП, ПРОЛОГ).

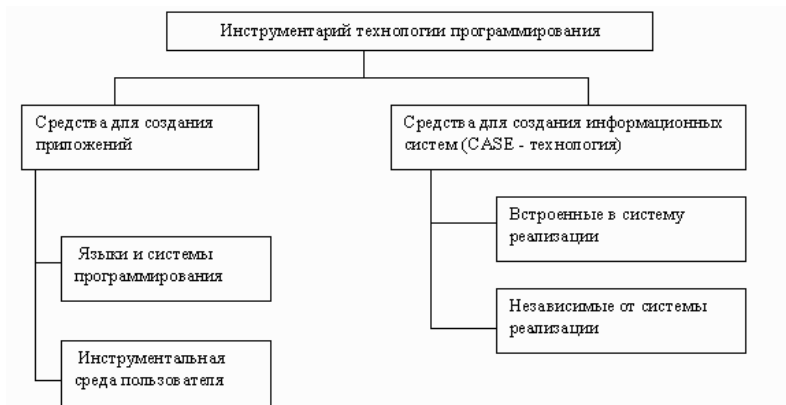


Рис. 2.2. Инструментарий технологии программирования

**Системы программирования** включают:

- интерпретаторы (компилятор, транслятор) – компилятор транслирует всю программу без ее выполнения, трансляторы выполняют пооперационную обработку и выполнение программы;
- интегрированную среду разработки программ (не всегда);
- отладчик – специальные программы, предназначенные для трассировки и анализа выполнения других программ. Трассировка – это обеспечение выполнения в пооператорном варианте;
- средства оптимизации кода программ;
- набор библиотек;
- редактор связей;

- сервисные средства (утилиты) (для работы с библиотеками, текстовыми и двоичными файлами);
- справочные системы;
- систему поддержки и управления продуктами программного комплекса.

**Инструментальная среда пользователя** – это специальные средства, встроенные в пакеты прикладных программ, такие, как:

- библиотека функций, процедур, объектов и методов обработки;
- макрокоманды;
- клавишные макросы;
- языковые макросы;
- конструкторы экранных форм и объектов;
- генераторы приложений;
- языки запросов высокого уровня;
- конструкторы меню и др.

Интегрированные среды разработки программ объединяют набор средств для их комплексного применения на технологических этапах создания программы.

Примеры программных продуктов для создания приложений: Visual C++, Delphi, Visual Basic и т. д.

**Средства для создания информационных систем (CASE-технология)** – программный комплекс, автоматизирующий весь технологический процесс анализа, проектирования, разработки и сопровождения сложных программных систем.

Средства CASE-технологий делятся:

- **на встроенные в систему реализации** – все решения по проектированию и реализации привязки к выбранной СУБД;
- **независимые от системы реализации** – все решения по проектированию ориентированы на унификацию (определение) начальных этапов жизненного цикла программы и средств их документирования, обеспечивают большую гибкость в выборе средств реализации.

Все программное обеспечение по степени доступности можно разбить на следующие группы: коммерческое ПО; условно-бесплатное ПО; бесплатное ПО; ПО с открытыми исходными текстами; свободное ПО.

**Коммерческие программы** – это программное обеспечение, разработанное коммерческой организацией, которая хочет получать деньги за его использование. Большая часть коммерческого программного обеспечения является закрытой.



В группу условно-бесплатного ПО, называемого *shareware*, включены программы, для полноценного использования которых требуется тем или иным способом осуществить оплату. *Shareware* поставляется без разрешения сделать копию и установить ее, не платя за лицензию, даже для частных лиц, участвующих в некоммерческой деятельности. Чаще всего под видом *shareware*-программ выступают либо trial-версии, либо варианты коммерческих программ с урезанной функциональностью.

Бесплатное программное обеспечение (*freeware*) объединяет общая черта – за использование платить не нужно или не обязательно. Последняя категория (*donationware*) по сути своей гораздо больше соответствует *shareware* в его исконном значении: понравилась программа (или просто захотелось отблагодарить разработчика) – заплатил, не понравилась – не заплатил. Статус *freeware* обычно является временным, так как в случае успешности проекта бесплатный проект становится *shareware* или даже коммерческим.

**Свободное программное обеспечение** – программное обеспечение, пользователь которого свободен запускать с любой целью, изучать исходный код и адаптировать его к своим нуждам, распространять копии, модифицировать и распространять модифицированные версии.

Чтобы программное обеспечение считалось «свободным», его правообладатели должны дать пользователю все четыре вышеперечисленные свободы. Это достигается выпуском исходного кода программного обеспечения под одной из особого рода **лицензий**, называемых **свободной лицензией**. При этом автор программы сохраняет предоставленные ему законом авторские права на программу, однако лицензия явным образом разрешает свободное использование. Подавляющее большинство свободных программ является одновременно открытыми, и наоборот.

GNU General Public License (Универсальная общедоступная лицензия GNU или Открытое лицензионное соглашение GNU) – наиболее популярная лицензия на свободное программное обеспечение. Ее также называют GNU GPL или даже просто GPL. Цель GNU GPL – предоставить пользователю права копировать, модифицировать и распространять программы (что по умолчанию запрещено законом об авторских правах), а также гарантировать, что и пользователи всех производных программ получают вышеперечисленные права.

Перечень и форма документов, которая подтверждает правомерность использования каждой из компьютерных программ, как коммер-

ческих, так и программ свободного пользования, независимо от способа распространения компьютерной программы устанавливается автором или другим лицом, которое на законных основаниях владеет имущественным авторским правом на нее и, как правило, включает: лицензию или договор относительно распоряжения имущественными правами интеллектуальной собственности, в которых отмечаются разрешенные способы использования, в том числе и в случае коммерческого использования, документацию в печатном виде, носитель с экземпляром программы.

Распространение экземпляров компьютерных программ через сеть Интернет может проводиться на коммерческой или безвозмездной основе путем их размещения авторами (или другими лицами, которые имеют авторское право) на своих официальных сайтах. К незаконным способам распространения компьютерных программ через сеть Интернет относятся: размещение компьютерных программ лицами, которые не являются их авторами (или другими лицами, которые имеют авторское право); распространение контрафактных экземпляров программ через Интернет (через Интернет-магазины, доски сообщений и объявлений и т. п.).

Использование компьютерной программы (игры) без разрешения автора (или другого лица, которое имеет авторское право) является нарушением прав автора и может быть основанием для привлечения лица-нарушителя к гражданско-правовой, административной или криминальной ответственности.

## 2.2. Системы обработки текстовых документов

*Текстовый процессор* – это вид прикладного программного обеспечения, предназначенный для создания и обработки текстовых документов. Современный документ может содержать кроме текста и другие объекты (таблицы, диаграммы, рисунки и т. д.).

По типу обрабатываемого объекта текстовые редакторы делятся на четыре группы: простые текстовые редакторы, редакторы форматированных текстов, научные редакторы, издательские системы.

**Простые текстовые редакторы** используются для написания простого текста и относятся к самым нужным и используемым программам для создания заметок или небольших документов, небольших программ, web-страниц.

**Редакторы форматированных текстов** (текстовые процессоры) предназначены для работы со сложными документами, структура ко-

торых может содержать вложенные документы, страницы, абзацы и т. д. Документ может включать также таблицы, диаграммы, рисунки, фотографии, звук и видео. При этом таблицы, диаграммы и рисунки могут быть созданы как в самом редакторе, так и в других приложениях. Существует большое количество редакторов форматированных текстов различной сложности. Наиболее распространены в среде компьютерных пользователей редакторы **Microsoft Word**, **Word Perfect**, **WordStar**.

**Редакторы научных текстов** обеспечивают подготовку и редактирование текстов, содержащих большое количество математических формул, графиков, специальных символов и т. п. Распространенными и известными среди пользователей являются научные редакторы **ChiWriter**, **MathOr**, **MathWord**, **TEX**.

**Издательские системы** используются для подготовки больших и сложных документов, предназначенных для тиражирования, в том числе полиграфическими способами (книги, альбомы, журналы, газеты). К распространенным издательским системам относятся, например, программы **Publisher**, **PageMaker**.

**Текстовый процессор Microsoft Word** – приложение для подготовки, обработки, сохранения и печати документов различной сложности.

**Основные элементы документа.** Основными элементами текстового документа являются *символ*, *абзац* и *страница*.

**Символ** (знак) – минимальный элемент текста документа.

**Абзац** – фрагмент текста между двумя маркерами конца абзаца. Маркер абзаца – это служебный непечатаемый символ, обозначается знаком ¶. *Непечатаемые знаки* – служебные символы-маркеры, появляющиеся в рабочем окне документа при выполнении команд или нажатии некоторых клавиш клавиатуры, при печати документа не выводятся.

**Страница** – отдельный фрагмент документа, размер которого определяется параметрами страницы: размером бумаги, верхним, нижним, левым и правым полями.

Наиболее крупной единицей является собственно документ, где все составляющие его абзацы определенным образом структурированы, снабжены при необходимости заголовками, выстроена иерархия структурных разделов.

**Основные процедуры подготовки документов.** Процедурами подготовки документа являются *ввод текста*, *редактирование* документа и *форматирование*, *сохранение* и *печать документа*.

**Редактирование** документа – изменение его содержания с помощью перемещения, копирования или удаления и добавления фрагментов текста.

**Форматирование** документа – изменение внешнего вида документа с использованием различных шрифтов, эффектов отображения символов и псевдографики, междустрочных интервалов, отступов относительно границ текста и т. д.

**Шрифт** – набор символов с характерным рисунком и настраиваемыми параметрами, такими как размер, начертание и т. д. Каждый шрифт имеет специфический рисунок символов (с засечками или без засечек, с одинаковой или различной шириной букв) и название (Times New Roman, Arial, Courier, Verdana).

**Размер (кегель)** – высота символов в пунктах, равных  $1/72$  доле дюйма или около 0,35 мм, т. е. 1 пт = 0,35 мм.

**Подчеркивание** – различные варианты подчеркивания символов.

**Эффекты (видоизменение)** – верхний и нижний индексы, прописные и малые прописные, контур, скрытый, уплотненный, приподнятый, с тенью.

**Выравнивание** текстовых строк относительно границ – по левому и правому краю, по центру и по ширине.

**Псевдографика** – оформление текста с помощью рамок и цветового фона.

**Список** – последовательность абзацев, вначале каждого из которых размещены номера в нумерованном списке или маркеры в маркированном списке.

## **Интерфейс Microsoft Word 2007**

Основным средством для работы с содержимым документа в окне приложения Microsoft Word 2007 является **Лента** (рис. 2.3).

**Лента** – это область окна приложения, расположенная между строкой заголовка и окном редактирования документа, на которой размещены команды (пиктограммы), необходимые для работы с содержимым документа. Слева над **Лентой** размещена **Кнопка Office**, нажатие которой открывает доступ к меню с набором команд для управления файлами и **Панель быстрого доступа**, на которую пользователь может поместить команды, необходимые для работы.

Названия вкладок **Главная**, **Вставка**, **Разметка страницы**, **Ссылки** и т. д. на **Ленте** соответствуют этапам процесса создания докумен-

та. Набор команд, который отображается на **Ленте**, зависит от того, какая вкладка выбрана. Команды на **Ленте** организованы в группы по выполняемым действиям. Названия групп команд отображаются в нижней части **Ленты**. На **Ленте** активны только те кнопки (команды), которые могут быть применены к выделенному элементу документа.

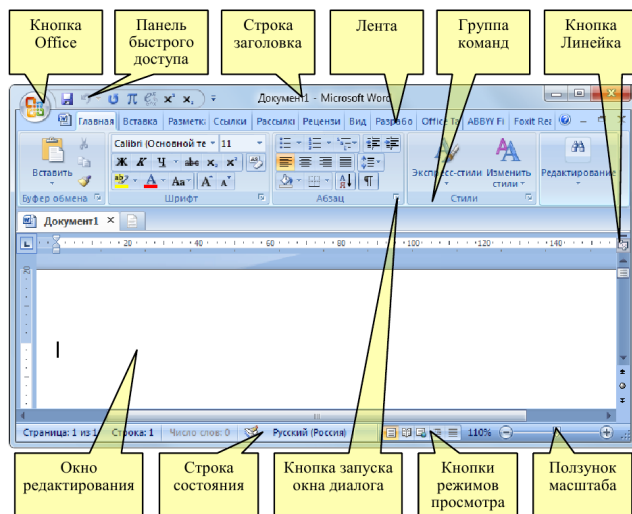


Рис. 2.3. Графический интерфейс Microsoft Word 2007

Для управления документом в целом применяются команды, собранные в меню **Файл**, которое открывается щелчком по **кнопке Office**. Меню состоит из двух панелей, на левой панели отображаются группы команд для работы с файлами, а на правой панели – список команд, содержащихся в выделенной группе команд. В нижней области окна расположена кнопка **Параметры Word** для доступа к настройке параметров приложения.

Первичные настройки текстового процессора и доступ к ним осуществляются следующими командами: Кнопка **Office** ➤ Кнопка **Параметры Word**, команды меню **Основные**, **Экран**, **Правописание**, **Сохранение**, **Дополнительно**.

Кроме **Ленты** и меню **Файл** для работы с документами в приложении Microsoft Word 2007 применяется **Панель быстрого доступа**. По умолчанию на этой панели расположены три пиктограммы: **Со**

**хранить, Отменить и Вернуть.** Панель быстрого доступа можно настраивать и помещать на нее требуемые пользователю инструменты, например команду **Автотекст, Правописание** и т. д.

**Обработка больших документов. Стилиевое форматирование документа.**

Под форматированием понимают оформление текста. Существуют два способа форматирования текста:

- *прямое форматирование* выделенных фрагментов;
- *стилевое форматирование*, с помощью которого можно оформить сразу несколько абзацев без их предварительного выделения.

**Стиль** – это сохраненный под уникальным именем набор форматов как единое целое для многократного использования. В момент применения стиля к тексту все инструкции форматирования выполняются одновременно. По умолчанию при вводе текста в новом документе используется стиль *Обычный (экспресс-стиль)*, в котором установлены основные параметры форматирования абзаца: шрифт – Calibri, Основной текст, выравнивание символов – По левому краю, междустрочный интервал – множитель 1,15 ин., интервал После абзаца – 10 пт., Запрет висячих строк.

Каждый встроенный шаблон в Word имеет набор стилей, которые «передаются» создаваемому документу. В последующем при работе с документом можно изменять параметры стилей, можно добавлять стили из других шаблонов и документов, можно создавать новые стили. Стили можно сохранять в документе или в шаблоне.

Существуют три основных типа стилей.

1. **Стиль абзаца** (обозначение – ¶) – определяет внешний вид абзаца, т. е. параметры шрифта, выравнивание текста, позиции табуляции, междустрочный интервал и границы. Используется для задания стилей заголовков, оглавлений, различных вырезок, примечаний, аннотаций и т. д.

2. **Стиль знака** (обозначение – а) – задает форматирование выделенного фрагмента текста внутри абзаца, определяя такие параметры текста, как шрифт и размер, а также полужирное и курсивное начертание. Используется для форматирования определений, формул, названий и т. д.

3. **Стиль таблицы** (обозначение – ☐) – задает вид границ, заливку, выравнивание текста и шрифты.

Абзацы, содержащие параметры форматирования абзацев и знака, обозначаются – а¶.

Помимо указанных, стили могут включать в себя и некоторые другие параметры.

Изменение какого-либо стиля вызывает его автоматическое обновление по всему документу.

В Word существует большое количество *стандартных* (т. е. *встроенных*) *стилей*: **Обычный**, **Заголовок 1..9**, **Строгий**, **Схема документа** и т. д. Некоторые из встроенных стилей находятся в коллекции экспресс-стилей (вкладка **Главная**/в группе **Стили** кнопка **Экспресс-стили**).

Коллекция экспресс-стилей используется для быстрого применения стилей к предварительно выделенным абзацам или фрагментам текста документа. При наведении курсора мыши на эскиз экспресс-стиля можно увидеть, как выделенный фрагмент текста или абзац, в котором установлен курсор, примет выбранное форматирование.

Стиль текущего абзаца отображается в списке стилей области задач **Стили**, которая открывается кнопкой группы **Стили** на вкладке **Главная**. Выбор какого-либо стиля из списка вызовет немедленное его применение к предварительно выделенному абзацу.

Стили в Word 2007 можно:

- *переименовывать*, т. е. изменять имя существующего стиля;
- *применять*, т. е. применять другие стили к абзацу или фрагменту текста (осуществлять наложение стилей);
- *изменять* существующие стили;
- *создавать новые* (т. е. пользовательские).

**Создание стиля.** Создать *собственный стиль* можно одним из следующих способов.

#### **1 способ:**

1) выбрать вкладку **Главная**/кнопку группы **Стили**/кнопку **Создать стиль** ();

2) в окне **Создание стиля** в поле **Имя** наберите имя нового стиля;

3) в списке **Стиль** выберите тип создаваемого стиля – стиль *абзаца* или *знака* (т. е. символа);

4) в списке **Основан на стиле**, выберите стиль, на котором должен базироваться создаваемый стиль;

5) нажмите кнопку **Формат** и установите параметры форматирования нового стиля, открывая окна **Шрифт**, **Абзац**, **Табуляция**, **Граница**, **Язык**, **Рамка**, **Нумерация**, **Сочетание клавиш** с помощью соответствующих команд открывшегося списка. Нажмите кнопку **ОК**, чтобы подтвердить установки;

6) если необходимо отображение этого стиля в галерее *Экспресс-стили*, то в окне **Создание стиля** установите флажок **Добавлять в список экспресс-стилей**. Для автоматического обновления стиля можно установить флажок **Обновлять автоматически**. Чтобы добавить создаваемый стиль в шаблон, на основе которого создан текущий документ, для того чтобы этот стиль был доступен во всех вновь создаваемых документах, выберите опцию **В новых документах, использующих этот шаблон**;

7) в заключение в окне **Создание стиля** нажмите кнопку **ОК**.

Созданный стиль вы найдете в окне **Стили** (вкладка **Главная**/кнопка группы **Стили**) и к предварительно выделенному фрагменту текста вы сможете его применить, выбрав имя созданного стиля в списке **Стили**.

## **2 способ:**

- 1) набрать абзац;
- 2) применить к нему форматирование;
- 3) выделить этот абзац в тексте;
- 4) выбрать вкладку **Главная**/кнопку **Экспресс-стили**/команду **Сохранить выделенный фрагмент как новый экспресс-стиль**;
- 5) в открывшемся окне **Создание стиля**, в поле **Имя** ввести новое имя стиля (для внесения изменений в создаваемый стиль можно воспользоваться кнопкой **Изменить**) и нажать кнопку **ОК**.

**Изменение стиля.** Для того чтобы изменить параметры стиля, выберите вкладку **Главная**/кнопку группы **Стили**. В списке **Стили** выделите имя изменяемого стиля и щелкните по кнопке со стрелкой справа, затем из раскрывшегося меню выберите команду **Изменить**.

**Удаление стиля.** Выберите вкладку **Главная**/кнопку группы **Стили**. В списке **Стили** выделите имя удаляемого стиля и щелкните по кнопке со стрелкой справа, затем из раскрывшегося меню выберите команду **Удалить**.

**Управление стилями.** При создании большого числа новых стилей возникает необходимость в их организации (т. е. управлении стилями). Для этого выберите вкладку **Главная**/кнопку группы **Стили**/кнопку



**Управление стилями** ( ). С помощью окна **Управление стилями** (кнопки **Импорт/экспорт**) можно:

- переименовывать стили;
- копировать стили из других документов и шаблонов;
- удалять стили (встроенные стили удалить нельзя).



**Работа с разделами документа.** Разделы – это части документа, которые могут быть отформатированы независимо друг от друга.

Разделы особенно полезны при создании документов, состоящих из отдельных глав, и верстке документов, имеющих сложное полиграфическое оформление. Разделы позволяют начать каждую главу с нечетной страницы и задать колонтитулы, нумерацию страниц для этой главы.

По умолчанию весь документ представляет собой один раздел. Для того чтобы увеличить количество разделов, необходимо вставить разрыв раздела. Так же, как и разрыв страницы, разрыв раздела отображается как двойная пунктирная линия со словами *«разрыв раздела»* в обычном режиме просмотра. В режиме просмотра разметки страницы разрыв раздела отображается, если включен режим отображения непечатаемых символов. Линии, отображающие разрыв раздела, никогда вообще не печатаются.

**Создание колонтитулов.** Колонтитулы содержат информацию, которая повторяется на каждой странице документа. В простейшем случае, например, верхний колонтитул может содержать название главы, а нижний – номер страницы. В более сложных случаях колонтитулы могут содержать логотип фирмы или другую графику, имя автора, дату сохранения или дату печати документа и вообще любую информацию, которая может понадобиться. Обычно верхний колонтитул располагается в верхнем поле, а нижний колонтитул – в нижнем поле. С колонтитулами можно делать все, что можно делать с обычным текстом.

**Работа с номерами страниц.** Большие документы легче читать и использовать, когда страницы пронумерованы. Word позволяет легко вставить номера страниц в верхний или нижний колонтитул. Поэтому при работе с номерами страниц используются те же приемы, что и для работы с колонтитулами.

**Фоновая разбивка на страницы.** По умолчанию Word автоматически производит разбивку на страницы в процессе работы с документом. Это называется фоновой разбивкой на страницы. В режиме просмотра структуры, главного документа и обычном режиме просмотра фоновую разбивку на страницы можно отменить. Однако в режиме просмотра разметки страницы, а также в режиме предварительного просмотра перед печатью этого сделать нельзя. Отмена фоновой разбивки на страницы немного ускоряет работу программы.

**Работа с колонками.** Гладкий текст (строчка за строчкой от одного поля страницы до другого) – иногда не самый лучший способ подачи материала. Чтобы удерживать интерес читателя и сделать материал более привлекательным, помогает расположение текста в несколько колонок на одной странице. Исследования показали, что текст, расположенный в колонках газетной ширины, гораздо быстрее читается. Так что колонки делают материал не только более привлекательным, но и более читабельным.

**Создание колонок.** При необходимости можно создавать колонки как равной ширины, так и нет. Можно также создавать по-разному оформленные колонки в разных разделах документа. Между колонками можно задать вертикальные линии.

Максимально можно создать до тринадцати колонок на странице книжной ориентации и до восемнадцати колонок на странице альбомной ориентации. Однако с точки зрения удобства для чтения слишком большое количество колонок также нежелательно. Имеет смысл создавать не более трех колонок на стандартной странице книжной ориентации и более пяти колонок на странице альбомной ориентации.

**Форматирование колонок.** Интервал – это пространство между колонками. Ширина колонки – это расстояние от ее левого края до правого. При форматировании колонок нужно иметь в виду следующее:

- если нужно отформатировать весь документ в несколько колонок, а документ имеет только один раздел, то точку вставки можно поместить где угодно;
- если нужно отформатировать лишь один раздел в несколько колонок, и документ уже разделен на разделы, то точку вставки необходимо поместить в форматлируемый раздел. Отформатировать несколько разделов сразу можно, если их предварительно выделить;
- если форматирование по колонкам нужно начать с определенного места в документе, а документ еще не разделен на разделы, то точку вставки необходимо поместить в то место, где должно начаться форматирование колонок. Форматирование по колонкам можно применить от точки вставки вперед до конца документа. В этом случае в точку вставки будет вставлен разрыв раздела;
- если нужно отформатировать по колонкам только часть текста, а документ еще не разделен на разделы, то необходимо предварительно выделить форматлируемую часть текста. Разрыв раздела при этом будет помещен до и после выделения.

Количество колонок, а также их ширина определяются исходя из следующих факторов: ширина бумаги, размеры полей, ширина колонок, интервалы между колонками.

Кнопка *Колонки* позволяет создавать колонки одинаковой ширины. Диалоговое окно *Колонки* предоставляет гораздо больше возможностей. На схеме *Образец* окна *Колонки* можно видеть схему размещения колонок и выбрать нужный тип.

Добавление вертикальных разделительных линий между колонками улучшает внешний вид текста. Разделительные линии совпадают по длине с самой длинной из колонок данного раздела. Линии отображаются только в режиме просмотра разметки страницы или в режиме предварительного просмотра перед печатью.

**Сноски.** *Сноски* – это один из приемов размещения в документе дополнительной информации. Сноски применяют для самых разнообразных целей, например, для указания источника при цитировании или для указания перевода, если в основной текст документа включен фрагмент на иностранном языке.

Вообще говоря, использование сносок не является обязательным, поскольку тех же целей можно достичь, заключая дополнительную информацию в скобки или отделяя ее от основного текста иным способом. Однако в больших документах, особенно научного содержания, сноски используются очень часто, поскольку они повышают удобочитаемость и компактность основного текста.

Вообще говоря, *сноска* – это дополнительный текст, связанный с определенным местом в основном тексте, которое отмечается специальным символом – *символом сноски*. Символ сноски форматируется особым образом, чтобы его можно было легко отличить от символов основного текста. Сам же текст сноски размещается в другом месте, но ему предшествует тот же самый символ сноски, чтобы было видно, к какому месту основного текста относится сноска

Word поддерживает два вида сносок: **обычные сноски и концевые сноски.**

**Обычная** сноска<sup>1</sup> размещается внизу той страницы, на которой находится символ сноски<sup>2</sup>, и отделяется от основного текста горизонтальной чертой.

**Концевая сноска** размещается в конце раздела документа или всего документа, в котором присутствует символ сноски, и также отделяется от основного текста горизонтальной чертой.

Если на странице имеется несколько обычных сносок, то используется только одна горизонтальная черта. Все концевые сноски также отделяются от основного текста только одной горизонтальной чертой. Word позволяет включать в текст сноски не только обычные символы, но и информацию любого типа, например таблицы, иллюстрации, примечания и т. д.

**Ссылки.** Ссылки используются для перехода из одной части документа на другую или на какой-либо сайт. Существует несколько видов ссылок: *внутренние* и *внешние*.

Внутренние ссылки Word – это ссылки, которые ведут на определенную часть в документе, помогают правильно находить информацию пользователю.

Внешние ссылки Word – ссылки, которые переводят с документа на определенный сайт или другой документ.

**Закладка** указывает на фрагмент текста, которому присвоено уникальное имя. Это имя можно использовать для последующих ссылок. Например, с помощью закладки можно обозначить текст, который необходимо проверить позже. Вместо поиска текста по всему документу можно перейти к нему с помощью диалогового окна *Закладки*. К закладкам можно добавлять перекрестные ссылки. Например, вставив закладку в документ, можно создать перекрестные ссылки, указывающие на этот фрагмент из других фрагментов текста.

**Создание оглавления.** *Оглавление* представляет собой список заголовков документа. Оно используется для просмотра тем, обсуждаемых в документе. Для автоматического создания оглавления документ должен иметь структуру – заголовки, для которых установлены определенные уровни. Тогда Microsoft Word найдет все необходимые заголовки, отсортирует их по уровню, добавит соответствующие номера страниц и отобразит оглавление в документе. (Вставка – Ссылка – Оглавление и указатели).

**Предметный указатель.** Предметный указатель содержит список терминов и тем, рассматриваемых в документе, с указанием страниц, на которых они упоминаются. Чтобы его создать, в документе следует пометить элементы предметного указателя (т. е. указать имя основного элемента и перекрестных ссылок в документе), а затем выполнить сборку предметного указателя.

Можно создать запись указателя для отдельного слова, фразы или символа для темы, которая охватывает диапазон страниц, или для ссылки на другую запись, например «Автотранспорт. См. Грузовые

автомобили». Если вы выберете текст и пометите его как запись указателя, Word добавит специальный знак ХЕ (запись указателя), который содержит помеченный основной текст и любые перекрестные ссылки, которые вы выбрали.

После того как все элементы предметного указателя будут помечены, следует выбрать вид указателя и собрать его. Приложение Word найдет все элементы с соответствующей пометкой, отсортирует их по алфавиту, добавит номера страниц, удалит повторяющиеся элементы и отобразит указатель в документе.

**Графические возможности.** В документах Microsoft Word можно использовать графические объекты и рисунки. Графические объекты включают автофигуры, надписи, кривые, линии и объекты WordArt. Эти объекты являются частью документа Word. Для изменения этих объектов, а также их цветов, заливок, границ и других параметров служит вкладка **Вставка**, группа **Иллюстрации**.



**Рисунки** – включают точечные рисунки, сканированные изображения и фотографии.

**Клип** – картинки из галереи ClipArt. Коллекция клипов включает большой выбор рисунков, упрощающих оформление документа профессионально сделанными изображениями. В ней можно найти все что угодно: от живописного фона до карты и от зданий до людей.

**Фигуры** – команда содержит несколько категорий фигур: линии, основные фигуры, элементы блок-схем, звезды и ленты, а также выноски, которые можно использовать в документах. Можно менять их размер, поворачивать, отражать и комбинировать с другими фигурами, например, с кругами и квадратами, для создания более сложных фигур.

Для вставки художественно оформленного текста служат объекты **WordArt**. С помощью этого инструмента можно создавать наклоненный, повернутый и растянутый текст, а также текст с тенью и текст, вписанный в определенные фигуры.

Объект, рисунок или схема **SmartArt** служит для построения небольших логических схем. Инструмент SmartArt имеется не только в

программе MS Word, но и в остальных приложениях пакета Microsoft Office (Excel, PowerPoint).

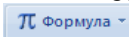
**Диаграмма.** Процесс создания диаграмм в текстовом редакторе Word несложен, но здесь эта функция несколько ограничена по сравнению с аналогичными возможностями табличного процессора Microsoft Excel. Однако если диаграммы будут не очень сложными, возможностей Word вполне хватит. Для создания диаграммы в текстовом редакторе используется специальный модуль – Microsoft Graph.

Диаграмма – это графический способ отображения некоторых числовых данных. Она всегда связана с таблицей значений, в которой размещены числовые данные, собственно представленные в диаграмме. Поэтому построение диаграммы осуществляется одновременно с созданием таблицы значений или после этого. В строках таблицы значений отображены серии данных – наборы чисел, которые будут воспроизведены на вертикальной оси диаграммы, также называемой осью значений. На диаграмме классического вида – гистограмме – серии данных отображаются в виде столбцов одинакового цвета. Столбцы таблицы значений представляют собой категории. Они отображаются на диаграмме вдоль горизонтальной оси, которая также называется осью категорий.

Легендой называют специальную таблицу, в которой указано, что означают цвета элементов диаграммы.

**Редактор формул.** MS Word 2007 оснащен специальным встроенным модулем **Microsoft Equation** (*редактором формул*), который помогает создавать на экране математические уравнения, а в дальнейшем обеспечивает их печать на принтере. Если формула создана с помощью надстройки Microsoft Equation 3.0 в более ранней версии Microsoft Word, изменить эту формулу можно только в Equation 3.0.

Для создания формулы можно использовать встроенный редактор **Microsoft Equation3.0** (лента **Вставка** группа **Текст** команда **Объект**) или **конструктор формул**.

Для вставки формулы, используя конструктор формул, необходимо на вкладке **Вставка** в группе **Символы** щелкнуть по стрелке на кнопке **Формула** , а затем в открывшемся списке выбрать команду **Вставить новую формулу**. В результате появится вкладка **Конструктор** (рис. 2.4), на которой находятся кнопки для создания формул.

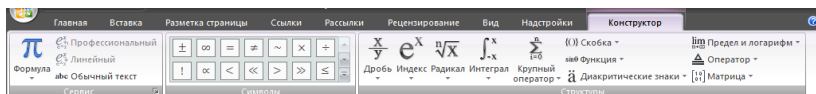


Рис. 2.4. Вкладка, предназначенная для создания формул

На вкладке **Вставка** в группе **Символы** щелкните по стрелке на кнопке **Формула**, а затем в открывшемся списке выберите команду **Вставить новую формулу**. На вкладке **Конструктор** в группе **Структуры** выберите необходимый тип структуры (например, дробь или радикал), а затем щелкните значок необходимой структуры.

Если структура содержит местозаполнители, щелкните местозаполнитель и введите нужные числа или символы.

*Местозаполнители формулы* – это небольшие пунктирные поля в формуле

$$\left( \frac{\square}{2} \right).$$

**Использование мастера слияния при подготовке документов.** **Слияние** – это объединение основного документа, содержащего постоянную часть информации, и источника данных, содержащего переменную часть. При использовании функции слияния в Word происходит объединение основного документа со списком получателей, в результате чего создается комплект выходных документов.

В основном документе содержится текст, являющийся одинаковым для всех выходных документов. В документе может содержаться бланк, текст и инструкции в полях слияния для вставки текста (такого как имена и адреса получателей) для отдельных выходных документов.

Список получателей является базой данных с данными для объединения в выходных документах. Например, список получателей может быть представлен в виде файла базы данных Microsoft Access или листа Excel. Эта база данных обычно представляет собой список имен, адресов, номеров телефонов и личной информации других категорий.

Результатом слияния являются выходные документы. Текст всех выходных документов может быть одинаковым, но возможно форматирование отдельных документов.

### 2.3. Табличные процессоры

**Электронная таблица** (англ. *spreadsheet*) – приложение для работы с компьютерным эквивалентом обычной таблицы, в клетках (ячейках) которой записаны различные данные (числа, даты, тексты) и формулы для вычислений. В учебниках электронные таблицы часто называют табличными процессорами.

*История появления электронных таблиц.* Первая электронная таблица под названием Visi Calc (от англ. *Visible Calculator*) появилась на рынке США в 1979 г. и предназначалась для компьютеров Apple II. Идея создания этой программы принадлежала инженеру-программисту фирмы DEC Дэниелу Бриклину, посещавшему тогда курсы в Гарвардской школе бизнеса.

Экономические вычисления того времени выполнялись на большом листе разлинованной бумаги и при малейшей описке или ошибке даже в одной цифре числа приходилось пересчитывать все связанные с этим числом величины. Программистский опыт натолкнул Бриклина на мысль, что все эти нудные вычисления можно поручить выполнять компьютеру. Идеей он поделился с внештатным инженером-программистом Робертом Фрэнкстоном, который в конце 1978 г. начал писать программу, а к весне следующего года закончил ее. Программа VisiCalc создавала на экране персонального компьютера строки и столбцы чисел, которые могли изменяться, если изменялась какая-либо позиция этой электронной таблицы (рис. 2.5).



Рис. 2.5. Компьютер Apple II и снимок экрана с приложением Visi Calc



В течение первого года программа Visi Calc разошлась тиражом около 100 тыс. экземпляров, существенно увеличив продажи компьютеров Apple II и став одним из первых «приложений-убийц» (англ. *Killerapplication*). Термин «приложение-убийца» является маркетинговым и означает программное приложение настолько желанное пользователями или необходимое им, что они покупают вычислительное оборудование (обычно дорогое) для работы именно этого приложения.

Популярность первых электронных таблиц привела к появлению множества программ-подобий. В 1982 г. Visi Calc была адаптирована под аппаратную архитектуру IBM. В следующем году появилась еще одна популярная программа электронных таблиц Lotus 1-2-3 компании Lotus Software, обогнавшая по продажам Visi Calc.

Компания Microsoft выпустила свою первую версию электронных таблиц Excel для компьютеров Apple Macintosh 30 сентября 1985 г., а первая версия программы для компьютеров под управлением операционной системы Windows вышла в ноябре 1987 г. Сегодня MS Excel является одним из самых популярных приложений в мире.

MS Excel как прикладная программа Windows выполняется в своем собственном окне приложения. Окно приложения MS Excel может содержать несколько окон рабочих книг – документов MS Excel, поэтому одновременно можно работать с несколькими рабочими книгами. Каждая рабочая книга состоит из нескольких рабочих листов, каждый из которых может содержать самостоятельную информацию.

В графическом интерфейсе MS Excel используются различные элементы управления, сгруппированные с помощью объекта, называемого лентой. Она разработана для облегчения доступа к командам и состоит из вкладок, связанных с определенными целями или объектами. Каждая вкладка, в свою очередь, состоит из нескольких групп взаимосвязанных элементов управления. Структура ленточного меню MS Excel представлена на рис. 2.6.

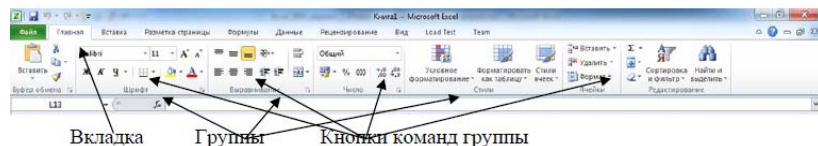


Рис. 2.6. Структура ленты

**Пользовательский интерфейс «лента».** Все команды сведены в группы, состав которых отображается при выборе определенной вкладки, расположенной на ленте (рис. 2.6).

**Вкладки** ориентированы на выполнение конкретной задачи, **группы** на каждой вкладке разбивают задачи на ее составляющие, например, **группа Шрифт** для форматирования элементов текста, **группа Выравнивание** для настройки параметров выравнивания данных в ячейках и т. д. **Кнопки команд** в каждой группе служат для быстрого выполнения команд.

Вкладку можно выбрать, щелкнув по ней левой кнопкой мыши (**ЛКМ**). Для выбранной вкладки отобразятся группы с кнопками команд. В группах справа от названия групп располагаются маленькие значки – **Кнопки вызова диалоговых окон**. По нажатию такой кнопки открывается соответствующее диалоговое окно или область задач, содержащая дополнительные параметры, связанные с данной группой. **Окна диалога** – используются для введения дополнительных данных, необходимых для выполнения тех или иных действий. Некоторые **ОД** содержат **Вкладки** для более удобной группировки настраиваемых параметров. **ОД** можно перемещать по экрану с помощью строки заголовка. **ОД** нельзя свернуть. После настройки всех необходимых параметров **ОД** необходимо закрыть любым корректным способом: нажатием на кнопку **ОК** или **Отмена**. Кроме стандартного набора вкладок, которые отображаются на ленте при запуске Office Excel, имеются вкладки, называемые контекстными инструментами, которые появляются в интерфейсе в зависимости от выполняемой задачи.

*Контекстные вкладки.* Контекстные вкладки позволяют работать с элементом, который выделен на странице, например, с таблицей, изображением или графическим объектом. Если щелкнуть такой элемент, относящийся к нему набор контекстных вкладок, выделенный цветом, появится рядом со стандартными вкладками.

Например, после добавления сводной таблицы появились новые вкладки **Конструктор** и **Параметры**, предназначенные для работы со сводными таблицами (рис. 2.7).

*Контекстные меню.* Контекстные меню содержат команды, которые применяются наиболее часто к выделенным элементам или объектам. Содержание этих меню зависит от контекста обращения к ним.

Для вызова контекстного меню необходимо подвести указатель мыши (**УМ**) к элементу или объекту, с которым вы хотите работать, и выполнить щелчок правой кнопкой мыши (**ПКМ**) (рис. 2.8).

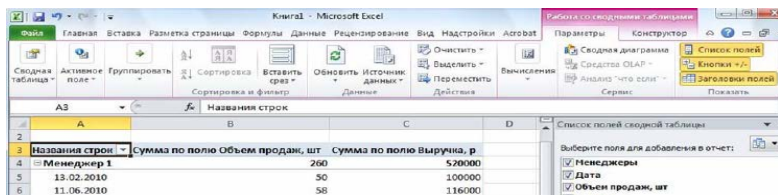


Рис. 2.7. Контекстные инструменты

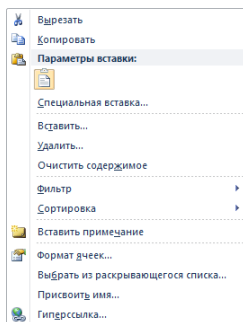


Рис. 2.8. Контекстное меню

*Вкладка (меню) Файл.* Вкладка **Файл** (рис. 2.9) всегда расположена в ленте первой слева. По своей сути вкладка **Файл** представляет собой МЕНЮ.

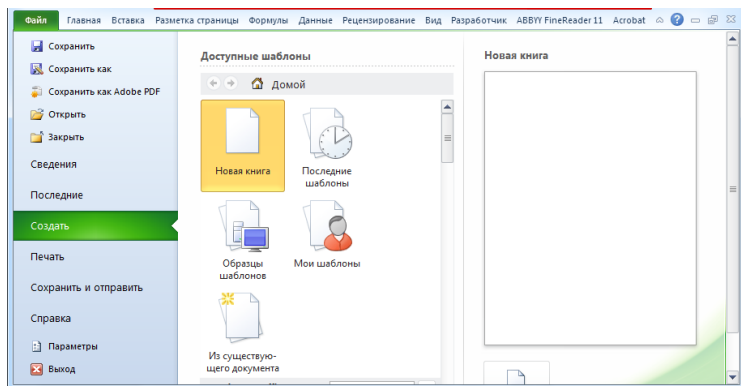


Рис. 2.9. Меню **Файл**

Меню содержит команды для работы с файлами (**Сохранить как, Открыть, Закрыть, Последние, Создать**), для работы с текущим документом (**Сведения, Печать, Доступ**), а также для настройки Excel (**Справка, Параметры**). Команды **Сохранить как** и **Открыть** вызывают соответствующие окна для работы с файловой системой.

Команда **Сведения** открывает раздел вкладки для установки защиты документа, проверки совместимости документа с предыдущими версиями Excel, работы с версиями документа, а также просмотра и изменения свойства документа.

Команда **Последние** открывает раздел вкладки со списком последних файлов, с которыми работали в Excel, в том числе закрытых без сохранения.

*Панель быстрого доступа* по умолчанию расположена в верхней части окна приложения Excel и предназначена для быстрого доступа к наиболее часто используемым функциям. Панель быстрого доступа можно настраивать, добавляя в нее новые команды.

*Смарт-теги* – это набор «интеллектуальных кнопок», которые появляются, когда в них возникает необходимость, и позволяют выбрать наиболее быстрый способ выполнения задачи. В MS Excel смарт-теги помогают управлять функциями автозамены и автозаполнения, изменять форматирование копируемых данных перед их вставкой, устранять ошибки, возникающие в формулах и функциях и т. д.

Смарт-теги предоставляют доступ к параметрам, которые можно настроить без нажатия на кнопки на ПИ или выбора необходимых пунктов меню. Для открытия меню смарт-тега выполните следующие действия:

1. Подведите УМ к тексту с индикатором смарт-тега и дождитесь появления кнопки **Действия смарт-тегов**.

2. Нажмите **кнопку списка**  рядом с кнопкой смарт-тега.

3. Выберите необходимое действие.

**Строка заголовка** содержит информацию об имени программы и имени активной рабочей книги. При открытии новой рабочей книги ей присваивается временное имя **Книга N**.

**Строка формул** отображает действительное содержимое активной ячейки.

**Поле имени** расположено в левой части строки формул и отображает имя активной ячейки.

**Полосы прокрутки** (вертикальная и горизонтальная) предназначены для просмотра содержимого рабочей книги по горизонтали и вертикали с помощью мыши. Бегунок на полосе прокрутки показывает положение текущего отображаемого фрагмента относительно всего содержимого рабочей книги, открытой в окне.

**Ярлычки рабочих листов** содержат имена рабочих листов и используются для выбора нужного листа рабочей книги.

**Строка состояния** представляет собой горизонтальную полосу в нижней части окна рабочей книги. В строке состояния отображаются данные о текущем состоянии содержимого окна и другие сведения, зависящие от контекста.

**Маркер ввода** перемещается по мере ввода текста и указывает место в редактируемом документе, куда будет произведена вставка текста.

**Окна диалога** используются для введения дополнительных данных, необходимых для выполнения тех или иных действий. Некоторые окна диалога (ОД) содержат **Вкладки** для более удобной группировки настраиваемых параметров. ОД можно перемещать по экрану за строку заголовка. ОД нельзя свернуть.

После настройки всех необходимых параметров ОД необходимо закрыть, используя соответствующие кнопки в нижней части ОД.

*Изменение внешнего вида рабочей книги.* Для удобства работы с документом MS Excel предоставляет возможность изменять размеры и форму своего рабочего окна, настраивать скрытие и отображение интерфейсных элементов и т. д. Совокупность параметров отображения листов рабочей книги называется **Представлением**.

*Скрытие и отображение интерфейсных элементов.* В программе MS Excel каждый пользователь имеет возможность настроить отображение или скрытие некоторых элементов интерфейса по своему усмотрению.

Следует учесть, что:

- скрытие или отображение заголовков строк и столбцов, линий разбивки на страницы и сетки распространяется только на активный рабочий лист или группу выделенных листов;
- скрытие или отображение полос прокрутки и ярлычков листов действует в пределах рабочей книги;
- скрытие или отображение строки формул производится для рабочего окна MS Excel в целом;
- строка состояния в MS Excel всегда отображается по умолчанию.

**Управление листами рабочей книги.** При работе с MS Excel все данные хранятся в рабочих книгах. **Рабочая книга MS Excel** – это файл, который может состоять из одного или нескольких рабочих листов (по умолчанию – 3 листа).

**Рабочий лист** – это сами таблицы, диаграммы, слайды, макросы или модули VBA. Использование нескольких рабочих листов, объединенных в одну рабочую книгу, позволяет сгруппировать все данные, относящиеся к конкретной работе.

Имена рабочих листов находятся на ярлычках, расположенных в нижней части окна книги (рис. 2.10). Для перехода с одного листа на другой необходимо указать соответствующий ярлычок, щелкнув по нему **ЛКМ**.

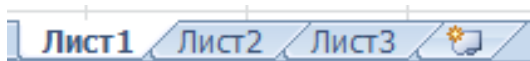


Рис. 2.10. Ярлычки листов

При необходимости пользователь может управлять листами рабочей книги:

- изменять имена рабочих листов;
- добавлять и удалять рабочие листы;
- копировать и перемещать рабочие листы;
- объединять рабочие листы (режим группового выделения).

**Выделение листов.** Для выполнения многих операций по управлению объектами рабочей книги необходимо уметь правильно выделять рабочие листы. Правила выделения листов рабочей книги приведены в табл. 2.1.

Таблица 2.1. Правила выделения рабочих листов

| Объект выделения              | Необходимые действия                                                                                                                                     |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Отдельный лист                | Щелкните <b>ЛКМ</b> ярлычок листа, для поиска невидимого ярлычка листа используйте кнопки прокрутки листов                                               |
| Два или более смежных листа   | Щелкните <b>ЛКМ</b> ярлычок первого листа, а затем, удерживая нажатой клавишу <b>SHIFT</b> , щелкните <b>ЛКМ</b> ярлычок последнего листа                |
| Два или более несмежных листа | Щелкните <b>ЛКМ</b> ярлычок первого листа, а затем, удерживая нажатой клавишу <b>CTRL</b> , последовательно щелкните <b>ЛКМ</b> ярлычки остальных листов |
| Все листы книги               | Щелкните <b>ПКМ</b> на ярлычке любого листа → <b>Выделить все листы</b>                                                                                  |

**Переименование листа.** Во вновь создаваемых рабочих книгах листы всегда имеют имена **Лист1**, **Лист2**, и т. п. Для удобства работы можно изменить встроенное имя листа рабочей книги согласно его содержанию листа. Для этого выполните следующие операции:

1. Щелкните **ПКМ** на ярлычке листа → **Переименовать**. MS Excel выделит имя.
2. Введите с клавиатуры имя, которое вы хотите присвоить листу.
3. Нажмите **ENTER**.

**Изменение цвета ярлычка.** Для более удобной идентификации листов рабочей книги вы можете «раскрасить» ярлычки листов. Для этого выполните следующие операции:

1. Выделите листы, которым требуется назначить цвета.
2. Щелкните **ПКМ** на ярлычке листа → **Цвет ярлычка...**
3. Выберите требуемый цвет из палитры.
4. Нажмите **ОК**.

**Вставка нового листа.** В случае если в рабочей книге недостаточно рабочих листов, вы можете добавить любое количество листов нужного типа (рис. 2.11) в любое место рабочей книги. Для этого выполните следующие операции:

1. Щелкните **ПКМ** на ярлычке листа → **Вставить...**
2. Выберите нужный шаблон (например, **Лист**) → **ОК**.

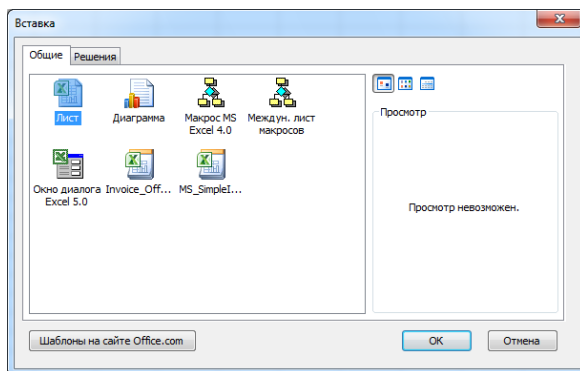


Рис. 2.11. Вставка листа

MS Excel вставит новый рабочий лист слева от выделенного, присвоив ему имя **Лист N**, где **N** – порядковый номер листа в рабочей книге.

Для быстрой вставки листа можно щелкнуть по ярлыку **Вставить лист**, который располагается последним среди ярлыков листов или воспользоваться комбинацией **Shift + F11**.

*Удаление листа.* Совершенно не обязательно хранить в рабочей книге те листы, которые не используются в работе. Пустые листы или листы с устаревшими данными можно удалить. Для этого выполните следующие операции:

1. Щелкните **ПКМ** на ярлычке листа → **Удалить**.

2. Нажмите кнопку **Удалить**. MS Excel удалит лист. Вы не можете отменить удаление листа.

*Перемещение, копирование листов.* Помимо вставки и удаления листов во время работы с книгой может понадобиться скопировать или переместить некоторые листы в пределах одной книги или из одной рабочей книги в другую.

*Перемещение или копирование листов в пределах одной рабочей книги.* Для **перемещения** листов необходимо выполнить следующие действия.

1. Выделить перемещаемый лист.

2. Удерживая нажатой **ЛКМ**, переместить лист в нужное место. Место вставки листа помечается черным треугольником.

Для **копирования** листа его следует перемещать, удерживая нажатой клавишу **CTRL**.

1. Выделите перемещаемый лист.

2. Удерживая нажатыми клавиши **CTRL** и **ЛКМ**, переместите лист в нужное место.

*Перемещение или копирование листов между разными книгами.*

1. Откройте книгу, в которую нужно переместить или скопировать листы.

2. Переключитесь в книгу, содержащую листы, которые требуется скопировать или переместить.

3. Выделите эти листы.

4. Щелкните **ПКМ** на ярлычке листа → **Переместить/скопировать...** (рис. 2.12).

5. В поле **в книгу:** → выберите имя книги, в которую будете перемещать листы.

6. В поле **перед листом:** → выделите лист, перед которым хотите вставить перемещаемые листы.

7. Нажмите **ОК**.

Для **копирования** листа установите флажок в поле **Создать копию** (рис. 2.12).



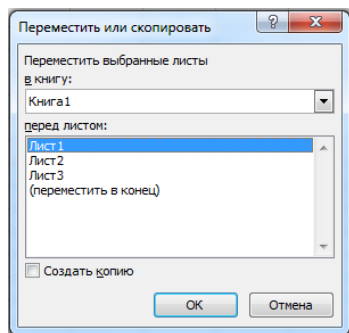


Рис. 2.12. Перемещение/Копирование листов между разными книгами

***Объединение листов.*** MS Excel дает возможность заполнять данными сразу несколько листов рабочей книги – работать в режиме группового выделения. Если использовать групповой режим, то вводимая в один из листов информация будет отображаться на всех листах, входящих в группу. Для объединения листов в группу необходимо их выделить.

Все объединенные листы вы можете одновременно удалить.

Для отмены объединения щелкните **ПКМ** на любом (из объединенных) ярлычке листа и выберите **Разгруппировать листы**.

***Изменение количества листов новой рабочей книги.*** Чтобы изменить число листов во вновь создаваемой рабочей книге, необходимо выполнить следующие операции:

1. Меню **Файл** → **Параметры** → **Общие**.
2. В поле **При создании новых книг** в счетчике **Число листов** установите необходимое количество листов (до 255).
3. Нажмите **ОК**.

После этого все новые книги будут создаваться с указанным количеством листов.

***Ввод и изменение информации на рабочем листе.*** В ячейку рабочего листа можно вводить различную информацию. Ввод всегда производится в активную ячейку. Процесс ввода аналогичен вводу текста в любой другой программе. Весь введенный текст отображается в строке формул и в самой ячейке.

***Выбор активной ячейки.*** Для ввода информации на рабочий лист необходимо **активизировать** ячейку. Активная ячейка обрамляется рамкой, а адрес данной ячейки отображается в поле имени (рис. 2.13).

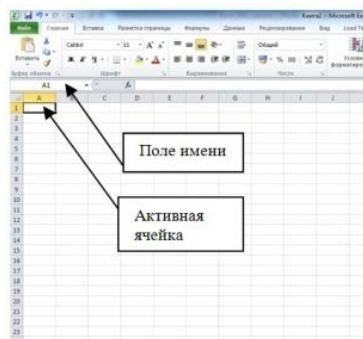


Рис. 2.13. Выбор ячейки

Для выбора ячейки необходимо:

- либо щелкнуть **ЛКМ** по ячейке – указатель мыши должен иметь вид;
- либо переместиться к ней, используя клавиши навигации на клавиатуре, приведенные в табл. 2.2.

Таблица 2.2. **Клавиши навигации и клавиатура**

| Клавиши                  | Действие                                                |
|--------------------------|---------------------------------------------------------|
| Клавиши навигации        | Переход на одну ячейку вверх, вниз, влево или вправо    |
| Ctrl + клавиша навигации | Переход к краю текущей области, содержащей данные       |
| Home                     | Переход в начало строки                                 |
| Ctrl + Home              | Переход в начало листа – на ячейку A1                   |
| Ctrl + End               | Переход в последнюю ячейку листа, содержащую информацию |
| Page Down                | Переход на один экран вниз                              |
| Page Up                  | Переход на один экран вверх                             |
| Alt + Page Down          | Переход на один экран вправо                            |
| Alt + Page Up            | Переход на один экран влево                             |

**Ввод Информации.** Ввод данных в активную ячейку осуществляется следующим образом.

1. Выбрать нужную ячейку.
2. Ввести информацию.
3. Завершить ввод одним из способов:
  - нажать клавишу **ENTER** – курсор переместится на ячейку вниз;
  - нажать клавишу **TAB** – курсор переместится на ячейку вправо;

– нажать кнопку  в **Строке формул** – курсор останется в текущей ячейке.

Для отмены ввода информации в ячейку необходимо:

– либо нажать клавишу **ESC** на клавиатуре;

– либо нажать кнопку  в **Строке формул**;

– если ширина столбца не достаточна для отображения информации, помещенной в ячейку, MS Excel либо отсечет часть информации (если это текст), либо отобразит в ячейке #####. Для нормального отображения содержимого ячейки необходимо увеличить ширину столбца.

*Типы информации.* Ввод и редактирование данных может выполняться по-разному в зависимости от типа данных. MS Excel позволяет вводить в ячейку информацию, одного из следующих типов:

- текст;
- числа;
- денежная информация;
- дата и время;
- формулы.

Правила ввода соответствующих типов данных зависят от параметров региональных настроек ОС.

*Текст.* Для MS Excel текст – это любая строка, состоящая из цифр, пробелов и нецифровых символов, длиной не более 255 символов. Если в ячейке необходимо начать новую строку, следует нажать комбинацию клавиш **Alt + Enter**.

Для того чтобы ввести в виде текста данные другого типа, необходимо перед ними вставить символ апострофа (').

Чтобы MS Excel воспринимал любые данные, вводимые в ячейку, как текст, необходимо предварительно в ячейке задать текстовый формат данных (вкладка **Главная** → группа **Число** → кнопка справа от названия группы → **Числовые форматы** → **Текстовый**).

MS Excel автоматически выравнивает текст по левому краю ячейки.

*Числа.* Чтобы данные соответствовали числовому формату, они могут содержать только следующие символы:

**0 1 2 3 4 5 6 7 8 9 + - ( ) / \$ % Е е.**

Вводимые в ячейку числа интерпретируются как константы. Стоящий перед числом знак плюс (+) игнорируется. Символ, используемый в качестве разделителя целой и дробной частей числа, зависит от параметров **Региональных настроек ОС**.

Независимо от количества отображаемых разрядов, числа хранятся с точностью до 15 разрядов. Все разряды после 15-го преобразуются в нули (0).

Ввод в ячейку простых дробей осуществляется следующим образом:

**[целая часть] пробел [числитель]/[знаменатель]**

Целая часть должна вводиться всегда, чтобы дроби не воспринимались как даты.

Для интерпретации чисел, например, инвентаризационных номеров, как текста необходимо назначить текстовый формат незаполненным ячейкам или начинать ввод числа с символа апострофа (').

MS Excel автоматически выравнивает числа по правому краю ячейки.

**Денежная информация.** В качестве денежной информации MS Excel воспринимает число с указанием денежной единицы. Символ, используемый в качестве денежной единицы, зависит от **Региональных настроек ОС**.

MS Excel автоматически выравнивает деньги по правому краю ячейки.

**Дата и Время.** В MS Excel дата и время интерпретируются как числа. При вводе значений даты или времени происходит их автоматическое распознавание, и общий формат ячейки заменяется встроенным форматом даты или времени. Время суток и даты могут быть использованы в вычислениях.

Символы, использующиеся в качестве разделителей в форматах даты и времени, зависят от **Региональных настроек ОС**.

Для ввода даты и времени суток в одну ячейку необходимо в качестве разделителя даты и времени ввести пробел. Индикатор времени суток (AM или PM) также отделяется от времени пробелом.

MS Excel автоматически выравнивает дату и время по правому краю ячейки.

**Формулы.** MS Excel использует **адрес ячейки** для идентификации ячейки, содержащей данные, используемые в формуле. Для создания формулы необходимо выполнить следующие шаги.

1. Выбрать ячейку, в которую будет помещен результат.
2. Ввести с клавиатуры знак равенства (=).
3. Ввести фиксированное значение (константу) или **ЛКМ** выделить ячейку, содержащую нужные данные.

4. Ввести математический оператор для вычисления результата:  
+ (сложение), – (вычитание), \* (умножение), / (деление), ^ (возведение в степень), % (вычисление процента от числа).

5. Завершить ввод.

Ячейка, содержащая формулу, отображает результат вычислений, но не саму формулу. MS Excel отображает фактическую формулу в **Строке формул** (рис. 2.14).

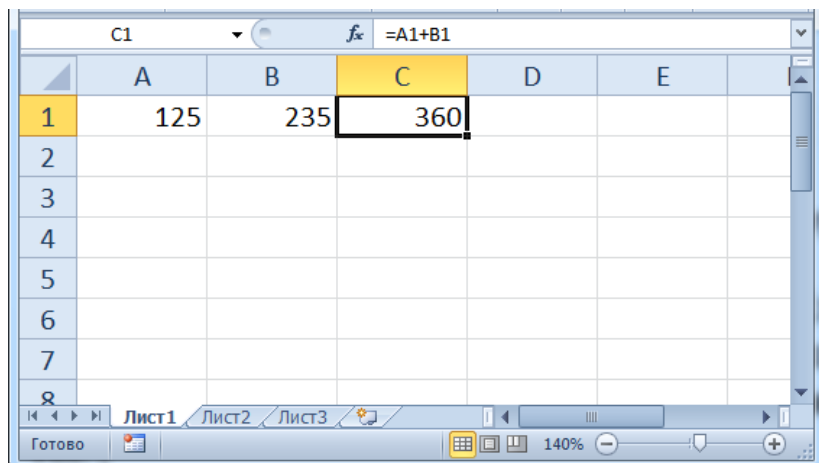


Рис. 2.14. Отображение формулы в **Строке формул**

**Настройка региональных установок.** Для просмотра и изменения региональных установок выполните следующие действия.

1. **Пуск** → **Панель управления** → **Язык и региональные стандарты**.

2. Просмотрите настройки данных: **Число, Деньги, Время, Краткая и Полная дата**.

3. Для изменения настроек нажмите кнопку **Изменить этот формат...**

4. В **ОД Настройка региональных параметров** настройте параметры данных на каждой вкладке: **Число, Денежная единица, Время и Дата**.

5. Нажмите **ОК**.

**Изменение Информации.** Выделение информации на рабочем листе. Под выделением понимается маркировка одной ячейки или груп-

пы ячеек (диапазона) с целью выполнения в дальнейшем операций с этой ячейкой или группой.

Выделенная область отличается от невыделенных частей рабочего листа цветом ячеек и более широкой линией границы.

Правила выделения диапазонов ячеек рабочего листа приведены в табл. 2.3.

Таблица 2.3. Правила выделения диапазонов ячеек рабочего листа

| Выделяемый диапазон                  | Правила выделения                                                                                                                                                                       |
|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Отдельная ячейка                     | Щелкните ЛКМ по ячейке или перейдите к ней, используя клавиши со стрелками (клавиши навигации)                                                                                          |
| Диапазон смежных ячеек               | Выделите первую ячейку диапазона, а затем, удерживая нажатой клавишу SHIFT, выделите последнюю ячейку диапазона. Для перемещения к последней ячейке можно использовать полосы прокрутки |
| Несмежные ячейки или диапазоны ячеек | Выделите первую ячейку (или первый диапазон ячеек), а затем, удерживая нажатой клавишу CTRL, последовательно выделите остальные ячейки (или диапазоны)                                  |
| Все ячейки листа                     | Нажмите кнопку Выделить все. <div data-bbox="789 710 918 782" data-label="Image"> </div>                                                                                                |
| Всю строку или весь столбец          | Щелкните ЛКМ по заголовку строки (1) или заголовку столбца (2) <div data-bbox="464 829 638 909" data-label="Image"> </div>                                                              |

Активная ячейка выделенного диапазона остается неокрашенной. Остальная часть области затенена.

Чтобы отменить выделение ячеек, щелкните ЛКМ в любой ячейке на листе.

**Удаление информации.** В случае если информация в ячейке (или диапазоне ячеек) больше не нужна, ее следует удалить. Для этого выполните следующие операции.

1. Выделите ячейки, информацию из которых необходимо удалить.
2. Выполните одно из следующих действий:
  - выберите вкладку Главная → группу Редактирование → меню кнопки Очистить → Очистить содержимое;
  - нажмите клавишу Delete на клавиатуре;
  - ПКМ → Очистить содержимое.

**Изменение Информации.** При работе с большим количеством информации часто возникает необходимость исправления данных.

Вы можете изменить информацию, прежде чем закончите вводить ее в активную ячейку, или можете изменять существующую информацию в активной ячейке. Редактирование данных активной ячейки производится либо непосредственно в ячейке, либо в строке формул.

**Редактирование содержимого ячейки в строке формул.** Для редактирования данных активной ячейки можно воспользоваться строкой формул. Для этого выполните следующие действия.

1. Выберите ячейку, содержимое которой необходимо отредактировать.

2. Поместите указатель мыши в **Строке формул** щелчком ЛКМ.

3. Внесите необходимые изменения.

4. Нажмите кнопку  в **Строке формул** для завершения редактирования.

**Редактирование содержимого в ячейке.** Чтобы произвести непосредственное редактирование содержимого ячейки, выполните следующие действия.

1. Выберите ячейку, содержимое которой необходимо отредактировать, и нажмите клавишу **F2**.

2. Отредактируйте содержимое ячейки.

3. Либо нажмите клавишу **Enter**, либо нажмите клавишу **Tab** для завершения редактирования.

При редактировании в любой момент можно отказаться от внесенных изменений:

- либо нажав клавишу **Esc** на клавиатуре;

- либо нажав кнопку  в **Строке формул**.

Если необходимо полностью заменить содержимое ячейки, выделите ее и введите новое значение.

**Использование средств автоматизации ввода данных.** Для ускорения набора повторяющихся текстовых данных в MS Excel предусмотрено средство автоматического ввода – **автозавершение**.

Автозавершение позволяет автоматически заполнять ячейку, если первые введенные символы совпадают с начальными символами ранее введенных записей, или повторно вводить данные, используя команду **Выбрать из списка**.

Другим средством, позволяющим ускорить ввод однотипной информации в смежные ячейки, является автозаполнение. С помощью этого средства мы можем заполнить смежные ячейки календарными

или рабочими датами, повторяющейся текстовой информацией или изменяющимися числовыми данными.

**Автоматический ввод одинаковой информации в диапазон ячеек.** Если в некоторый диапазон ячеек рабочего листа необходимо ввести одну и ту же информацию, необходимо выполнить следующие действия.

1. Выделите ячейки, в которые необходимо ввести одинаковую информацию.
2. Введите данные в активную ячейку.
3. Завершите ввод, нажав **Ctrl + Enter**.

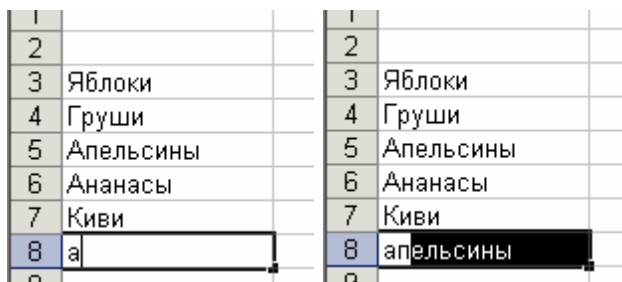
**Автозавершение.** При вводе **текстовых данных** в столбец MS Excel создает список.

Столбец не должен содержать пустых ячеек. При последующем вводе данных в столбец MS Excel просматривает список **Автозавершения** и подсвечивает предлагаемый текст.

Для использования метода автозавершения выполните следующие действия.

1. Выделите внизу столбца, содержащего текстовую информацию, ячейку, в которую необходимо ввести следующее значение.
2. Начните ввод текста. Если первые введенные символы совпадают с начальными символами ранее введенной записи, MS Excel отобразит эту запись в ячейке (**Ошибка! Источник ссылки не найден.**).
3. Если подсказка содержит необходимый текст, нажмите **Enter** (рис. 2.15).

В случае если необходимо набрать другой текстовый элемент, продолжите ввод самостоятельно!



|   |           |   |           |
|---|-----------|---|-----------|
| 1 |           | 1 |           |
| 2 |           | 2 |           |
| 3 | Яблоки    | 3 | Яблоки    |
| 4 | Груши     | 4 | Груши     |
| 5 | Апельсины | 5 | Апельсины |
| 6 | Ананасы   | 6 | Ананасы   |
| 7 | Киви      | 7 | Киви      |
| 8 | а         | 8 | апельсины |
| 9 |           | 9 |           |

Рис. 2.15. Использование **Автозавершения**

**Автозавершение** работает только с текстовыми данными.



Вы можете выбирать данные из списка **Автозавершения**, используя **ALT + ↓** либо **ПКМ → Выбрать из раскрывающегося списка... (Pick From List...)**.

Вы можете отключить **Автозавершение**: меню **Файл → Параметры → Дополнительно** → в поле **Параметры правки** снимите флажок **Автозавершение значений ячеек** → **ОК**.

**Автозаполнение.** В случае если необходимо быстро заполнить диапазон смежных ячеек однородной, последовательно возрастающей или убывающей информацией, можно воспользоваться **Автозаполнением (копирование в смежные ячейки)**. При **Автозаполнении** диапазона ячеек формулами MS Excel автоматически корректирует формулу.

Для копирования информации в смежные ячейки необходимо выполнить следующие действия.

1. Введите информацию в первую ячейку диапазона.
2. Подведите указатель мышки к **маркеру заполнения** до появления черного крестика + (рис. 2.16, шаг 1).
3. Удерживая нажатой **ЛКМ**, протащите выделенную ячейку в направлении заполнения.
4. Нажмите на кнопку смарт-тега и выберите нужное действие (рис. 2.16, шаг 2).

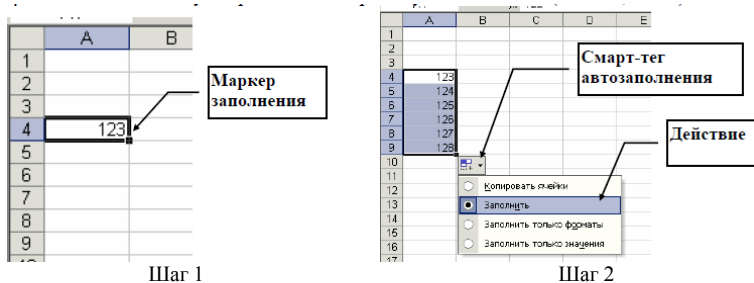


Рис. 2.16. Автозаполнение смежных ячеек

В случае заполнения смежных ячеек датами, список действий смарт-тега увеличивается (рис. 2.17): предоставляется возможность заполнения по дням, по рабочим дням, по месяцам либо по годам.

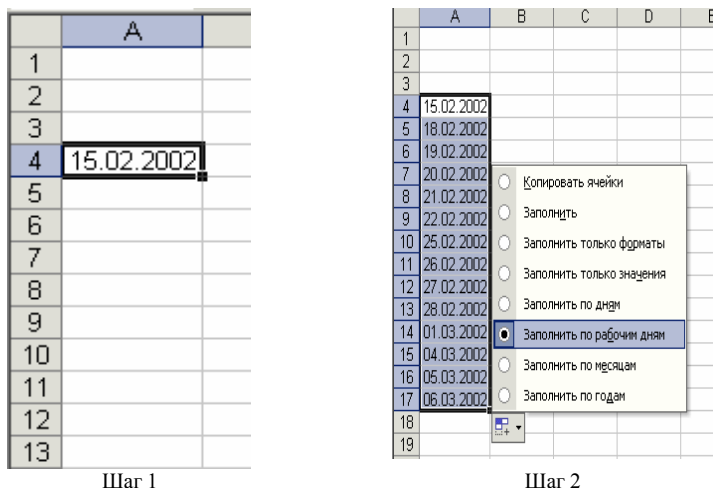


Рис. 2.17. Автозаполнение смежных ячеек датами

В случае выбора **Заполнять только форматы**, в смежные ячейки копируется только оформление исходной ячейки.

**Создание Пользовательских** списков автозаполнения. При работе с MS Excel может возникнуть потребность в создании не только числовых, но и текстовых последовательностей. MS Excel позволяет создавать последовательности текстовых значений автоматически, при этом такая возможность существует в нескольких вариантах: Excel умеет распознавать числа в текстовых значениях; Excel может создавать последовательность на основе заданного ранее списка автозаполнения.

**Список автозаполнения** – список текстовых значений, расположенных в определенном порядке. Если при автозаполнении в первой ячейке диапазона находится значение, принадлежащее одному из списков, то остальные ячейки будут последовательно заполнены значениями из этого списка (рис. 2.18).

Для создания собственного списка **Автозаполнения** необходимо выполнить следующие действия:

1. Меню **Файл** → **Параметры** → **Дополнительно** → **Общие** → **Изменить списки...** (рис. 2.19).

2. В поле **Элементы списка** введите желаемый список, разделяя элементы списка нажатием клавиши **ENTER**.

3. Нажмите кнопку **Добавить**.

4. Нажмите **ОК**.

|    | A                                 | B     | C        | D                                       | E         | F         |
|----|-----------------------------------|-------|----------|-----------------------------------------|-----------|-----------|
| 1  | <b>Примеры встроенных списков</b> |       |          | <b>Примеры пользовательских списков</b> |           |           |
| 2  | <b>автозаполнения</b>             |       |          | <b>автозаполнения</b>                   |           |           |
| 3  | Среда                             | июнь  | Сентябрь | Лейтенант                               | Шестерка  | Виноград  |
| 4  | Четверг                           | июль  | Октябрь  | Старший лейтенант                       | Семерка   | Яблоки    |
| 5  | Пятница                           | авг.  | Ноябрь   | Капитан                                 | Восьмерка | Груши     |
| 6  | Суббота                           | сент. | Декабрь  | Майор                                   | Девятка   | Апельсины |
| 7  | Воскресенье                       | окт.  | Январь   | Подполковник                            | Десятка   | Мандарины |
| 8  | Вторник                           | нояб. | Февраль  | Полковник                               | Валет     | Виноград  |
| 9  | Среда                             | дек.  | Март     | Генерал-майор                           | Дама      | Яблоки    |
| 10 |                                   | янв.  | Апрель   | Генерал-лейтенант                       | Король    | Груши     |
| 11 |                                   | фев.  | Май      | Генерал-полковник                       | Туз       | Апельсины |
| 12 |                                   | март  | Июнь     |                                         |           |           |
| 13 |                                   | апр.  | Июль     |                                         |           |           |
| 14 |                                   | май   | Август   |                                         |           |           |

Рис. 2.18. Примеры текстовых последовательностей

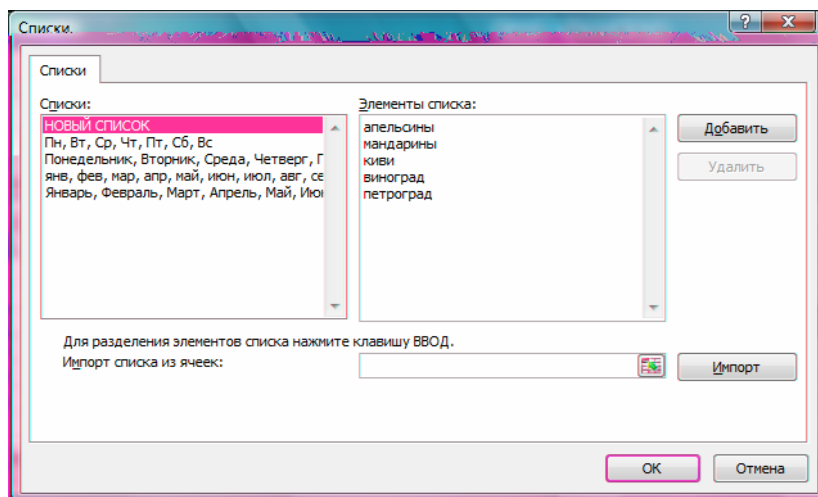


Рис. 2.19. Создание списка Автозаполнения

**Перемещение и копирование данных.** Часто бывает необходимо поместить одинаковую информацию в разных ячейках одного и того же листа или на разных листах одной рабочей книги. Для этого выполняется операция копирования данных. В случае если необходимо пе-

ренести информацию из одной ячейки в другую или с одного листа на другой, необходимо выполнить процедуру перемещения информации.

MS Excel предоставляет различные способы перемещения и копирования информации: с использованием буфера обмена и с помощью мышки.

**Перемещение и копирование данных через Буфер обмена.** Для передачи информации из одного документа в другой (из одного места документа в другое) используется буфер обмена. **Буфер обмена (БО)** – это зарезервированная область памяти, в которую помещаются копируемые или перемещаемые объекты. **Буфер обмена** является общим для всех приложений MS Office и позволяет скопировать до 24 различных фрагментов информации. Для копирования или перемещения информации через буфер обмена следует выполнить следующие действия.

1. Выделите перемещаемый или копируемый фрагмент.
2. Поместите его в БО одним из следующих способов:
  - выберите команду **Вырезать** или **Копировать** на вкладке **Главная** из группы **Буфер обмена** или контекстного меню (**ПКМ**);
  - нажмите кнопку **Вырезать**  или **Копировать**  на панели быстрого доступа.
3. Установите курсор в том месте, куда вы хотите поместить перемещаемый или копируемый фрагмент.
4. Вставьте фрагмент из БО одним из следующих способов:
  - нажмите клавишу **ENTER** на клавиатуре;
  - выберите команду **Вставить** на вкладке **Главная** из группы **Буфер обмена** и выберите необходимый вариант (рис. 2.20);
  - нажмите кнопку контекстного меню (**ПКМ**).

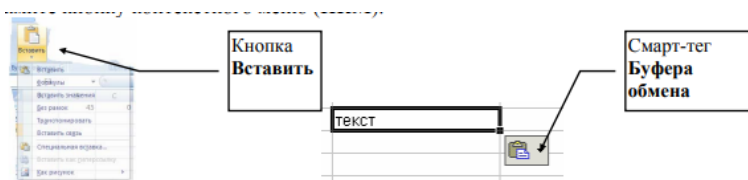


Рис. 2.20. Вставка информации из **Буфера обмена**

5. Нажмите кнопку смарт-тега и выберите необходимое действие:
  - отобразите содержимое БО в ОЗ (рис. 2.21), используя список разделов ОЗ;

- для вставки фрагмента из БО в ОЗ в выбранную ячейку рабочего листа нажмите кнопку списка возле фрагмента → **Вставить**;
- если фрагмент больше не нужен, удалите его из окна БО в ОЗ, выбрав из списка возле фрагмента – **Удалить**;
- если необходимо полностью очистить БО, нажмите кнопку **Очистить все**.

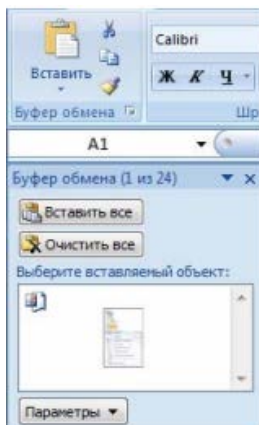


Рис. 2.21. Содержимое  
Буфера обмена

**Простое копирование формул.** MS Excel позволяет копировать формулы, используя те же способы, что и при копировании данных на рабочем листе. При копировании формулы, адреса ячеек, участвующих в формуле, изменяются в направлении копирования.

Для копирования ячейки, содержащей формулу, выполните следующие действия.

1. Выделите ячейку, содержащую формулу, которую требуется скопировать.
2. Нажмите кнопку **Копировать** на вкладке **Главная**.
3. Выделите ячейку, в которую требуется вставить скопированную формулу.
4. Нажмите стрелку снизу от кнопки **Вставить**, а затем выберите команду **Формулы**.

Для вставки копируемой формулы в новую ячейку достаточно нажать клавишу **ENTER** на клавиатуре.

Для очистки БО нажмите клавишу **ESC**.

**Специальное копирование формул.** Если нет необходимости пересчитывать формулу, можно формулу заменить на вычисленное значение.

Для этого сделайте следующее:

- выделите ячейки, содержащие формулы.

- нажмите кнопку **Копировать** на вкладке **Главная**.

- нажмите стрелку снизу кнопки **Вставить** на вкладке **Главная** и выберите вариант **Вставить значения**.

Для специального копирования можно также воспользоваться **ОД Специальная вставка** (рис. 2.22) – кнопка **Вставить** → **Специальная вставка...**

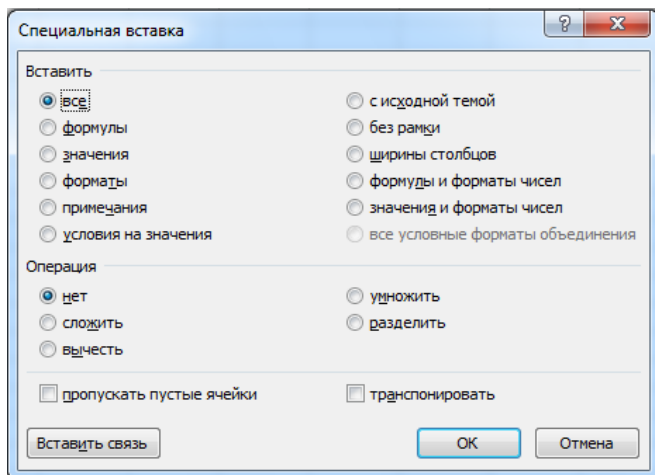


Рис. 2.22. Диалоговое окно **Специальная вставка**

**Форматирование элементов и данных рабочего листа.** *Управление элементами рабочего листа.* При создании таблиц часто приходится удалять ненужные или добавлять недостающие столбцы или строки. Для корректного отображения содержимого ячейки необходимо управлять размерами столбца или строки. MS Excel предоставляет возможность настройки ширины строк и столбцов, добавления и удаления строк и столбцов или некоторых диапазонов ячеек, а также объединения диапазона ячеек в одну ячейку.

*Изменение ширины столбца или высоты строки.* В случае если информация, размещаемая в ячейке, требует увеличения или уменьшения ширины столбца или изменения высоты строки, необходимо выполнить следующие действия.

1. Подведите указатель мыши к правой границе заголовка столбца, размер которого хотите изменить. (Для изменения высоты строки, указатель мыши нужно подвести к нижней границе заголовка строки).

2. Удерживая нажатой ЛКМ, переместите границу в сторону изменения размера столбца (строки).

3. Отпустите ЛКМ.

Для автоматического изменения ширины столбца или высоты строки сделайте двойной щелчок ЛКМ на границе заголовка столбца или строки.

*Изменение ширины столбца или высоты строки, используемых по умолчанию.* Все столбцы и строки рабочего листа по умолчанию имеют одинаковую ширину и высоту соответственно. Если необходимо изменить это значение, выполните следующие действия:

1. Щелкните ПКМ ярлычок любого листа → **Выделить** все листы.

2. Вкладка **Главная** → группа **Ячейки** → **Формат** → **Ширина по умолчанию...**

Для изменения высоты строк выполните команды: **Вкладка Главная** → группа **Ячейки** → **Формат** → **Высота строки**.

3. Введите новое значение.

4. Нажмите **ОК**.

**Добавление нового столбца или строки.** Для добавления в таблицу нового столбца или строки, выполните следующие действия.

1. Выделите столбец, **перед** которым вы хотите добавить новый столбец.

Для добавления новой строки выделите строку, **выше** которой вы хотите вставить новую.

Щелкните кнопку **Вставить** в группе **Ячейки** на вкладке **Главная**. MS Excel добавит новый столбец левее выделенного.

*Удаление столбца или строки.* Если таблица содержит лишние строки или столбцы, их следует удалить. Для этого выполните следующие действия.

1. Выделите столбец или строку, которые вы хотите удалить.

2. Щелкните ПКМ на выделенном диапазоне → **Удалить**. MS Excel удалит выделенный столбец или строку.

*Вставка диапазона ячеек.* Часто бывает необходимо добавить в таблицу некоторый диапазон ячеек, для размещения новых данных. Для этого выполните следующие действия.

1. Выделите диапазон, перед которым вы хотите вставить ячейки.
2. Щелкните ПКМ на выделенном диапазоне → **Вставить...**
3. В ОД **Добавление ячеек** выберите подходящий вариант (рис. 2.23).
4. Нажмите **ОК**. MS Excel вставит новый диапазон ячеек и сдвинет информацию в соответствии с вашим выбором.

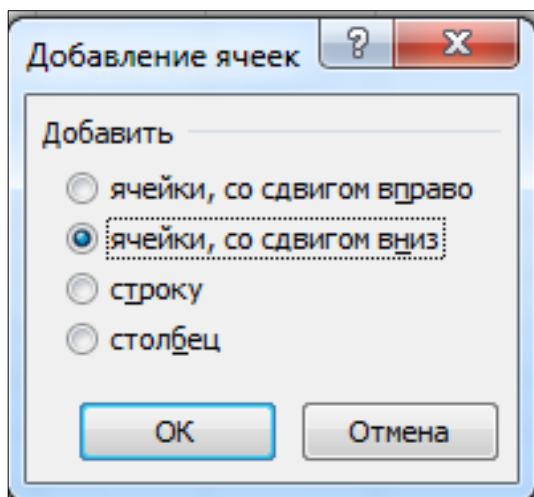


Рис. 2.23. Окно **Добавление ячеек**

*Удаление диапазона.* При перестраивании рабочей книги иногда бывает необходимо не очистить, а удалить диапазон ячеек. Удаляя диапазон, вы удаляете и содержимое ячеек и сами ячейки, при этом смежные ячейки вместе с содержимым будут перемещены вверх или вниз, чтобы заполнить освободившееся пространство.

Для удаления диапазона, используйте следующие действия.

1. Щелкните ПКМ на диапазоне, который вы хотите удалить → **Удалить...**
2. В окне **Удаление ячеек** выберите нужный вариант.
3. Нажмите **ОК**. MS Excel удалит диапазон и сдвинет смежные ячейки в указанном направлении.



Для удаления целиком строки или столбца щелкните ПКМ на заголовке строки или столбца и выберите **Удалить**.

*Защита нескольких ячеек рабочего листа.* Создадим документ следующего вида (рис. 2.24):

|    | A               | B            | C                             | D              |
|----|-----------------|--------------|-------------------------------|----------------|
| 1  | Ф.И.О.          | Гражданство  | Маршрут туристических поездок |                |
| 2  |                 |              | начальный пункт               | конечный пункт |
| 3  | Абадей А.А.     | Россия       | Казань                        | Бангкок        |
| 4  | Арбузов Р.Б.    | Россия       | Санкт-Петербург               | Канберра       |
| 5  | Гарсиа Р.       | Мексика      | Мехико                        | Асунсьон       |
| 6  | Ивановская К.М. | Белоруссия   | Витебск                       | Астрахань      |
| 7  | Кэмпбелл С.     | США          | Лос-Анджелес                  | Монреаль       |
| 8  | Лукишин Д.И.    | Россия       | Москва                        | Магадан        |
| 9  | Петренко С.П.   | Украина      | Киев                          | Сингапур       |
| 10 | Саввич М.       | Румыния      | Белград                       | София          |
| 11 | Сидоров П.А.    | Россия       | Нижний Новгород               | Париж          |
| 12 | Феррас Л.       | Бразилия     | Сан Паулу                     | Рабат          |
| 13 | Цариган С.С.    | Айзербайджан | Баку                          | Ереван         |
| 14 |                 |              |                               |                |
| 15 |                 |              |                               |                |

Рис. 2.24. Образец документа

Выделите весь рабочий лист. (Кнопка выделения всего рабочего листа находится на пересечении заголовков строк и столбцов).

Выполните команду **Главная/(Ячейки) Формат/Формат ячеек** и на вкладке **Защита** выполните команду сбросить оба флажка – **Защищаемая ячейка** и **Скрыть формулы**, после чего нажмите кнопку **ОК** (рис. 2.25).

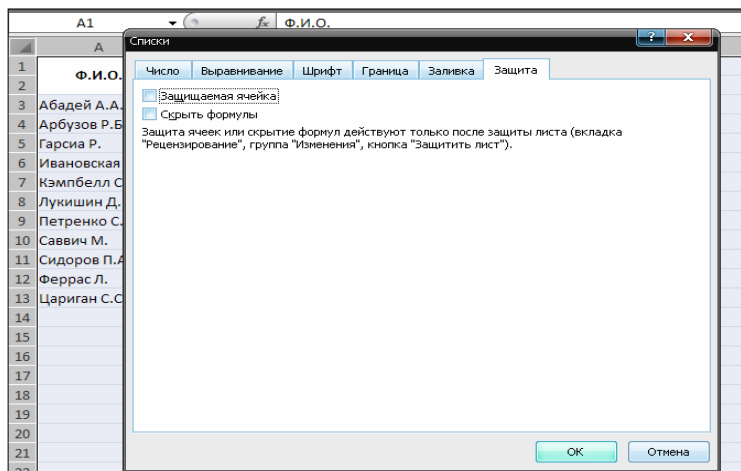


Рис. 2.25. Вкладка **Защита** для всего рабочего листа

Выделите ячейки **A1:B14** и **C1:D2**. Выполните команду **Главная/ (Ячейки) Формат/Формат ячеек** и на вкладке **Защита** установите флажок **Защищаемая ячейка**. Нажать **ОК** (рис. 2.26).

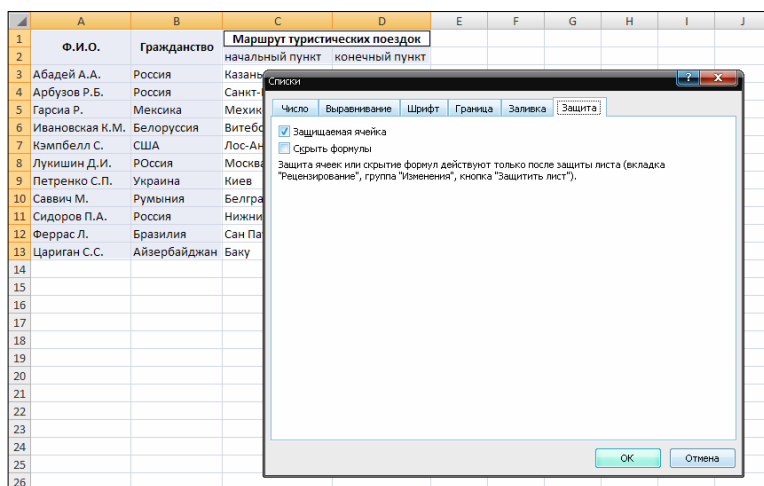


Рис. 2.26. Вкладка **Защита** для диапазона ячеек

Для включения защиты активного рабочего листа выполните команду **Рецензирование/(Изменения) Защитить лист**, затем в открывшемся диалоговом окне установите флажок **Защитить лист и содержимое защищаемых ячеек** (рис. 2.27).

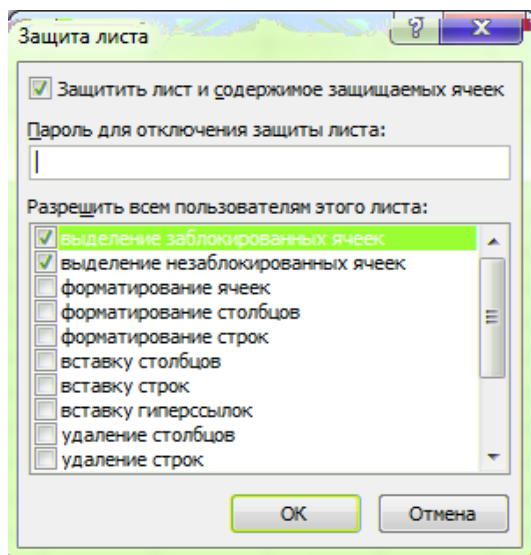


Рис. 2.27. Защита листа

*Защита рабочего листа за исключением нескольких ячеек.* Создадим документ следующего вида (рис. 2.28):

|   | A                            | B      | C          | D           |
|---|------------------------------|--------|------------|-------------|
| 1 | Наименование                 | Кол-во | Цена       | Сумма       |
| 2 | Nvidia GeForce MX440 64Mb    | 2      | 1 326,00р. | 2 652,00р.  |
| 3 | Nvidia GeForce4 FX5700 256Mb | 1      | 5 053,00р. | 5 053,00р.  |
| 4 | ATI Radeon 9200SE 64Mb       | 3      | 1 554,00р. | 4 662,00р.  |
| 5 | ATI Radeon 9800SE 128Mb      | 2      | 3 175,00р. | 6 350,00р.  |
| 6 |                              |        | ИТОГО      | 18 717,00р. |

Рис. 2.28. Образец документа

Выделите весь рабочий лист, выполните команду **Главная/ (Ячейки) Формат/ Формат ячеек** на вкладке **Защита** установите оба флаж-

ка – **Защищаемая ячейка** и **Скрыть формулы**. Нажмите кнопку **ОК** (рис. 2.29).

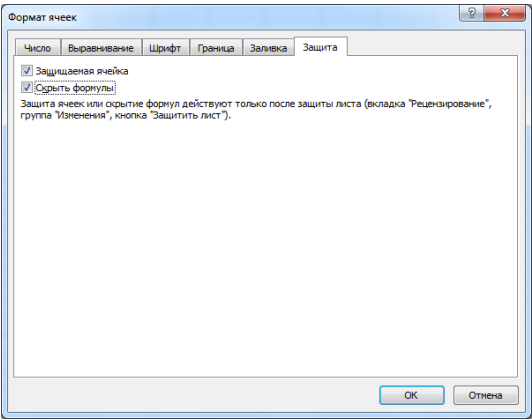


Рис. 2.29. Вкладка **Защита**

Выделите ячейки **B2:C5**, выполните команду **Главная/ (Ячейки) Формат/Формат ячеек** и на вкладке **Защита** снимите оба флажка – **Защищаемая ячейка** и **Скрыть формулы**. Нажмите кнопку **ОК** (рис. 2.30).

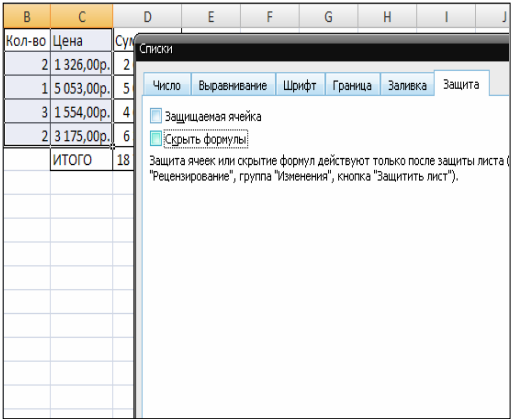


Рис. 2.30. Вкладка **Защита**

Установите защиту листа так, как описано в предыдущем примере.

Установите пароль: **Главная/ (Ячейки) Защитить лист**, в области **Пароль для отключения защиты листа** введите ваш пароль. При просьбе подтверждения пароля введите тот же самый пароль, что и в первый раз (рис. 2.31).

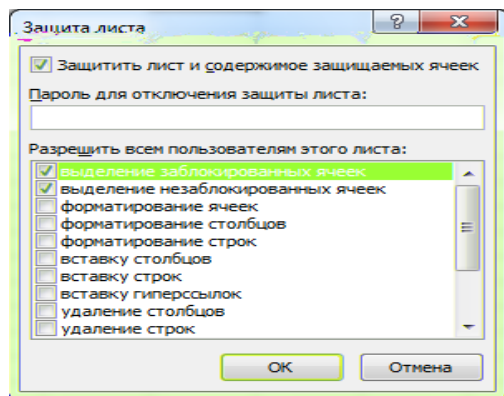


Рис. 2.31. Защита листа

**Снятие защиты листа.** Для снятия защиты листа нужно выполнить команду **Рецензирование/(Изменения) Снять защиту листа**. Если при защите был установлен пароль, то нужно ввести этот пароль в диалоговом окне **Снять защиту листа** (рис. 2.32).

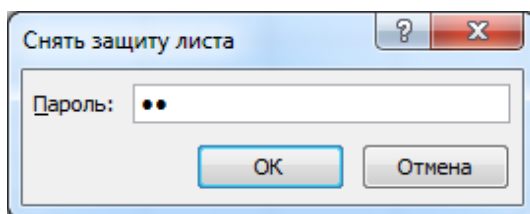


Рис. 2.32. Снять защиту

**Форматирование чисел и текста.** Вводимая в ячейки MS Excel информация может быть отображена на экране различными способами. Для изменения формы отображения и доступа к информации MS Excel использует средства форматирования и защиты. MS Excel

распознает тип вводимой информации и выбирает форму ее представления или формат самостоятельно. Форматирование информации на рабочем листе позволяет создавать удобное представление табличных данных, делая их более привлекательными и наглядными.

**Форматирование чисел.** Независимо от количества отображаемых разрядов, в MS Excel числа хранятся с точностью до 15 разрядов. Если число имеет больше 15 значащих цифр, то разряды после 15-го преобразуются в нули. Очень маленькие, очень большие, а также числа, не помещающиеся в ячейку, отображаются в экспоненциальном формате (например,  $4.2E+07$ ). Если после преобразования числа в экспоненциальный формат оно все равно не помещается в ячейке, то ячейка заполняется символами решетки (#) (рис. 2.33).

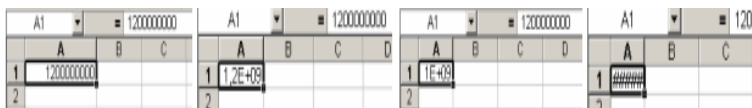


Рис. 2.33. Представление числа при разной ширине ячейки

Вы можете изменить формат представления данных в ячейках рабочей книги по своему усмотрению. При этом форматировать ячейки можно как до, так и после ввода данных.

**Использование ленты** для оформления чисел. Вкладка **Главная** (группа **Число**) предлагает вам наиболее быстрый способ форматирования чисел. На рис. 2.34 представлены кнопки группы **Число**, предназначенные для быстрого форматирования чисел.

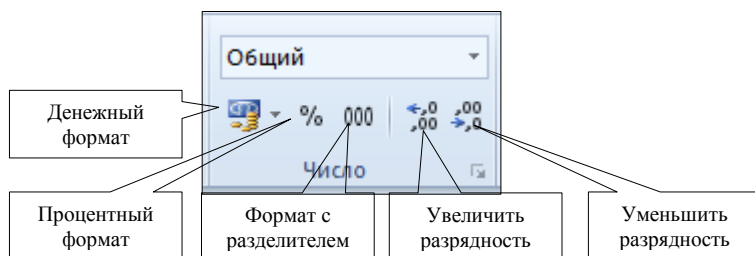


Рис. 2.34. Кнопки форматирования чисел в группе **Число**

**Использование дополнительных числовых форматов.** Для задания более сложного оформления чисел вы можете воспользоваться окном

диалога **Формат ячеек**, выбрав подходящий формат из списка существующих форматов. Для этого выполните следующие действия:

1. Выделите ячейки, для которых необходимо задать числовой формат.

2. Вкладка **Главная** → группа **Число** → нажмите кнопку справа от названия группы **Число**. Откроется ОД **Формат ячеек** (рис. 2.35).

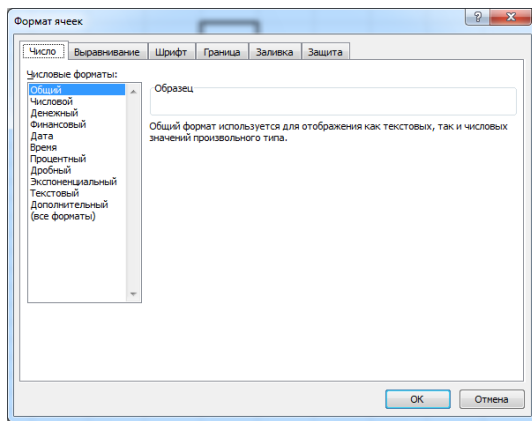


Рис. 2.35. Диалоговое окно **Формат ячеек**: вкладка **Число**

3. В списке **Числовые форматы** → выберите нужный формат.

4. Задайте все необходимые настройки в правой части окна диалога.

5. Проверьте правильность оформления в окне **Образец**.

6. Нажмите **ОК**.

Данный вариант предоставляет больше возможностей для форматирования чисел.

*Создание пользовательского формата числа.* В случае если в списке **Числовые форматы** нет нужного варианта, можно создать пользовательский формат числа.

Для этого выполните следующие действия.

1. Выделите ячейки, для которых необходимо создать пользовательский формат.

2. Вкладка **Главная** → группа **Число** → нажмите кнопку справа от названия группы **Число**.

3. В списке **Числовые форматы** выберите категорию **Все форматы**.

4. В поле **Тип** введите нужный формат, используя соответствующие символы шаблона, представленные в табл. 2.4.

5. Нажмите **ОК**.

Таблица 2.4. Символы, используемые при создании пользовательских форматов

| Символ                 | Назначение                                                                                                                              |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| # (решетка)            | Отображает только значащие цифры, незначащий ноль – не отображается                                                                     |
| 0 (ноль)               | Отображает любое число; незначащие нули отображаются, если количество разрядов числа меньше количества нулей в формате                  |
| ? (вопрос)             | До и после десятичной запятой вместо незначащих нулей отображаются пробелы (используется для выравнивания положения десятичной запятой) |
| , (запятая)            | Отображает десятичные знаки                                                                                                             |
| пробел                 | Отображает числа с разделением на группы по разрядам                                                                                    |
| условие                | Вводится в секцию формата в квадратных скобках и состоит из оператора сравнения и значения                                              |
| <b>Шаблоны дат</b>     |                                                                                                                                         |
| М                      | Месяцы в виде 1–12                                                                                                                      |
| ММ                     | Месяцы в виде 01–12                                                                                                                     |
| МММ                    | Месяцы в виде Янв–Дек                                                                                                                   |
| ММММ                   | Месяцы в виде Январь – Декабрь                                                                                                          |
| Д                      | Дни в виде 1–31                                                                                                                         |
| ДД                     | Дни в виде 01–31                                                                                                                        |
| ДДД                    | Дни в виде Пн–Вс                                                                                                                        |
| ДДДД                   | Дни в виде Понедельник – Воскресенье                                                                                                    |
| ГГГГ                   | Год в виде 1900–9999                                                                                                                    |
| <b>Шаблоны времени</b> |                                                                                                                                         |
| ч                      | Часы в виде 0–23                                                                                                                        |
| чч                     | Часы в виде 00–23                                                                                                                       |
| [ч]:мм                 | Прошедшее время в часах; например, 25:02                                                                                                |
| м                      | Минуты в виде 0–59                                                                                                                      |
| мм                     | Минуты в виде 00–59                                                                                                                     |
| [мм]:сс                | Прошедшее время в минутах; например, 63:46                                                                                              |
| с                      | Секунд в виде 0–59                                                                                                                      |
| сс                     | Секунд в виде 00–59                                                                                                                     |
| [сс]                   | Прошедшее время в секундах                                                                                                              |
| <b>Шаблоны текста</b>  |                                                                                                                                         |
| @                      | Отождествляет любой введенный в ячейку текст. Если в текстовой секции знак @ пропущен, то вводимый текст не будет отображен             |

Формат может содержать до четырех разделов, отделяемых друг от друга точкой с запятой и определяющих правила написания: положительных чисел, отрицательных чисел, нулевых значений и текста положительное\_число; отрицательное\_число; ноль; текст



В соответствующем разделе формата в квадратных скобках можно указать **Цвет**. Например: # ##0,00;[Красный]-# ##0,00;[Синий]0,00. Для английской версии MS Office цвет необходимо указать по-английски [Red]. Для отображения текстовой константы используются двойные кавычки. Например: # ##0,00“ рублей” или 0,00“ кг.”

**Выравнивание информации в ячейках.** MS Excel производит автоматическое выравнивание текста по левому краю, а чисел, дат, времени и результатов вычисления формул – по правому краю. Вы можете изменить выравнивание информации по своему усмотрению.

*Использование ленты* для выравнивания информации. Для выравнивания информации в ячейках можно воспользоваться кнопками в группе **Выравнивание** на вкладке **Главная**, представленными на рис. 2.36.

*Использование ОД для выравнивания информации.* Окно диалога **Формат ячеек** предоставляет возможность задать горизонтальное и вертикальное выравнивание информации в ячейках, а также изменить ориентацию содержимого ячеек. Для этого выполните следующие действия.

1. Выделите оформляемые ячейки. Вкладка **Главная** → группа **Выравнивание** → кнопка вызова вкладки **Выравнивание** ОД **Формат ячеек** (рис. 2.36).

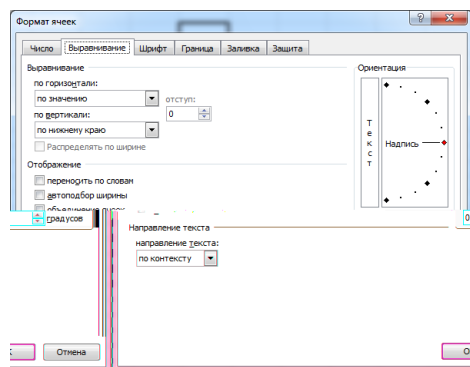


Рис. 2.36. Диалоговое окно **Формат ячеек**:  
вкладка **Выравнивание**

2. Выберите необходимый вариант выравнивания из списка по горизонтали.

3. Выберите необходимый вариант выравнивания из списка по вертикали.

4. Задайте необходимую ориентацию в группе **Ориентация**.

5. Нажмите **ОК**.

*Задание границ и цвета для диапазона ячеек.* Для более наглядного отображения данных таблица может быть оформлена с применением границ и заливки ячеек цветом.

Вы можете задать границы выделенного диапазона ячеек, используя специальные кнопки в группе **Шрифт** на вкладке **Главная** (рис. 2.37). Для этого выполните следующие действия.

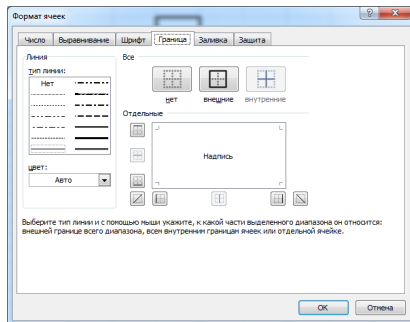
1. Выделите диапазон, который вы хотите оформить.

2. Выберите из списка кнопки **Границы** необходимый стиль границы.

3. Выберите из списка кнопки **Цвет заливки** нужный цвет.

*Использование ОД для обрамления информации.* Для расширения возможностей оформления ячеек с помощью задания границ и цвета используйте соответствующие вкладки окна диалога **Формат ячеек**:

Вкладка **Главная** → группа **Шрифт** → кнопка вызова **ОД Формат ячеек** → вкладка **Граница** используется для задания границ выделенного диапазона (рис. 2.38).



2. Вкладка **Главная** → группа **Стили** → кнопка **Форматировать как таблицу**.

В поле **Список форматов** содержится список встроенных форматов (рис. 2.39).



Рис. 2.39. Окно стилей таблицы

3. Наведите мышь на нужный формат. Стиль сразу же отобразится на выделенном диапазоне.

Если вы хотите создать новый стиль, нажмите кнопку **Создать стиль таблицы...**

4. Нажмите ЛКМ для подтверждения выбранного стиля. MS Excel отформатирует диапазон ячеек согласно настройкам выбранного формата.

*Копирование форматов.* Часто бывает необходимо скопировать оформление одного диапазона на другой диапазон ячеек. Для копирования формата используйте следующие действия.

1. Выделите диапазон ячеек, содержащий необходимое форматирование.

2. Щелкните по кнопке **Формат по образцу** на вкладке **Главная** в группе **Буфер обмена** (рис. 2.40). MS Excel отобразит указатель мыши в форме кисти.

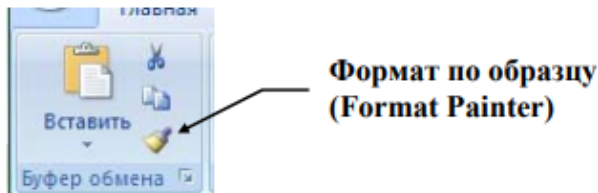


Рис. 2.40. Кнопка **Формат** на вкладке **Главная**

Для многократного копирования формата дважды щелкните кнопку **Формат по образцу**.

3. Выделите диапазон (или ячейку), на который вы хотите скопировать оформление.

4. Щелкните по кнопке **Формат по образцу** на вкладке **Главная** для отключения режима копирования форматов (или нажмите клавишу ESC на клавиатуре).

*Быстрое форматирование ячеек.* Для быстрого форматирования диапазона ячеек вы можете воспользоваться предопределенными форматами. Для этого выполните следующие действия.

1. Выделите диапазон, который необходимо отформатировать.

2. Вкладка **Главная** → группа **Стили** кнопка → **Стили ячеек**.

В **Галерее стилей** содержится список встроенных стилей (рис. 2.41).

3. Наведите мышь на нужный формат. Стилль сразу же отобразится на выделенном диапазоне.

4. Нажмите ЛКМ для подтверждения выбранного стиля. MS Excel отформатирует диапазон ячеек согласно настройкам выбранного формата. **Формат по образцу Галерея стилей**.

Вы можете создать свой новый стиль ячеек. Для этого выполните следующие действия:

– нажмите кнопку **Создать стиль ячейки...** ;

– ознакомьтесь с информацией в **ОД Стилль** (рис. 2.42) и нажмите кнопку **Формат**.

Выберите параметры форматирования → **ОК**. Введите имя вашего стиля и нажмите **ОК**. Стилль отобразится вверху галереи в категории **Пользовательские**.

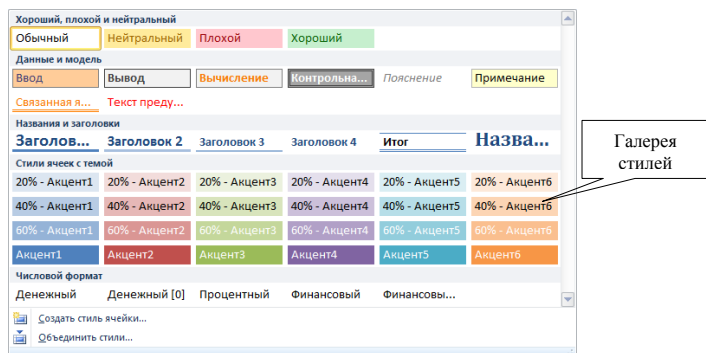


Рис. 2.41. Окно списка стилей

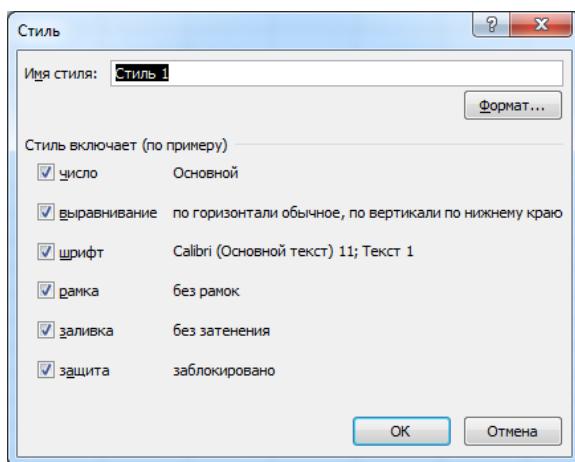


Рис. 2.42. Диалоговое окно **Стиль**

*Очистка форматов.* Для того чтобы отменить форматирование диапазона, выполните следующие действия:

1. Выделите диапазон ячеек, для которого вы хотите удалить форматирование.

2. Вкладка **Главная** → группа **Редактирование** → кнопка **Очистить** → **Очистить форматы**. MS Excel очищает форматирование выбранного диапазона.

Происходит очистка всех форматов ячейки: форматирования текста, пользовательского формата чисел (если он был задан), заливки и обрамления ячеек.

Для отмены конкретных элементов оформления ячейки используйте настройки **ОД Формат Ячеек**.

**Выполнение вычислений. Автовычисление. Работа с функциями.** Традиционные средства, которые связаны с понятием «электронная таблица» – это средства вычислений. MS Excel предоставляет в наше распоряжение мощный механизм, позволяющий выполнять вычисления над данными рабочего листа, используя встроенные средства автовычислений, формулы и функции.

*Использование Автовычисления.* Если необходимо выполнить быстрое вычисление значений диапазона без ввода формулы в ячейку, можно воспользоваться полем **Автовычисления** (в строке состояния). По умолчанию в поле **Автовычисления** отображается сумма выделенного диапазона (рис. 2.43).

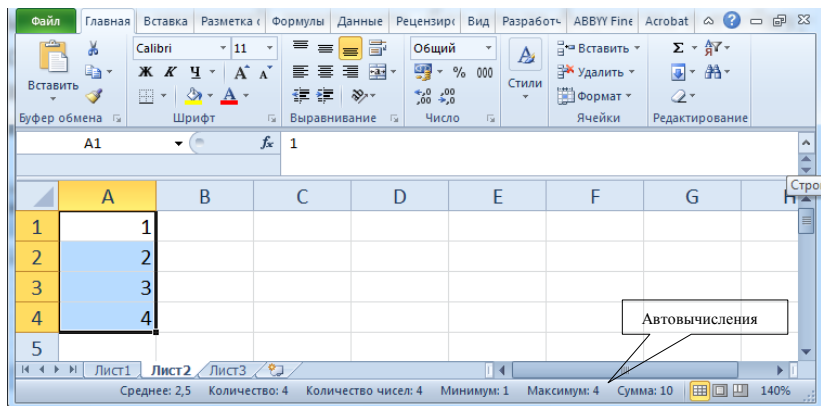


Рис. 2.43. Поле **Автовычисления**

Используя контекстное меню (ПКМ) в поле **Автовычисления**, можно добавить (убрать) результаты вычисления выделенного диапазона на основании других функции (рис. 2.44).

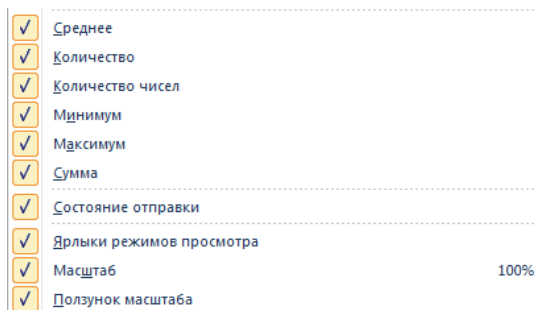


Рис. 2.44. Контекстное меню поля **Автовычисления**

*Использование функций.* Функции – заранее определенные формулы, которые выполняют вычисления по заданным величинам, называемым аргументами, и в указанном порядке, определяемом синтаксисом. Функции MS Excel позволяют выполнять как простые, так и сложные вычисления, связанные с решением определенных задач. Некоторые вычисления могут быть выполнены как с помощью формул, так и с помощью аналогичных им функций. Например: Формула  $=C7+D7+E7$  складывает содержимое ячеек C7, D7 и E7.

Функция  $=СУММ(C7:E7)$  – выполняет то же самое.

Функции, используемые в программе MS Excel, имеют следующий синтаксис:  $=\text{ФУНКЦИЯ}(\text{аргумент1}, \text{аргумент2}, \dots)$

В некоторых случаях может потребоваться использование функции как одного из аргументов другой функции. Например, на рис. 2.45 показано, как функции СРЗНАЧ и СУММ вложены в функцию ЕСЛИ.

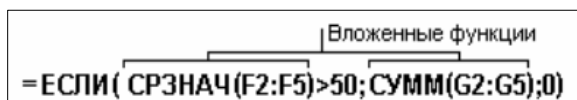


Рис. 2.45. Формула с вложенными функциями

*Ввод функций вручную.* В случае если вы хорошо знакомы с используемыми функциями, вы можете вводить их вручную непосредственно в ячейки рабочего листа.

Для этого выполните следующие действия.

1. Установите курсор в ячейку результата.
2. Введите знак равенства.

3. Введите имя функции. Как только вы начнете вводить имя функции, всплывет подсказка-список функций, выберите нужную функцию и нажмите клавишу TAB.

4. Введите открывающую скобку (при использовании подсказки скобка появится автоматически). MS Excel отобразит всплывающую подсказку, содержащую синтаксис и аргументы указанной функции (рис. 2.46).

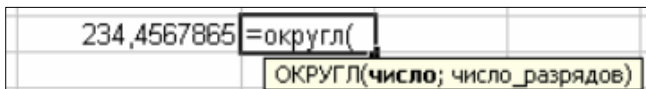


Рис. 2.46. Всплывающие подсказки для функций

5. Введите значения аргументов функции.

Аргументами функции могут быть: число, текст, логическое значение (ИСТИНА и ЛОЖЬ (TRUE, FALSE)), ссылка на ячейку или диапазон ячеек (смежных или не смежных), формулы или функции. В каждом конкретном случае необходимо использовать соответствующий тип аргумента.

6. Нажмите клавишу ENTER.

MS Excel отобразит формулу в строке формул, как только вы ее введете. В ячейке отобразится результат вычисления.

*Использование Мастера функций.* Вы можете упростить процедуру создания формулы, используя **Мастер функций**, который предоставляет доступ ко всем встроенным функциям MS Excel. Для того чтобы создать функцию с помощью **Мастера**, выполните следующие действия.

1. Выделите ячейку, в которую вы хотите ввести функцию.

2. Щелкните по кнопке списка, справа от кнопки **Автосумма** на вкладке **Главная** в группе **Редактирование**.

3. Выберите либо функцию, предложенную в списке, либо **Другие функции...** (рис. 2.47).

MS Excel отобразит **ОД Мастер функций** (рис. 2.48).

4. В списке **Категория** выберите необходимую категорию функций.

Последние 10 недавно использованных функций находятся в соответствующей категории – 10 недавно использовавшихся.



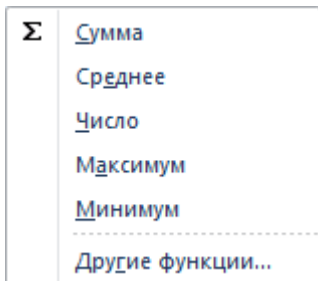


Рис. 2.47. Кнопка **Автосумма** на вкладке **Главная**

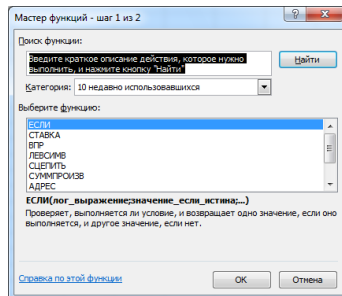


Рис. 2.48. Окно диалога **Мастер функций**

5. Выберите необходимую функцию в списке **Выберите функцию**. Появится окно **Аргументы функции** (рис. 2.49).

MS Excel автоматически предлагает диапазон прилежащих смежных ячеек в качестве первого аргумента функции.

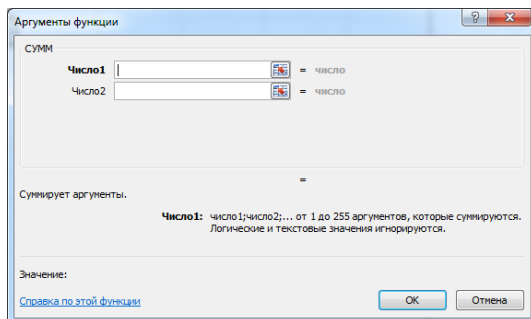


Рис. 2.49. Диалоговое окно **Аргументы функции** для функции СУММ

6. Задайте необходимые аргументы функции.

7. Нажмите кнопку **ОК**.

**Ошибки в формулах и функциях.** Ошибки формулы могут привести к ошибочным значениям, а также вызвать непредсказуемые результаты. Если формула содержит ошибку, не позволяющую выполнить вычисления или отобразить результат, MS Excel отобразит сообщение об ошибке. В табл. 2.5 представлено описание ошибок, возникающих при работе с формулами, и указаны возможные причины, которые могли вызвать данную ошибку.

Таблица 2.5. Коды ошибок в формулах

| Обозначение ошибки | Причина возникновения                                                                                                                             |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| ####               | Столбец недостаточно широк или дата и время являются отрицательными числами                                                                       |
| #ЗНАЧ!             | Используется недопустимый тип аргумента или операнда                                                                                              |
| #ДЕЛ/0             | Деление числа на 0 (нуль)                                                                                                                         |
| #ИМЯ?              | MS Excel не может распознать имя, используемое в формуле                                                                                          |
| #Н/Д               | Значение недоступно функции или формуле                                                                                                           |
| #ССЫЛКА!           | Ссылка на ячейку указана неверно                                                                                                                  |
| #ЧИСЛО!            | Используются неправильные числовые значения в формуле или функции                                                                                 |
| #ПУСТО!            | Задано пересечение двух областей, которые в действительности не имеют общих ячеек. Оператором пересечения областей является пробел между ссылками |

*Обнаружение ошибок.* При возникновении ошибки в ячейке MS Excel отображает в ее левом верхнем углу зеленый треугольник (смарт-тег ошибки). При выборе такой ячейки появляется – смарт-тег проверки ошибок (рис. 2.50).

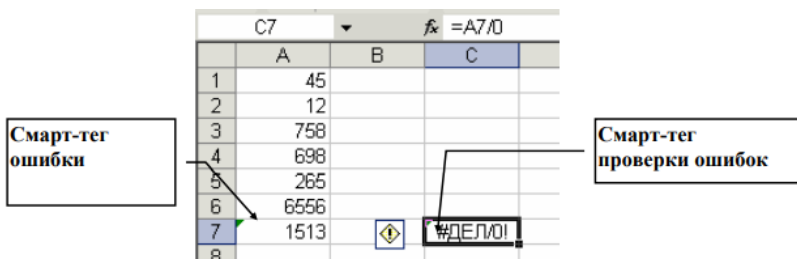


Рис. 2.50. Смарт-тег проверки ошибок

*Исправление ошибок.* Для исправления ошибки можно воспользоваться списком действий, предоставляемым смарт-тегом проверки ошибок (рис. 2.51). В том случае если будет выбран пункт **Пропустить ошибку**, такая ошибка при последующих проверках отображаться не будет.

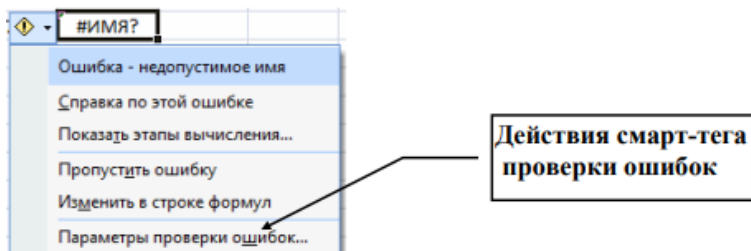


Рис. 2.51. Действия смарт-тега проверки ошибок

Для проверки ошибок необходимо выполнить следующие действия.

1. Выберите лист, который требуется проверить на наличие ошибок.
2. Вкладка **Формулы** → группа **Зависимости формул** → **Проверка наличия ошибок**. Появится ОД **Контроль ошибок** (рис. 2.52).

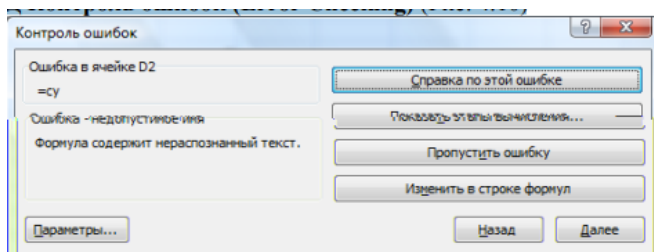


Рис. 2.52. ОД Контроль ошибок

3. Нажмите в правой части ОД кнопку **Далее**.

4. Доведите до конца проверку ошибок.

Для повторной проверки пропущенных ранее ошибок выберите команду **Параметры...** → **Сброс пропущенных ошибок** → **ОК** → **Продолжить**.

Вы можете самостоятельно определить, какие типы ошибок должны отслеживаться системой (в поле **Правила контроля ошибок**: меню **Файл** → **Параметры** → **Формулы**).

**Имена и абсолютные ссылки. Условное форматирование. Использование ссылок.**

*Понятие относительных и абсолютных ссылок в формулах.* В зависимости от выполняемых задач в MS Excel можно использовать от-

носительные ссылки, определяющие положение ячейки относительно положения ячейки, содержащей формулу, или абсолютные ссылки, которые всегда указывают на конкретные ячейки. Если перед буквой или номером стоит знак доллара, например, \$A\$1, то ссылка на столбец или строку является абсолютной. При копировании формулы относительные ссылки автоматически корректируются, а абсолютные ссылки остаются неизменными.

*Создание Абсолютных ссылок.* Если необходимо, чтобы ссылки не изменялись при копировании формулы в другую ячейку, воспользуйтесь абсолютными ссылками. Для этого выполните следующие действия.

1. Выделите ячейку с формулой.
2. В строке формул выделите ссылку, которую необходимо изменить (например, H12).

3. Нажимайте F4, пока не получите желаемой ссылки.

Каждое нажатие F4 переключает тип ссылки в следующей последовательности:

- \$H\$12 – абсолютный столбец и абсолютная строка;
- H\$12 – относительный столбец и абсолютная строка;
- \$H12 – абсолютный столбец и относительная строка;
- H12 – относительный столбец и относительная строка.

4. Нажмите ENTER. MS Excel запишет формулу с абсолютными ссылками.

**Условное форматирование.** В некоторых случаях удобно использовать разные форматы представления данных в одной и той же ячейке в зависимости от значения содержащихся в ней данных. Одним из способов решения указанной проблемы является использование условного форматирования. В сравнении со стандартными форматами, условные форматы предоставляют несколько большие возможности по количеству накладываемых условий и позволяют в зависимости от условий менять не только цвет, но также шрифт и обрамление.

*Условное форматирование по значению.* Если значения форматизируемых ячеек являются критерием для их оформления, выполните следующие действия.

1. Выделите ячейки, к которым хотите применить условное форматирование.

2. Вкладка **Главная** → группа **Стили** → **Условное форматирование...** . В появившемся меню выберите нужный набор правил: **Правила выделения ячеек** или **Правила отбора первых и последних значений**.

3. При выборе **Правила выделения ячеек** выберите **Больше**, откроется ОД **Больше** (рис. 2.53).

4. В левом окне установите число. В правом окне в выпадающем списке выберите подходящий формат. Вы можете выбрать **Пользовательский формат...** и в ОД **Формат ячеек** установить требуемые параметры форматирования.

5. При выборе **Правила отбора первых и последних значений** выберите **Первые 10 %**, откроется ОД **Первые 10 %** (рис. 2.54).

6. В левом окне установите требуемое значение границы, определяющей % наивысших значений ячеек, для которых выполнятся правила форматирования. В правом окне в выпадающем списке выберите подходящий формат. Вы можете выбрать **Пользовательский формат...** и в ОД **Формат ячеек** установить требуемые параметры форматирования.

7. Для добавления новых условий повторите шаги 2–6.

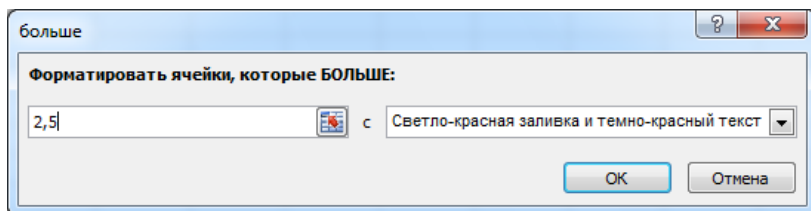


Рис. 2.53. ОД Условное форматирование по критерию **Больше**

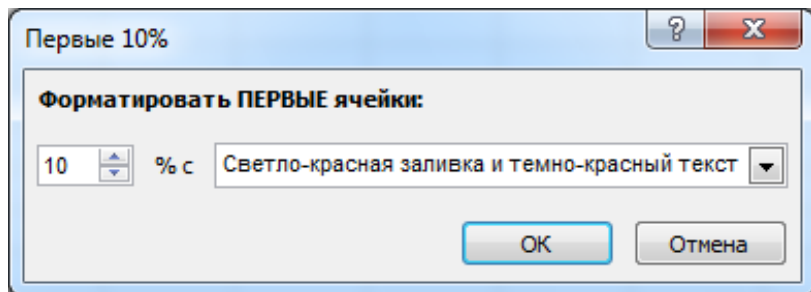


Рис. 2.54. ОД Условное форматирование по критерию **Первые 10 %**

*Создание правил условного форматирования.* Вы можете разработать дополнительные правила для визуализации данных, для этого выполните следующие действия.

1. Выделите ячейки, к которым хотите применить условное форматирование.

2. Вкладка **Главная** → группа **Стили** → **Условное Форматирование...** → **Создать правило...**. В том случае, если в качестве критерия оформления ячеек должна выступать формула в появившемся ОД Создание правила форматирования, выберите **Использовать формулу** для определения форматлируемых ячеек (рис. 2.55).

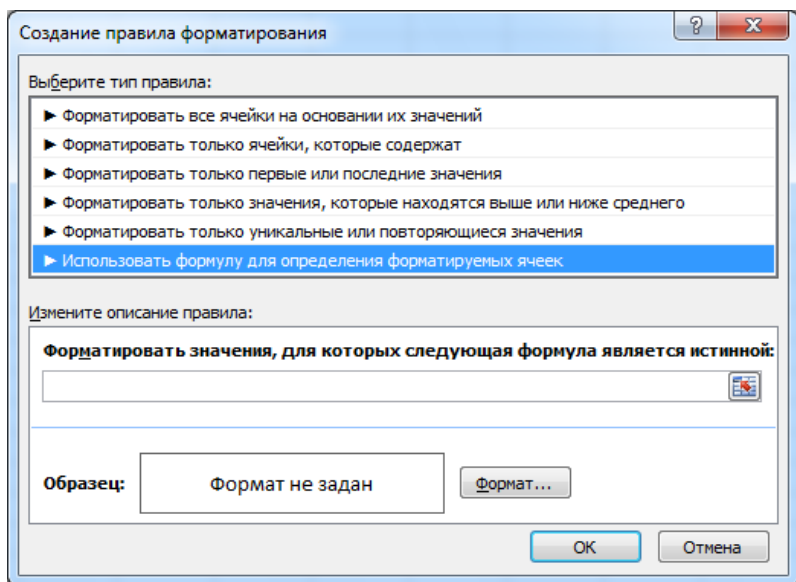


Рис. 2.55. ОД Создание правила форматирования по критерию **Формула**

3. В первое поле введите формулу. Формула должна принимать логическое значение Истина или Ложь.

4. Используя кнопку **Формат...**, задайте необходимое форматное оформление. Оформление применяется только в случае выполнения условий. Для закрытия окна нажмите **ОК**.

*Средства визуализации данных.* Помимо правил в MS Excel есть средства визуализации данных, помогающие понять и проиллюстри-

ровать изменения и сравнение значений: гистограммы, цветовые шкалы и наборы специальных значков.

**Гистограммы.** Немедленно отображают значения выделенных ячеек в виде сравниваемых полос значений. Для применения гистограмм: **Выделите ячейки** → Вкладка **Главная** → **Условное форматирование** → **Гистограммы** → выберите вид отображения (рис. 2.56).

**Цветовые шкалы.** Применяют выбранные цветовые схемы к определенному диапазону значений так, что отдельные ячейки отображаются определенным цветом в зависимости от их значений. Для применения цветовых шкал: **Выделите ячейки** → Вкладка **Главная** → **Условное форматирование** → **Цветовые шкалы** → выберите вид отображения.

**Наборы значков.** Применяются к ячейкам для отражения различных изменений и тенденций. Для применения наборов значков: **Выделите ячейки** → Вкладка **Главная** → **Условное форматирование** → **Наборы значков** → выберите подходящий набор значков (рис. 2.56). Весь диапазон ячеек делится на интервалы по числу значков в наборе и каждому интервалу соответствует определенный значок тенденции.

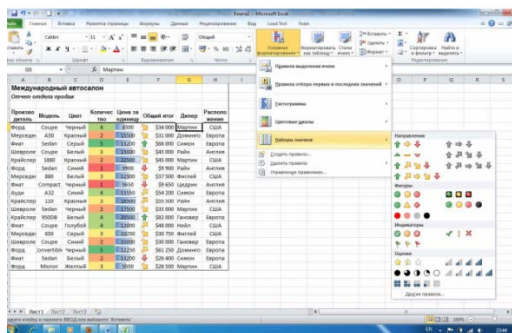


Рис. 2.56. Отображение диапазона с помощью средств визуализации

Для отмены условного форматирования: **Выделите ячейки** → Вкладка **Главная** → **Условное форматирование** → **Удалить правила** → выберите требуемый вариант **Удалить правила** из выделенных ячеек (или со всего листа).

**Примечания к ячейкам.** *Создание примечаний.* Для удобства коллективной работы или при наличии на листе сложных формул, требу-

ющих пояснений, к любой ячейке можно добавить текстовое или звуковое примечание. Для этого выполните следующие действия.

1. Укажите ячейку, к которой необходимо добавить примечание.
2. Вкладка **Рецензирование** → **Создать примечание**.
3. В появившееся поле введите необходимый комментарий (рис. 2.57).

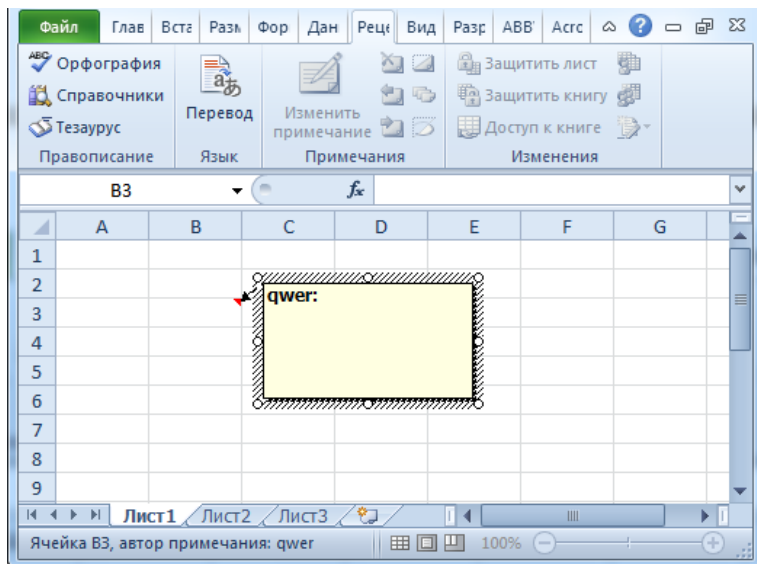


Рис. 2.57. Создание примечания

4. После завершения ввода щелкните **ЛКМ** вне поля комментария или дважды нажмите клавишу **ESC**.

Ячейке будет назначен комментарий, а в правом верхнем углу появится красный индикатор.

*Удаление примечания из ячейки.* В случае если примечание больше не нужно, его можно удалить из ячейки. Для этого выполните следующие действия.

1. Укажите ячейку, содержащую примечание, которое следует удалить.
2. Выполните одно из следующих действий:



– либо Вкладка **Главная** → группа **Редактирование** → кнопка **Очистить** → **Очистить примечания** → либо ПКМ на ячейке → **Удалить Примечание**.

**Работа со структурой данных списка.** *Введение в списки данных.*

MS Excel позволяет создавать базы данных непосредственно на рабочем листе.

Такая база данных представляет собой обычную таблицу, называемую списком. В MS Excel имеется набор средств, облегчающих обработку и анализ данных, содержащихся в списке. Например, для получения упорядоченной информации мы можем отсортировать список по 1, 2 или 3 столбцам. В случае необходимости можно скрыть не интересующую часть информации, используя фильтрацию. Для подсчета итогов по группам информации удобно воспользоваться возможностью подведения промежуточных итогов.

*Сортировка данных.* MS Excel позволяет выполнять простую сортировку (по одному столбцу) и сложную (по нескольким столбцам) данных в таблице-списке. Сортировка может осуществляться по алфавиту или по значениям в порядке убывания или возрастания.

MS Excel также определяет и исключает из сортировки заголовки столбцов.

*Простая сортировка данных.* Для быстрой сортировки данных списка по некоторому столбцу выполните следующие действия.

1. Выделите ячейку внутри сортируемого столбца.

2. Вкладка **Данные** → группа **Сортировка и фильтр** → нажмите кнопку **Сортировка по возрастанию** или **Сортировка по убыванию**.

Кнопки сортировки доступны также и на вкладке **Главная** в группе **Редактирование**.

Для отмены сортировки нажмите кнопку **Отменить** на панели быстрого доступа.

*Сложная сортировка данных.* Если необходимо отсортировать таблицу по нескольким столбцам, выполните следующие действия.

1. Выделите ячейку внутри сортируемого списка.

2. Вкладка **Данные** → группа **Сортировка и фильтр** → **Сортировка** MS Excel отобразит **ОД Сортировка** (рис. 2.58).

3. Используя кнопку списка в поле **Сортировать по**, выберите заголовков столбца, по которому будет производиться сортировка, выберите признак сортировки в поле **Сортировка** (значение, цвет) и режим сортировки в поле **Порядок** (по возрастанию, по убыванию).



2. Вкладка **Данные** → группа **Сортировка и фильтр** → **Фильтр**.
3. Нажмите **кнопку списка фильтра** (рис. 2.59) в столбце, по которому вы хотите фильтровать данные.
4. Выберите из списка элемент, в соответствие с которым вы хотите произвести фильтрацию.

| Производитель | Модель | Цвет    | Кол-во | Цена за единицу | Общий итог | Дилер   | Расположение |
|---------------|--------|---------|--------|-----------------|------------|---------|--------------|
| Ауди          | A32    | Синий   | 4      | 13550           | \$54 200   | Симон   | Европа       |
| Крайслер      | 1880   | Красный | 2      | 22500           | \$45 000   | Мартин  | США          |
| Крайслер      | 110    | Красный | 3      | 18500           | \$55 500   | Райн    | Англия       |
| Крайслер      | 950DB  | Белый   | 4      | 20500           | \$82 000   | Гановер | Европа       |

Рис. 2.59. Использование фильтра

После выполнения фильтрации вы также можете отфильтровать и результат, если необходимо.

MS Excel добавляет к **кнопке списка фильтра** специальный значок для определения отфильтрованных столбцов.

*Фильтрация с использованием критериев.* Для выполнения фильтрации на основе некоторых условий выполните следующие действия.

1. Выделите любую ячейку внутри списка.
2. Вкладка **Данные** → группа **Сортировка и фильтр** → **Фильтр**.
3. Нажмите на кнопку списка фильтра возле заголовка того столбца, по которому вы хотите произвести фильтрацию списка.

Выберите **Текстовые фильтры** (если в столбце текст) или **Числовые фильтры** (если в столбце числа), и затем критерий фильтрации (равно, не равно, больше и т. д.) MS Excel отобразит **ОД Пользовательский автофильтр** (рис. 2.60).

4. Нажмите кнопку списка в левом верхнем поле и выберите оператор сравнения, на основе которого будет производиться отбор данных, удовлетворяющих критерию.

5. В поле справа от поля оператора сравнения выполните одно из следующих действий:

- введите значение, которое вы хотите использовать при сравнении;
- нажмите на стрелку и выберите значение из списка, которое вы хотите использовать при сравнении.

6. Если необходимо, повторите шаги 5–6 для задания второго условия отбора данных и выполните одно из следующих действий:

- выберите **И**, если хотите, чтобы оба критерия сравнения применялись;
- выберите **ИЛИ**, если хотите применить отдельные критерии сравнения.

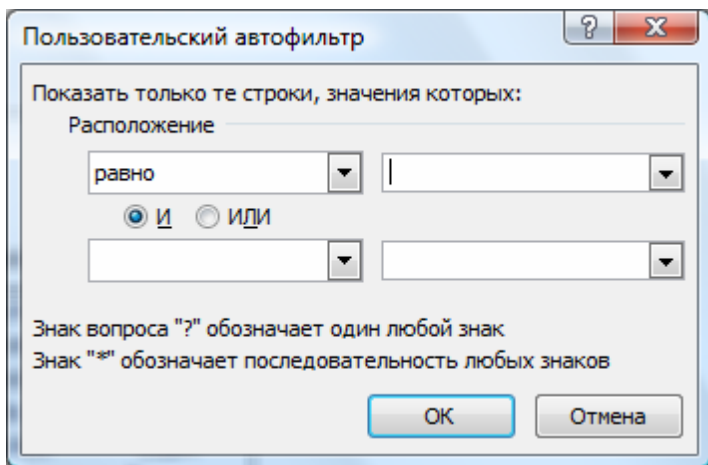


Рис. 2.60. Автофильтр **Условие**

7. Нажмите **ОК**.

MS Excel отобразит результат фильтрации данных на листе (рис. 2.61).

|    | A                       | B      | C       | D      | E               | F          | G       | H            |
|----|-------------------------|--------|---------|--------|-----------------|------------|---------|--------------|
| 1  | Международный автосалон |        |         |        |                 |            |         |              |
| 2  | Отчет отдела продаж     |        |         |        |                 |            |         |              |
| 3  |                         |        |         |        |                 |            |         |              |
| 4  | Производитель           | Модель | Цвет    | Кол-во | Цена за единицу | Общий итог | Дилер   | Расположение |
| 6  | Крайслер                | 1880   | Красный | 2      | 22500           | \$45 000   | Мартин  | США          |
| 7  | Крайслер                | 110    | Красный | 3      | 18500           | \$55 000   | Райн    | Англия       |
| 8  | Крайслер                | 950DB  | Белый   | 4      | 20500           | \$82 000   | Гановер | Европа       |
| 9  | Мерседес                | A50    | Красный | 2      | 15500           | \$31 000   | Доминго | Европа       |
| 21 | Шевролет                | Sedan  | Черный  | 2      | 17500           | \$35 000   | Мартин  | США          |

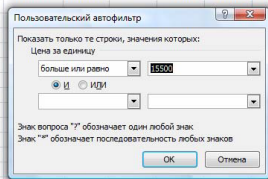


Рис. 2.61. Результат фильтрации данных с использованием **Критериев**

*Отображение всех записей.* Для отображения всех данных списка выполните следующие действия.

1. Нажмите на кнопку списка фильтра рядом с заголовком столбца, по которому производилась фильтрация (кнопка списка должна быть со специальным значком).

2. Из выпадающего списка выберите **Снять фильтр с «имя столбца»**.

Если вы использовали фильтрацию больше чем по одному столбцу, выберите из выпадающего списка команду **Снять фильтр с...** для каждого столбца.

*Удаление фильтра.* Для отключения фильтрации и удаления фильтра выполните следующие действия:

1. Вкладка **Данные** → группа **Сортировка и фильтр** → **Фильтр**.

MS Excel удалит **кнопки списка фильтра** рядом с каждым заголовком столбца в списке.

*Группировка данных.* При наличии одинаковых данных в ячейках столбца их можно сгруппировать.

Для этого выполните следующие действия.

1. Выделите ячейку в столбце, содержащем данные, которые вы хотите сгруппировать.

2. Нажмите кнопку **Сортировка по возрастанию** для сортировки столбца.

3. Выделите ячейки с одинаковыми данными для группировки.

4. Вкладка **Данные** → группа **Структура** → **Группировать** → **Группировать...**

Откроется ОД **Группирование**.

5. Выберите вариант **Сгруппировать строки** → **ОК**.

Слева от номеров строк появятся символы раскрытой структуры (значок –), при нажатии на него структура собирается и значок преобразуется в +.

Для снятия группирования выполните следующие действия.

1. Выделите все ячейки, входящие в группу.

2. Вкладка **Данные** → группа **Структура** → **Разгруппировать** → **Разгруппировать...**

Откроется ОД **Разгруппировать**.

3. Выберите вариант **Разгруппировать строки** → **ОК**.

Символы группирования исчезнут.

*Использование расширенного фильтра.* Для создания сложных условий отбора записей можно использовать расширенный фильтр.

Его особенностью является возможность создать отфильтрованный диапазон в другом месте листа. Для применения расширенного фильтра выполните следующие действия.

1. Создайте диапазон условий для будущего отбора в виде таблицы из двух строк: заголовков и критериев, например, как в строках 24 и 25 на рис. 2.62.

2. Выделите любую ячейку внутри списка.

3. Вкладка **Данные** → группа **Сортировка и фильтр** → **Дополнительно**.

Откроется **ОД Расширенный фильтр** (рис. 2.62).

4. В поле **Обработка** выберите, где будет расположен отфильтрованный список.

5. В окне **Исходный диапазон** введите диапазон исходного списка (при выполнении второго шага диапазон должен быть указан).

6. В окне **Диапазон условий** укажите адрес строк, в которых созданы условия отбора.

7. В окне **Поместить результат в диапазон** укажите адрес строк, где будет создан отфильтрованный список.

8. Нажмите **ОК**.

В указанном месте появится отфильтрованный список. Для примера на рис. 2.62. в результирующем списке останутся строки 7 и 8.

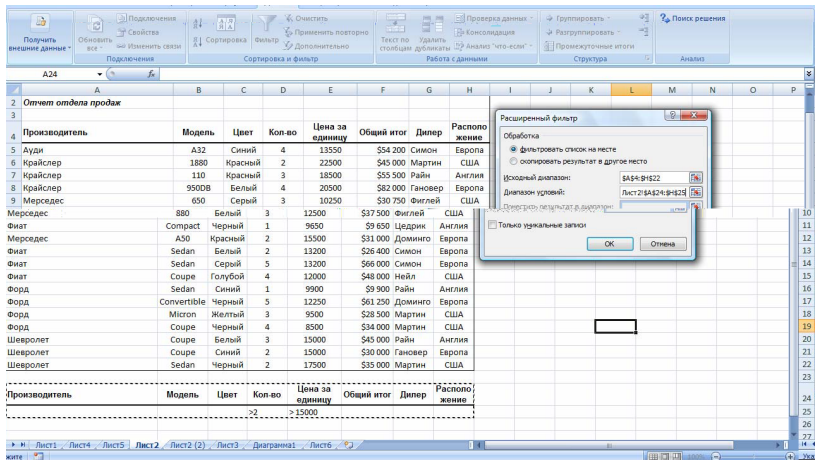


Рис. 2.62. Использование расширенного фильтра

### **Подведение итогов. Создание Итогов.**

Операция автоматического подведения промежуточных итогов в MS Excel не только производит вычисление итоговых сумм и их занесение на рабочий лист, но и одновременно создает на рабочем листе структуру, связанную с полученными итогами.

Для подведения простых промежуточных итогов выполните следующие действия.

1. Выделите ячейку в столбце, содержащем данные, для которого вы хотите подвести промежуточные итоги.

2. Нажмите кнопку **Сортировка по возрастанию** для сортировки столбца.

3. Вкладка **Данные** → группа **Структура** → **Промежуточные итоги**.

MS Excel отобразит **ОД Промежуточные итоги** (рис. 2.63).

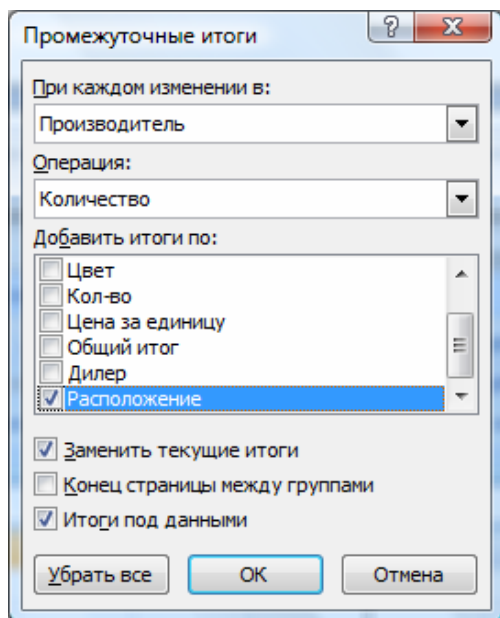


Рис. 2.63. ОД **Промежуточные итоги**

4. В поле **При каждом изменении в**, используя кнопку списка, выберите столбец, по которому происходила сортировка данных.

5. В поле **Операция**, используя кнопку списка, выберите функцию для расчета итоговых значений.

MS Excel использует по умолчанию функцию **Сумма** для числовых значений и функцию **Кол-во значений** для текста.

6. В списке **Добавить итоги по** установите флажки возле наименований тех столбцов, по которым будут производиться вычисления.

7. В случае необходимости снимите или установите флажки параметров.

8. Нажмите **ОК**.

Когда MS Excel отображает промежуточные итоги, он также отображает и символы иерархической структуры слева от рабочего листа. Вы можете управлять видом списка с промежуточными итогами, нажимая на символы иерархической структуры:

*Удаление промежуточных итогов.*

Для отображения списка в исходном виде необходимо удалить промежуточные итоги, выполните следующие действия.

1. Вкладка **Данные** → группа **Структура** → **Промежуточные итоги**.

2. Нажмите **Убрать все**.

MS Excel удалит все промежуточные итоги из списка.

**Создание Сводной таблицы.** *Понятие Сводной таблицы.* **Сводная таблица** – таблица, обеспечивающая фильтрацию данных по выбранным столбцам и подведение промежуточных итогов.

Приведенные ниже рисунки (рис. 2.64 и рис. 2.65) иллюстрируют таблицу-список MS Excel и результирующую **Сводную таблицу**.

|    |                         |             |         |        |                 |            |         |              |
|----|-------------------------|-------------|---------|--------|-----------------|------------|---------|--------------|
| 1  | Международный автосалон |             |         |        |                 |            |         |              |
| 2  | Отчет отдела продаж     |             |         |        |                 |            |         |              |
| 3  |                         |             |         |        |                 |            |         |              |
| 4  | Производитель           | Модель      | Цвет    | Кол-во | Цена за единицу | Общий итог | Дилер   | Расположение |
| 5  | Ауди                    | A32         | Синий   | 4      | 13550           | \$54 200   | Симон   | Европа       |
| 6  | Крайслер                | 1880        | Красный | 2      | 22500           | \$45 000   | Мартин  | США          |
| 7  | Крайслер                | 110         | Красный | 3      | 18500           | \$55 500   | Райн    | Англия       |
| 8  | Крайслер                | 950DB       | Белый   | 4      | 20500           | \$82 000   | Гановер | Европа       |
| 9  | Мерседес                | 650         | Серый   | 3      | 10250           | \$30 750   | Фиглей  | США          |
| 10 | Мерседес                | 880         | Белый   | 3      | 12500           | \$37 500   | Фиглей  | США          |
| 11 | Фиат                    | Compact     | Черный  | 1      | 9650            | \$9 650    | Цедрик  | Англия       |
| 12 | Мерседес                | A50         | Красный | 2      | 15500           | \$31 000   | Доминго | Европа       |
| 13 | Фиат                    | Sedan       | Белый   | 2      | 13200           | \$26 400   | Симон   | Европа       |
| 14 | Фиат                    | Sedan       | Серый   | 5      | 13200           | \$66 000   | Симон   | Европа       |
| 15 | Фиат                    | Coupe       | Голубой | 4      | 12000           | \$48 000   | Нейл    | США          |
| 16 | Форд                    | Sedan       | Синий   | 1      | 9900            | \$9 900    | Райн    | Англия       |
| 17 | Форд                    | Convertible | Черный  | 5      | 12250           | \$61 250   | Доминго | Европа       |
| 18 | Форд                    | Micron      | Желтый  | 3      | 9500            | \$28 500   | Мартин  | США          |
| 19 | Форд                    | Coupe       | Черный  | 4      | 8500            | \$34 000   | Мартин  | США          |
| 20 | Шевролет                | Coupe       | Белый   | 3      | 15000           | \$45 000   | Райн    | Англия       |
| 21 | Шевролет                | Coupe       | Синий   | 2      | 15000           | \$30 000   | Гановер | Европа       |
| 22 | Шевролет                | Sedan       | Черный  | 2      | 17500           | \$35 000   | Мартин  | США          |

Рис. 2.64. Исходная таблица



| Расположение      | Крайслер            | Мерседес             | Фiat                | Форд                 | Шевролет             | Общий итог           |
|-------------------|---------------------|----------------------|---------------------|----------------------|----------------------|----------------------|
| 110               | \$ 55 500,00        |                      |                     |                      |                      | \$ 55 500,00         |
| 500               |                     | \$ 30 750,00         |                     |                      |                      | \$ 30 750,00         |
| 880               |                     | \$ 37 500,00         |                     |                      |                      | \$ 37 500,00         |
| 1800              |                     | \$ 45 000,00         |                     |                      |                      | \$ 45 000,00         |
| 1900B             |                     | \$ 42 000,00         |                     |                      |                      | \$ 42 000,00         |
| A32               | \$ 54 200,00        |                      |                     |                      |                      | \$ 54 200,00         |
| A50               |                     | \$ 31 000,00         |                     |                      |                      | \$ 31 000,00         |
| Compact           |                     | \$ 9 650,00          |                     |                      |                      | \$ 9 650,00          |
| Convertible       |                     |                      | \$ 61 250,00        |                      |                      | \$ 61 250,00         |
| <b>Общий итог</b> | <b>\$ 54 200,00</b> | <b>\$ 182 500,00</b> | <b>\$ 99 250,00</b> | <b>\$ 150 050,00</b> | <b>\$ 133 650,00</b> | <b>\$ 729 650,00</b> |

Рис. 2.65. Результирующая Сводная таблица

*Создание Сводной таблицы.* Для создания сводной таблицы выполните следующие действия.

1. Установите курсор внутри списка, на основании которого вы хотите создать **Сводную таблицу**.

2. Вкладка Вставка → группа Таблицы → Сводная таблица... . MS Excel отобразит **ОД Создание сводной таблицы** (рис. 2.66).

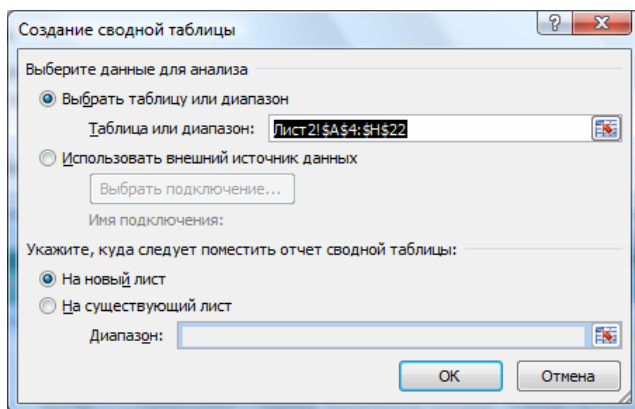


Рис. 2.66. ОД Создание сводной таблицы

3. Укажите диапазон ячеек для построения сводной таблицы и место, куда ее поместить. Нажмите кнопку **ОК**.

MS Excel отобразит в правой части окна **Список полей сводной таблицы** (рис. 2.67).

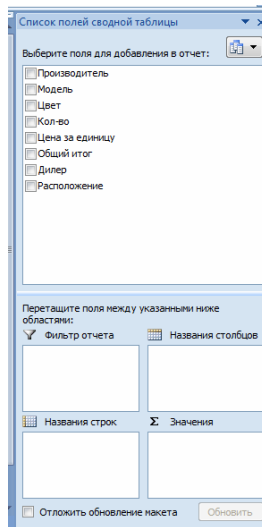


Рис. 2.67. Список полей  
Сводной таблицы

4. Удерживая нажатой ЛКМ, перетащите поле, по которому должна производиться **фильтрация**, в область **Фильтр отчета**. Слева на листе в верхнем углу появится выбранное поле.

5. Удерживая нажатой ЛКМ, перетащите поля, по которым должна производиться **группировка**, в область **Названия столбцов** и **Названия строк**. В областях строки и столбца появились поля.

6. Удерживая нажатой ЛКМ, перетащите поле, по которому должно производиться **подведение итогов**, в область **Σ Значения**. В области данных появились итоги.

Если требуется, примените соответствующий формат к ячейкам области данных (например, денежный).

#### 7. Сводная таблица построена.

Используя **Список полей сводной таблицы**, вы можете оперативно менять структуру, параметры и форматирование сводной таблицы.

**Удаление Сводной таблицы.** Для удаления Сводной таблицы, построенной на отдельном листе, – удалите лист.

Для удаления Сводной таблицы, построенной на существующем листе, – выделите все строки таблицы и удалите их.

**Графические возможности MS Excel.** MS Excel предоставляет возможность создания следующих графических объектов:

- линий и простых геометрических фигур;
- автофигур;
- объектов **WordArt**;
- рисунков;
- диаграмм.

*Графические объекты. Рисование линий и фигур.* Самыми простыми графическими объектами являются прямые линии и стандартные геометрические фигуры. Вы можете добавить в документ MS Excel любую фигуру, используя значки фигур на вкладке **Вставка** в группе **Иллюстрации** (рис. 2.68).

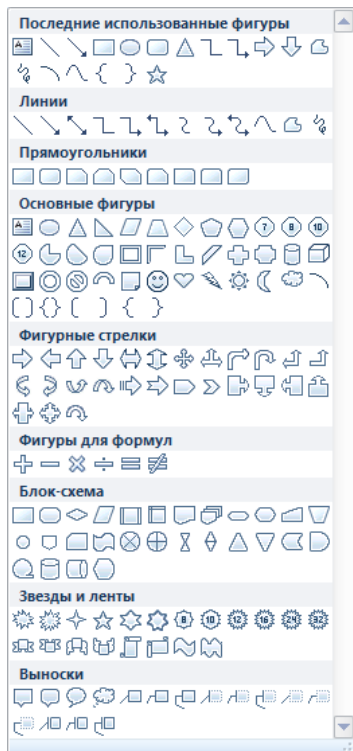


Рис. 2.68. ОД **Фигуры**

Для создания фигур выполните следующие действия

1. Выберите нужную фигуру в галерее фигур (рис. 2.68).
2. Переместитесь на лист MS Excel и, удерживая нажатой **ЛКМ**, нарисуйте объект.

Если вы будете рисовать объекты, удерживая нажатой клавишу **SHIFT**, вы получите правильные фигуры: квадрат, круг. По завершении рисования отпустите сначала **ЛКМ**, потом **SHIFT**.

3. Отпустите **ЛКМ**.

На ленте добавится новая вкладка **Формат**. С помощью кнопок на этой вкладке вы можете отредактировать графический объект: повернуть под желаемым углом (группа **Упорядочить**), задать цвет заливки, тип и цвет линии, тень, объем (группа **Стили фигур**), размер (группа **Размер**). Для этого его следует выделить.

*Выделение объектов.* Для большинства операций работы с объектом его требуется выделить. Для этого выполните следующие действия.

1. Подведите УМ к объекту и щелкните **ЛКМ**. Вокруг объекта появятся белые маркеры.
2. Для того чтобы отказаться от выделения, щелкните **ЛКМ** на ячейке.

*Добавление объекта WordArt.* Объект **WordArt** – это красиво оформленный текст. Выполните следующие действия, чтобы создать объект WordArt.

1. Вкладка **Вставка** → группа **Текст** → **WordArt**.
2. Выберите стиль вашей будущей надписи из предложенных вариантов.
3. Введите текст, который вы хотите оформить.
4. Перейдите на любую ячейку.

Для дальнейшего редактирования либо выделите объект → **ПКМ** → **Формат фигуры...**, либо используйте кнопки появившейся вкладки **Формат**.

*Клип.* Коллекция клипов включает большой набор рисунков, фильмов, фотографий, позволяющих оформить документ профессионально сделанными изображениями.

Коллекция содержит разнообразные по стилю рисунки: от живописного фона до карикатур. Все объекты разделены по категориям и описываются с помощью набора ключевых слов, что значительно облегчает их поиск.

Для вставки картинки выполните следующие действия.

1. Переместите курсор в то место, куда вы хотите поместить картинку.

2. Вкладка **Вставка** → **Клип**. ОЗ откроется на разделе **Клип**.

3. Для поиска нужной картинки воспользуйтесь областью **Искать** → введите ключевое слово → нажмите кнопку **Начать**.

4. Выберите требуемую картинку.

Используйте кнопку списка рядом с картинкой для настройки параметров рисунка и управления содержанием коллекций.

Щелкнув на ссылке **Упорядочить клипы...**, вы можете выбрать в ОД **Организатор клипов** (рис. 2.69) контекстные картинки:

1. В **Списке Коллекций** выберите **Коллекции Microsoft Office** → выберите категорию → если необходимо подкатегорию → щелкните по изображению ПКМ → **Копировать**.

2. Переместитесь на лист MS Excel. Выберите меню **Правка** → **Вставить**. Картинка появится в том месте, где находится курсор.

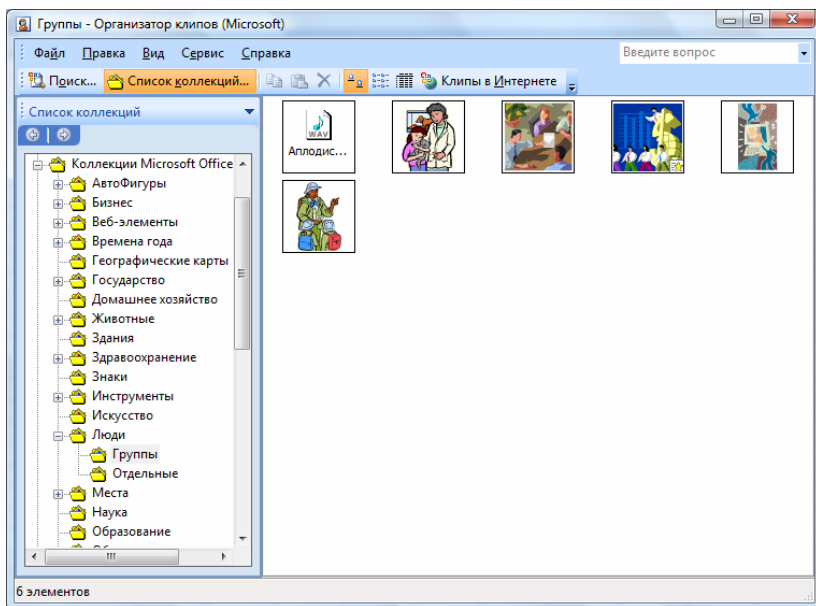


Рис. 2.69. Окно **Организатор клипов**

### Удаление фигур.

1. Выделите графический объект.
2. Нажмите клавишу **DELETE** на клавиатуре.

### Создание основной диаграммы.

MS Excel предоставляет удобные средства для построения диаграмм, с помощью которых вы можете превратить «сухие» табличные данные рабочего листа в привлекательные, профессиональные, хорошо оформленные диаграммы.

Диаграмма может размещаться на листе, содержащем исходные данные, либо на отдельном листе. При этом, в случае размещения диаграммы на существующем листе, с ней можно работать как с картинкой.

*Создание Диаграммы.* Для создания диаграммы выполните следующие действия.

1. Выделите диапазон ячеек, на основании которого вы будете строить диаграмму.

2. На вкладке **Вставка** в группе **Диаграммы** нажмите на нужный вам тип диаграммы (Гистограмма, График, Круговая и т. д.) и выберите конкретный вид вашей диаграммы.

Появится первоначальный вариант диаграммы (например, как на рис. 2.70).

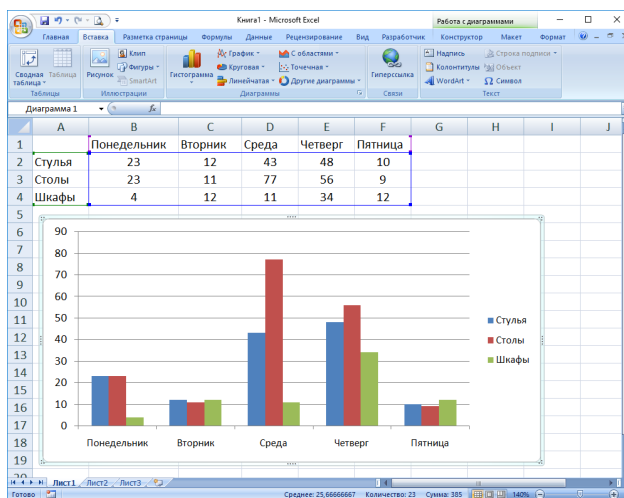


Рис. 2.70. Первоначальный вариант диаграммы

3. На вкладке **Конструктор** выберите подходящий стиль диаграммы, щелкая по значкам в группе **Стили диаграмм**.

4. В группе **Данные** проверьте правильность задания диапазонов данных: нажмите на кнопку **Выбрать данные**. Откроется ОД **Выбор источника данных** (рис. 2.71).

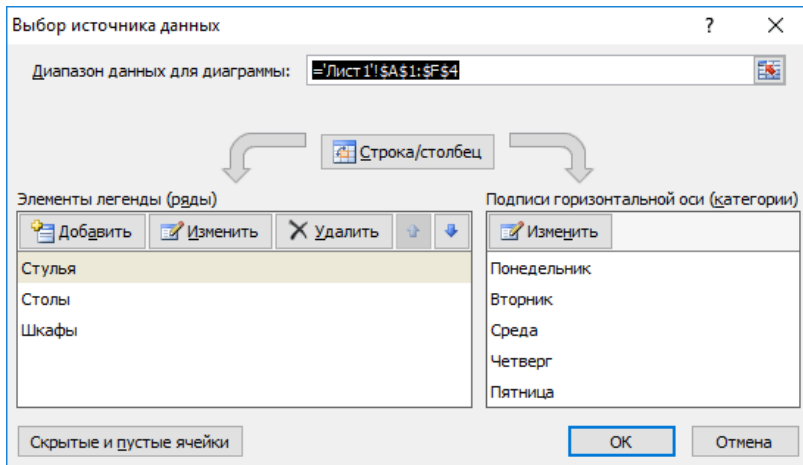


Рис. 2.71. Окно диалога **Выбор источника данных**

В случае необходимости вы можете добавить/удалить ряды данных с помощью соответствующих кнопок.

5. Для подписи данных по горизонтальной оси нажмите кнопку **Изменить** в правом окне (рис. 2.72). Откроется ОД **Подписи оси**.

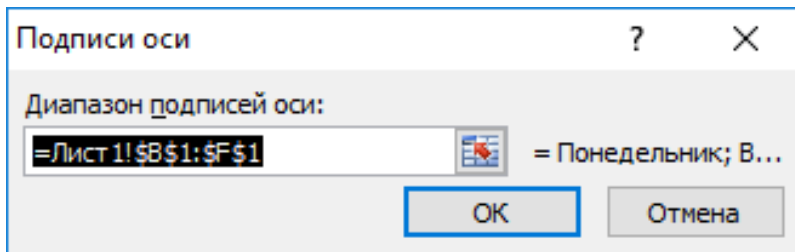


Рис. 2.72. Окно диалога **Подписи оси**

В окне **Диапазон подписей осей** укажите диапазон ячеек, содержащих данные для подписи по оси. Нажмите **ОК** → **ОК**.

6. Перейдите на вкладку **Макет**. В группе **Подписи** нажмите кнопку **Название диаграммы** и выберите вариант расположения названия. В появившемся объекте надпись **Название диаграммы** введите название своей диаграммы.

7. Для добавления названий осей нажмите на вкладке **Макет** кнопку **Названия осей** и выберите варианты названий горизонтальной и вертикальной осей. В появившихся объектах надпись **Название оси** введите названия осей своей диаграммы.

8. Для добавления легенды нажмите на вкладке **Макет** кнопку **Легенда** и выберите место расположения легенды.

9. Для добавления подписей данных нажмите на вкладке **Макет** кнопку **Подписи данных** и выберите место расположения подписей данных.

10. Для добавления таблицы данных к диаграмме нажмите на вкладке **Макет** кнопку **Таблица данных** и выберите вариант расположения таблицы данных.

11. Для форматирования и изменения разметки осей и сетки диаграммы воспользуйтесь кнопками **Оси** и **Сетка** в группе **Оси**.

12. Для определения местоположения диаграммы на вкладке **Конструктор** в группе **Расположение** нажмите кнопку **Переместить диаграмму**. Откроется **ОД**.

**Перемещение диаграммы.** В окне выполните одно из следующих действий:

- либо выберите переключатель **на отдельном листе** и введите имя нового листа, если диаграмма должна располагаться на новом рабочем листе;

- либо выберите переключатель на **имеющемся листе** и, используя кнопку списка, укажите необходимый лист в рабочей книге, если диаграмма должна располагаться на одном из существующих листов.

13. Нажмите **ОК**. MS Excel поместит вашу диаграмму в выбранное место.

Пример готовой диаграммы приведен на рис. 2.73.

На вкладке **Вставка** вы можете быстро изменить тип диаграммы с помощью кнопок в группе **Диаграммы**.

Все типы диаграмм доступны на вкладке **Конструктор** в группе **Тип** с помощью кнопки **Изменить тип диаграммы**.



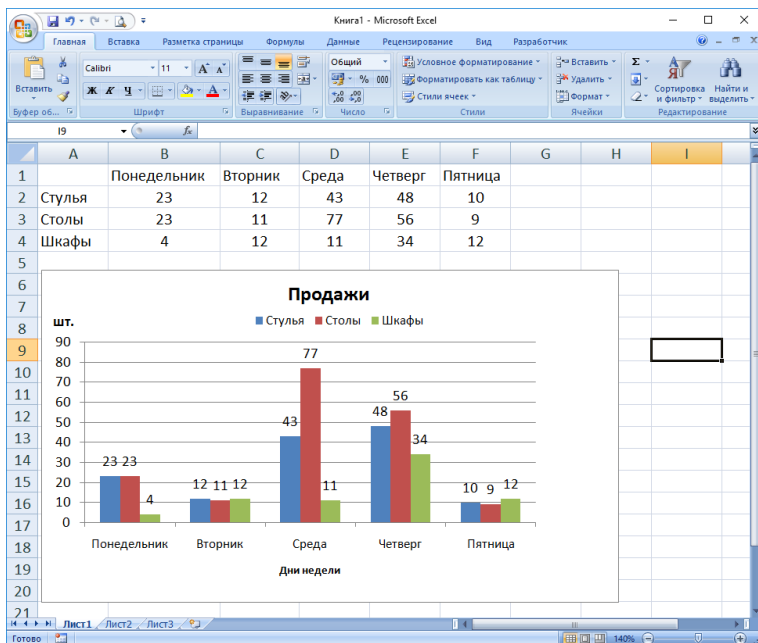


Рис. 2.73. Пример построения диаграммы на существующем рабочем листе

После того как вы создали диаграмму, можете перемещать ее и менять размеры.

При позиционировании УМ над элементами диаграммы появляются всплывающие подсказки. Используйте их для выделения нужного элемента диаграммы.

После создания диаграммы автоматически появляются **вкладки Конструктор, Макет, Формат**. Эти вкладки содержат инструментальные средства, позволяющие изменять характеристики диаграммы.

Для быстрого построения диаграммы с установками, принятыми по умолчанию, выполните следующее: выделите данные → нажмите **F11**.

**Удаление диаграммы.** Для удаления Диаграммы, построенной на отдельном листе, – удалите лист.

Для удаления Диаграммы, построенной на существующем листе, выполните действия.

1. Щелкните ЛКМ на **Область диаграммы**.
2. Нажмите на клавиатуре **Delete**.

**Функция в Excel: поиск решения.** Наряду со множеством других возможностей, в Microsoft Excel есть одна малоизвестная, но очень полезная функция под названием «Поиск решения». Несмотря на то, что найти и освоить ее, может быть, непросто, ее изучение и применение может помочь в решении огромного количества задач. Функция берет данные, перебирает их и выдает самое оптимальное решение из возможных.

Для того чтобы надстройка **Поиск решения** загружалась сразу при запуске Excel, выполните следующие действия.

1. Нажмите кнопку **Office**. В открывшемся меню нажмите кнопку **Параметры Excel**. Откроется диалоговое окно **Параметры Excel**.

2. В открывшемся окне выберите пункт **Надстройки**.

3. В поле **Управление** из раскрывающегося списка выберите **Надстройки Excel**. Нажмите кнопку **Перейти**. Откроется диалоговое окно **Надстройки**.

4. В списке **Доступные надстройки** установите флажок напротив надстройки **Поиск решений**. Нажмите кнопку **ОК**.

Создайте таблицу (рис. 2.74).

|   | A     | B       | C    | D       |
|---|-------|---------|------|---------|
| 1 | Товар | Закупка | Цена | Затраты |
| 2 | Мечта | 192     | 7,1  | 1363,2  |
| 3 | Марс  | 130     | 5,2  | 676     |
| 4 | Люкс  | 96      | 4,8  | 460,8   |
| 5 | Итого |         |      | 2500    |
| 6 |       |         |      |         |

Рис. 2.74. Образец задания

В таблице необходимо подобрать объемы закупки товаров (подобрать значения ячеек **B2:B4**), чтобы при известных ценах (значения в ячейках **C2:C4**) общая сумма затрат (результат в ячейке **D5**) составила 2 500 руб. При этом товара «Марс» необходимо купить не менее 120, а товара «Мечта» в два раза больше, чем товара «Люкс». Эта задача решается с помощью надстройки **Поиск решения**.

Создадим таблицу (рис. 2.75).

|   | A     | B       | C    | D       |
|---|-------|---------|------|---------|
| 1 | Товар | Закупка | Цена | Затраты |
| 2 | Мечта |         | 7,1  | 0       |
| 3 | Марс  |         | 5,2  | 0       |
| 4 | Люкс  |         | 4,8  | 0       |
| 5 | Итого |         |      | 0       |

Рис. 2.75. Образец таблицы

1. Введем названия исходных данных. Укажем значения в ячейках **C2:C4**.

2. В ячейку **D2** введем формулу **=B2\*C2**.

3. Аналогично определим формулы для ячеек **D3** и **D4**.

4. В ячейке **D5** введем **=сумм(D2:D4)**.

Перейдем к поиску решения. В данном случае целью будет достижение заданного значения в ячейке **D5** (общая сумма затрат составляет 2 500 руб.), а условиями: **B3>=120** (товара «Марс» необходимо купить не менее 120), **B2=2\*B4** (товара «Мечта» необходимо купить в два раза больше, чем товара «Люкс»).

1. Выделите ячейку **D5**. Выполните команду **Данные/(Анализ)Поиск решения**. Откроется диалоговое окно **Поиск решения**.

2. В поле **Установить целевую ячейку** укажите **D5**.

Для целевой ячейки можно выбрать одно из следующих действий:

- чтобы максимизировать значение путем изменения значений влияющих ячеек, установить переключатель в положение **максимально-му значению**;

- чтобы минимизировать значение конечной ячейки путем изменения значений влияющих ячеек, установить переключатель в положение **минимальному значению**;

- чтобы установить значение в конечной ячейке равным некоторому числу, установить переключатель в положение **значению** и ввести в соответствующее поле требуемое число.

В данном примере установите флажок в положении **значению** и введите число **2 500**.

3. В поле **Изменяя ячейки** следует указать, какие ячейки влияют на результат вычислений в целевой ячейке. В нашем примере это ячейки **B2:B4**.

*Добавление ограничений.*

1. Нужно нажать кнопку **Добавить**. Откроется диалоговое окно **Добавление ограничения**.

2. В поле **Ссылка на ячейку** указать ячейку **B3**.

3. Затем в раскрывающемся списке выбрать знак ограничения **>=**.

4. В поле **Ограничение** введите 120 (рис. 2.76).

5. Для перехода к установке другого ограничения в окне **Добавление ограничения** следует нажать кнопку **Добавить**. Аналогично добавьте следующее ограничение  $B2=2*B4$ .

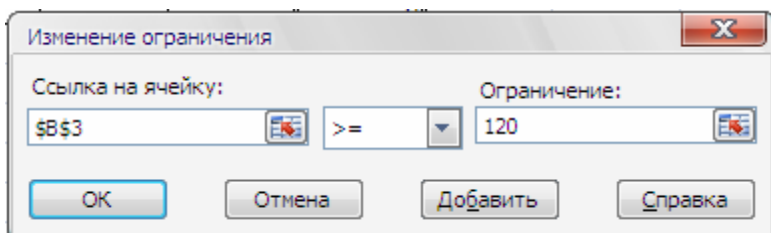


Рис. 2.76. Изменения ограничений

Кроме того, значения в ячейках B2:B4 должны быть целыми положительными числами:  **$B2:B4=\text{целое}$  и  $B2:B4 \geq 0$** .

Для завершения установки ограничений нужно нажать кнопку **ОК**.

В открывшемся окне **Поиск решения** отобразится список установленных ограничений (рис. 2.77).

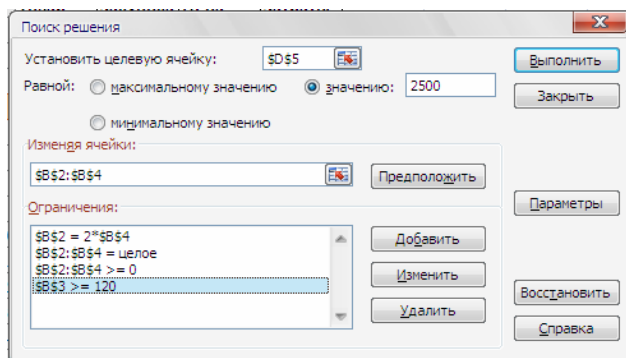


Рис. 2.77. Поиск решения

6. В окне **Поиск решения** нажмите кнопку **Выполнить**.

7. В окне **Результаты поиска** решения будет указано на наличие решения.

В случае если задача имеет корректное решение, на листе в соответствующие ячейки будут введены найденные значения.

8. Для сохранения найденного решения на листе следует установить переключатель **Сохранить найденное решение** и нажать кнопку **ОК** (рис. 2.78).

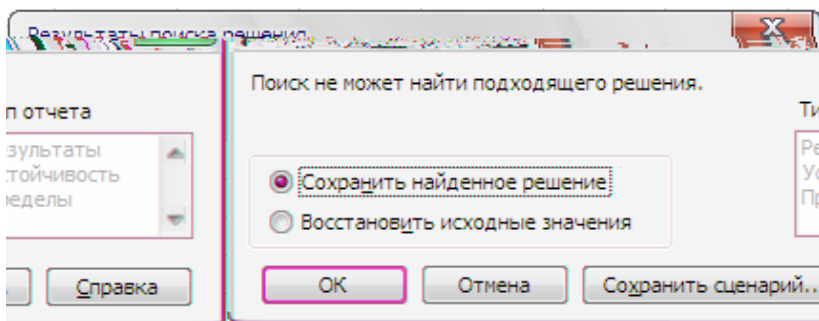


Рис. 2.78. Результаты поиска решения

*Изменение ограничений.* При решении какой-либо задачи иногда требуется экспериментировать с различными ограничениями. С помощью надстройки **Поиск решения** это можно осуществить. Для этого проделайте следующие действия.

1. Выполните команду **Данные/(Анализ)Поиск решения**. Откроется диалоговое окно **Поиск решения**.

2. В списке имеющихся ограничений выделите нужное ограничение и нажмите кнопку **Изменить**. Откроется диалоговое окно **Изменение ограничений**.

3. Измените ограничения и нажмите кнопку **ОК**. Снова откроется диалоговое окно **Поиск решения**.

4. Нажмите кнопку **Выполнить**.

*Удаление ограничений.* Чтобы удалить ограничение, выполните следующие действия.

1. Выполните команду **Данные/(Анализ)Поиск решения**. Откроется диалоговое окно **Поиск решения**.

2. В списке имеющихся ограничений выделите нужное ограничение и нажмите кнопку **Удалить**.

**Печать рабочей книги. Подготовка и печать рабочего листа. Предварительный просмотр перед печатью.**

1. Переместитесь на лист, который вы хотите распечатать.
2. Меню **Файл** → **Печать**.
3. Используя окно документа, просмотрите его содержимое.

*Настройка параметров печатной страницы.* Для настройки параметров печатной страницы выполните следующие действия.

1. Переместитесь на лист, который вы хотите распечатать.
2. Вкладка **Разметка страницы** → группа **Параметры страницы** → **Размер** → **Другие размеры страниц...**

MS Excel отобразит ОД **Параметры страницы** (рис. 2.79).

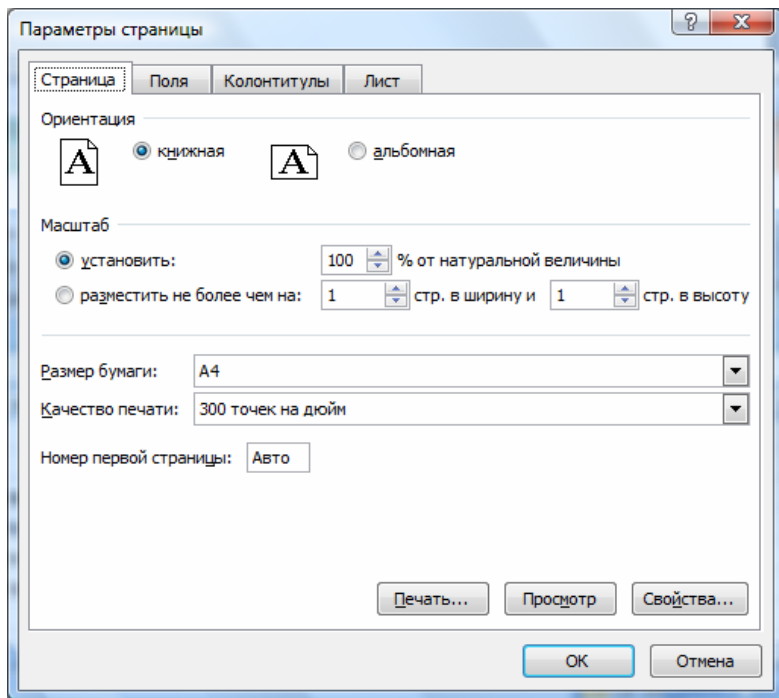


Рис. 2.79. Окно диалога **Параметры страницы**:  
вкладка **Страница**

3. На вкладке **Страница** произведите необходимые настройки.
4. Перейдите на вкладку **Поля** и установите размеры полей печатного документа.
5. Перейдите на вкладку **Колонтитулы** и создайте **Верхний колонтитул** или **Нижний колонтитул**, используя соответствующие кнопки (рис. 2.80).

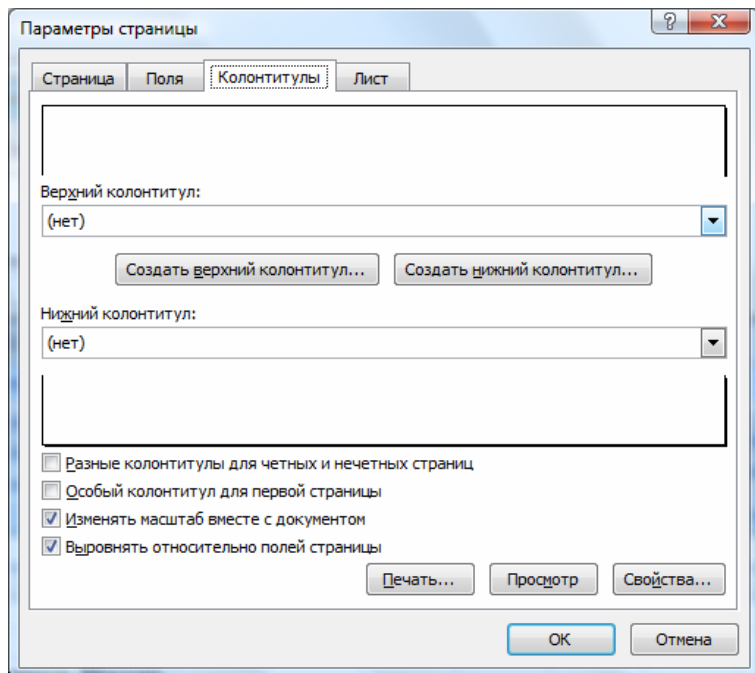


Рис. 2.80. ОД Параметры страницы: вкладка Колонтитулы

Вы можете создать дополнительные колонтитулы с помощью кнопок **Создать верхний колонтитул...** и **Создать нижний колонтитул...**.

На вкладке **Лист** (рис. 2.81), вы можете определить следующие параметры: диапазон печати, задать печать сетки рабочего листа MS Excel и заголовков строк и столбцов, примечаний, а также последовательность вывода на печать страниц, поставив галочку в поле **Последовательность вывода страниц**.

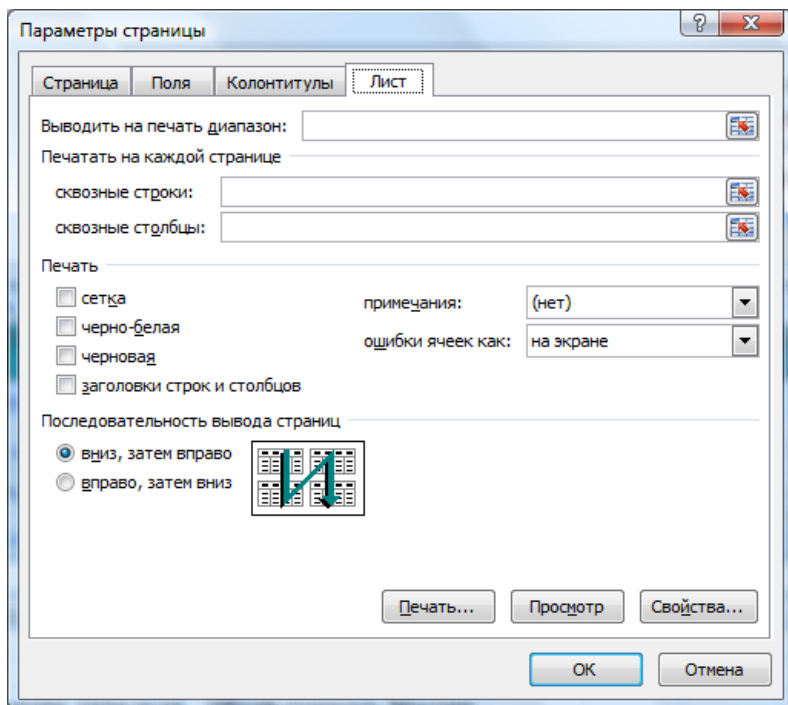


Рис. 2.81. ОД **Параметры страницы**: вкладка **Лист**

*Печать.* Для того чтобы напечатать данные из программы MS Excel, нужно установить принтер, который вы собираетесь использовать. Принтер должен быть включен в электросеть и подключен к вашему компьютеру или доступен по сети, если вы используете сетевой принтер.

Для печати данных MS Excel выполните следующие действия.

1. Меню **Файл** → **Печать** → **Печать**.
2. Произвести необходимую настройку в ОД **Печать** (рис. 2.82).
3. Нажмите **ОК**.



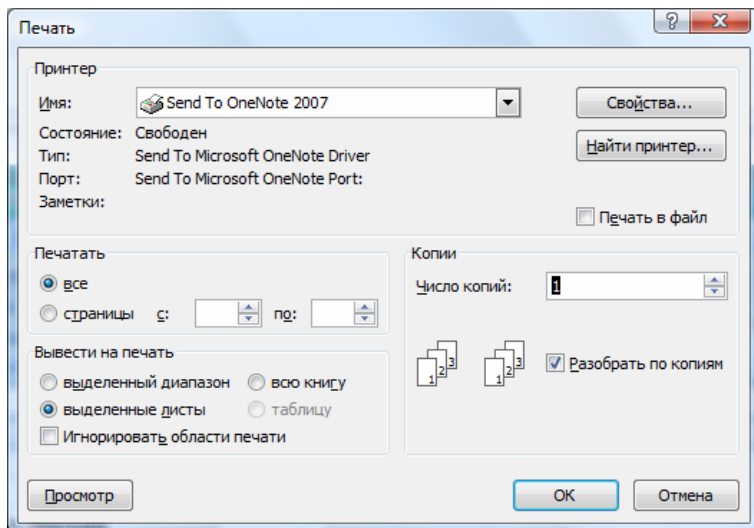


Рис. 2.82. Окно диалога **Печать**

**Связь Excel с другими приложениями.** Приложения пакета Microsoft Office поддерживают несколько способов обмена данными. Понятие обмен данными у пользователей, знакомых с Windows, как правило, ассоциируется с аббревиатурой OLE. Можно сказать, что использование технологии OLE (Object Linking and Embedding – связывание и внедрение объектов) является одним из наиболее ярких примеров реализации концепции обмена данными между приложениями Microsoft Office. Однако не следует забывать и о буфере обмена, с помощью которого также можно производить обмен данными. Этот способ является самым простым и наиболее распространенным, так как многие пользователи имеют представление только о нем.

**Буфер обмена.** В Windows обмен данными между приложениями осуществляется в зарезервированной области памяти, называемой буфером обмена. Для занесения данных в буфер обмена применяется команда **Главная** → **Копировать** или **Главная** → **Вырезать**, а для их вставки в нужное место документа – команда **Главная** → **Вставить**. Пользователи Windows привыкли к тому, что операция копирования или вырезания, выполняемая в любом Windows-приложении, приводит к замене содержимого буфера обмена. Во многих случаях это сопровождается определенными неудобствами. В пакете Microsoft Office

существует собственный буфер обмена, в который можно скопировать до 24 блоков информации (объектов). При этом блок, занесенный в буфер первым, не исчезает в результате помещения туда второго информационного блока. В данном случае речь идет не о буфере обмена системы Windows, а о так называемом офисном буфере обмена.

Доступ к офисному буферу обмена обеспечивается из всех приложений Microsoft Office, т. е., работая с этими приложениями, можно помещать объекты в буфер и копировать их оттуда. Объекты размещаются в буфере в той последовательности, в которой они туда заносятся. Копировать же их из буфера можно в любом порядке. Даже если объект помещается в буфер обмена Windows из приложения, которое не входит в пакет Microsoft, он все равно автоматически становится доступным в офисном буфере обмена. Если в буфере обмена уже содержатся 24 объекта, но пользователь вырезает или копирует еще один, данный объект помещается в буфер, а первый из числа там находящихся (скопированный раньше других) удаляется из буфера обмена, чтобы освободить место.

**Работа с фрагментами.** В Excel существует возможность сохранять или передавать данные в виде фрагментов документа. Фрагмент представляет собой отдельный файл, который можно сохранить на жестком диске.

Для того чтобы создать фрагмент, сначала необходимо выделить часть таблицы, которая должна стать его содержимым, щелкнуть на ней правой кнопкой мыши и в появившемся меню выбрать команду **Копировать**. Затем с помощью проводника Windows нужно найти папку, где этот фрагмент таблицы будет сохранен в виде файла, щелкнуть на ней правой кнопкой мыши и в появившемся контекстном меню выбрать команду **Вставить**. При необходимости вставить данный фрагмент в любое другое приложение Windows операции копирования и вставки следует повторить. Преимущество использования фрагментов по сравнению с применением буфера обмена состоит в том, что при выключении компьютера содержимое буфера обмена теряется, а фрагменты остаются на жестком диске до тех пор, пока не будут удалены пользователем.

**Связывание и внедрение данных.** При использовании технологии OLE данные Excel вставляются в другие файлы, в том числе и в Word-документы как объекты; связь с приложением-сервером при этом сохраняется. Чтобы запустить приложение-сервер и получить возможность обработать с его помощью объект, достаточно выполнить двой-

ной щелчок на данном объекте. После завершения работы с приложением-сервером объект в приложении-клиенте обновляется. Конечно же, для этого необходимо, чтобы приложение-клиент поддерживало технологию OLE. Приложения, поддерживающие технологию OLE, подразделяются, как вы теперь понимаете, на приложения-серверы и приложения-клиенты. Приложения-серверы предоставляют определенный объект в распоряжение приложения-клиента. Например, созданные в программе Paint рисунки могут быть вставлены в документ Excel, однако поместить Excel-таблицу в документ, созданный с помощью приложения Paint, невозможно. Таким образом, Excel может выступать и в качестве приложения-сервера, и качестве приложения-клиента, а Paint – только в качестве приложения-клиента. OLE-объектами могут выступать графические изображения, текст, таблицы, аудио- и видеофайлы.

## 2.4. Системы подготовки презентаций

Программа PowerPoint предназначена для создания экранных презентаций. Она позволяет не только создавать классические слайды для проектора, но также строить электронные презентации с включением всех возможностей мультимедиа и компьютерных сетей, включая Интернет.

Основные возможности PowerPoint:

- размещение необходимой информации (текстовой, графической, табличной) на слайдах;
- форматирование информации;
- создание анимации текстов и рисунков, добавление переходов от одного слайда к другому;
- создание гиперссылок и управляющих кнопок;
- подготовка материалов для раздачи слушателям;
- создание презентаций для использования в Интернете.

**Основные термины.** *Автофигура* – набор готовых фигур, включающий в себя такие фигуры, как овал, прямоугольник, фигурные стрелки, выноски и др.

*Анимация* – звуковое или визуальное оформление, которое можно добавить к тексту или другому объекту.

*Атрибут* – параметр текста или объекта, который может быть изменен с помощью средств рисования или команд меню.

*Интерактивные презентации* – построены на диалоге пользователя с компьютером с предоставлением пользователю возможностей самому находить нужную ему информацию.

*Линейные презентации* – в них материал расположен по порядку: начало – продолжение – завершение.

*Непрерывные презентации* – своеобразные рекламные ролики. Представляют собой завершённые информационные продукты, широко используемые на экспозициях.

*Презентации со сценариями* – предполагают показ слайдов, снабженных анимированными объектами, видеоматериалом, звуковым сопровождением и спецэффектами.

*Область заметок* – область в обычном режиме для ввода заметок к слайдам, распечатываемых в режиме страниц заметок или отображаемых на экране в режиме web-страницы.

*Страницы заметок* – распечатанные страницы с авторскими заметками, расположенными под соответствующими им слайдами.

*Образец слайдов* – слайд, содержащий данные по применённому шаблону оформления, включая параметры шрифтов, размеры и размещение рамок, параметры фона и цветовые схемы.

*Объект* – любая фигура, нарисованная в PowerPoint, а также любая графика или рисунок, преобразованные в формат PowerPoint.

*Прототип* – рамка с пунктирным контуром, которая отображается на создаваемом слайде и обозначает место для размещения таких объектов, как заголовок слайда, текст, таблица, диаграмма, рисунок и др.

*Рамки (местозаполнители)* – поля с пунктирными границами, являющиеся частью макетов большинства слайдов.

*Произвольный показ* – позволяет создавать презентацию внутри другой презентации, т. е. можно сгруппировать отличающиеся слайды одной презентации, предназначенные для разных аудиторий, присвоить им групповое имя и переходить к ним во время показа слайдов.

*Управляющие кнопки* – готовые кнопки, которые можно добавить в презентацию и назначить для кнопки гиперссылку.

*Цветовая схема* – это набор из восьми гармонично подобранных цветов, используемых в презентации. Кроме того, цветовые схемы можно применять в диаграммах и таблицах, а также для перекрашивания вставляемых в слайды рисунков.

*Шаблоны оформления* – файл, содержащий стили презентации, включая типы и размеры маркеров и шрифтов, размеры и положение рамок, параметры оформления фона, цветовые схемы и т. д.

### Основные приемы работы.

После запуска PowerPoint 2007 открывается основное окно приложения. Внешний вид этого экрана (рис. 2.83) несколько отличается от вида рабочих экранов Word 2007 и Excel 2007, которые были рассмотрены нами раньше. Хотя имеются и общие черты. К одной из таких черт относится «ленточный» интерфейс, в котором все команды программы размешены на ленте, разбитой на вкладки, а те, в свою очередь, на группы, включающие отдельные команды.

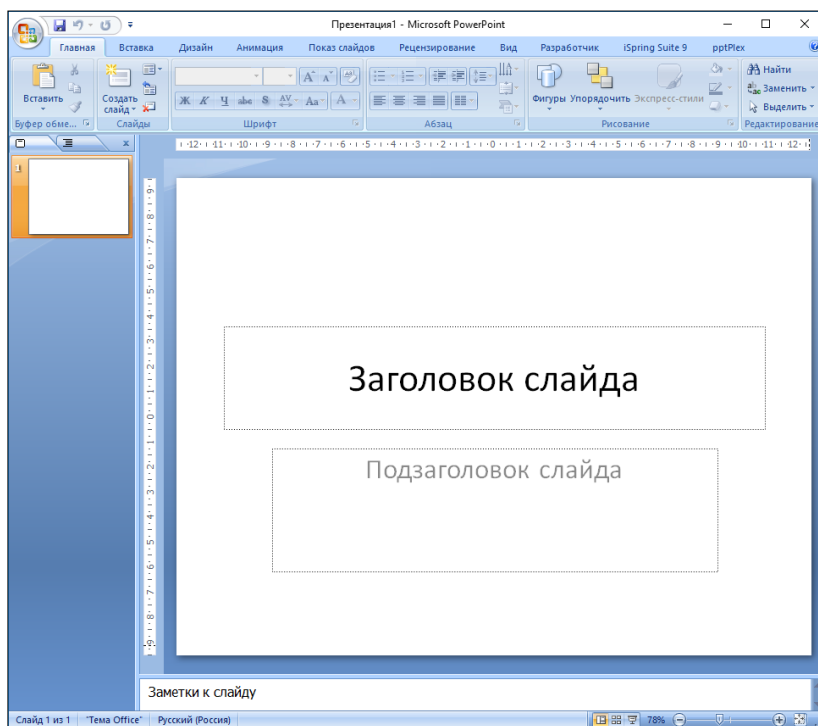


Рис. 2.83. Рабочее окно PowerPoint 2007

Лента содержит семь вкладок: **Главная**, **Вставка**, **Дизайн**, **Анимация**, **Показ слайдов**, **Рецензирование** и **Вид**. На каждой вкладке связанные общим назначением инструменты отображаются в виде групп. В нижней части окна расположена строка состояния. В левой

части строки отображается номер текущего слайда и количество слайдов в презентации, в правой части – кнопки переключения режимов отображения презентации и изменения масштаба.

В расположенной слева панели просмотра выбирается режим просмотра слайдов PowerPoint 2007. По умолчанию выбран режим **Слайды**, его можно изменить на режим **Структура**.

В основной части рабочего экрана отображается содержимое слайда, выбранного в правой панели. После первого запуска PowerPoint 2007 здесь отображается следующее: заголовок слайда, подзаголовок слайда и заметки к слайду.

**Создание презентации.** Создание презентации в PowerPoint можно выполнить тремя способами:

1) создание презентации на основе шаблона – самый легкий способ создания новой презентации. Позволяет применять заранее разработанные структуры, цветовые палитры и фон для создания набора слайдов по стандартным типам презентации;

2) создание презентации с использованием шаблона оформления;

3) создание новой пустой презентации открывает пустой холст для создания слайда и предоставляет больше свободы выбора пользователю.

**Режимы работы.** В Microsoft PowerPoint существует несколько основных режимов просмотра презентации: обычный, сортировщик слайдов, страницы заметок и показ слайдов. Переключение режимов просмотра осуществляется с помощью инструментов вкладки **Вид** или кнопок в строке состояния. Для удобства работы над презентациями существуют разные режимы.

**Обычный режим.** В этом режиме создаются и корректируются слайды, на которых можно расположить текст, графику или создать диаграммы.

**Сортировщик слайдов.** Данный режим позволяет определить очередность появления слайдов на экране, а также эффекты при переходе от слайда к слайду в процессе презентации.

**Режим страниц с заметками.** В этом режиме видны заранее подготовленные заметки к каждому слайду. Заметки оказывают помощь во время доклада или презентации, позволяя докладчику не держать всю нужную информацию у себя в памяти.

**Режим структуры.** Этот режим используется для подготовки заголовков и текста слайдов, а также выбора дизайна оформления. Графические объекты, диаграммы и т. п. в этом режиме в слайды не вставляются.

**Режим показа слайдов.** Режим позволяет просмотреть процесс прохождения презентации.

**Слайды презентации.** Для создания новой презентации нужно выбрать положение *новая презентация*. Новая презентация содержит один слайд с макетом **Титульный слайд**. Для изменения макета используйте контекстное меню, команду **Макет** или кнопку **Макет** на вкладке **Главная** в группе **Слайды**, в диалоговом окне **Тема Office** выберите необходимую разметку. После выбора одного из видов слайдов на экране появится соответствующий шаблон. Для создания следующего слайда необходимо выполнить команду **Главная** → **Создать слайд...** и в открывшемся после этого диалоговом окне **Тема Office** выбрать макет нового слайда.

Для удаления слайда используется команда **Удалить слайд**.

При создании новой презентации на базе шаблона ввод текста представляет собой замену вставленных PowerPoint элементов (местозаполнителей) на каждом слайде. Если же выбран пустой слайд, то текстовые поля в нем нужно создать самостоятельно с помощью команды **Вставка** → **Текст** → **Надпись**.

Редактирование введенного текста осуществляется аналогично редактированию текста в Word, выделив предварительно соответствующий текстовый фрагмент или текстовую область.

Для того чтобы отформатировать соответствующий текст, его также сначала нужно выделить, а затем применить соответствующие команды в группах **Шрифт** и **Выравнивание** на вкладке **Главная**. Прямой доступ к командам форматирования можно получить в контекстном меню текстового поля.

Для придания презентации завершенной формы следует создать единый стиль слайдов: единый фон, одинаковый шрифт и т. п. Для этого необходимо выполнить команду **Дизайн** и, используя набор стилей и инструментов в группах **Темы** и **Фон**, выбрать и применить подходящий стиль. Все слайды разработанной презентации получат выбранный дизайн.

Изменить последовательность слайдов в презентации можно в режиме сортировщика слайдов, который вызывается из раздела **Вид**. В этом режиме слайды можно расположить в нужной очередности обычным перетаскиванием, а также скопировать в буфер, вырезать, вставить из буфера или удалить.

**Анимация в процессе смены слайдов.** Для настройки перехода от одного слайда к другому нужно выделить слайд и выполнить команду

**Анимация** в группе **Переход к этому слайду**, указать, какой анимационный эффект будет использоваться при смене слайдов, какими звуками это будет сопровождаться, что будет вызывать смену кадров – щелчок мыши или истекший интервал времени и т. д. Выбранные настройки можно применить как к одному текущему слайду, так и сразу ко всем слайдам презентации.

**Анимация объектов слайда.** Эффекты для отдельных элементов слайда проще всего установить в режиме слайдов. Здесь всегда обрабатывается тот слайд, который виден на экране. Для того чтобы исследовать все возможности анимации, вызывается область задач **Настройка анимации**, открываемая одноименной командой в разделе **Анимация**.

**Интерактивная презентация.** Для создания интерактивной презентации необходимо в процессе демонстрации презентации иметь возможность изменять последовательность предъявления слайдов.

Существуют два различных способа создания переходов. Первый способ состоит в создании *гиперссылок* на другие слайды или на другие объекты (документы на локальном компьютере и web-страницы в Интернет). Для этого используется команда **Вставка → Связи → Действие**.

Второй способ состоит в размещении на слайдах *управляющих элементов*, например *Кнопок*. Если активизировать кнопку, то произойдет переход на другой слайд. Для этого с помощью команды **Вставка → Иллюстрации → Фигуры → Управляющие кнопки** необходимо добавить кнопку на слайд.

Во время экранной презентации один слайд может сменяться другим автоматически. Чтобы обеспечить нормальное восприятие каждого слайда, следует задать необходимую длительность отображения его на экране. В PowerPoint предусмотрено два способа проведения презентации: либо слайды сменяются на экране автоматически, и тогда необходимо установить время их воспроизведения, либо пользователь самостоятельно определяет момент перехода на следующий слайд.

### **Контрольные вопросы**

#### **Программное обеспечение персонального компьютера.**

1. Понятие программного обеспечения.
2. Классификация программного обеспечения.
3. Системное программное обеспечение и его составляющие.
4. Операционная система, ее функции.



5. Сервисное программное обеспечение.
6. Прикладное программное обеспечение, его классификация.
7. Пакеты прикладных программ офисного назначения, их компоненты.

8. Инструментальное программное обеспечение, его назначение.

9. Назовите наиболее распространенные текстовые процессоры.

### **Текстовый процессор Microsoft Word.**

10. Назовите основные элементы окна текстового редактора Microsoft Word.

11. Назовите основные вкладки ленты.

12. Как сохранить документ на диске?

13. Как вывести документ на печать?

14. Перечислите основные операции работы с текстом.

15. Поясните назначение кнопок в группе *Шрифт* ленты.

16. Поясните назначение кнопок в группе *Абзац* ленты.

17. Как создать маркированный, нумерованный, многоуровневый список?

18. Какие виды колонтитулов можно установить в документе?

19. Что такое сноска?

20. Как вставить в документ картинку?

21. Что такое перекрестная ссылка и гиперссылка. Как их вставить в документ?

22. Расскажите алгоритм вставки оглавления.

23. Что такое раздел и для чего он создается?

24. Что такое стиль? Для чего он создается?

25. Что такое шаблон? Для какой цели он создается?

### **Табличный процессор Microsoft Excel.**

26. Назовите основные функциональные возможности табличного процессора Microsoft Excel.

27. Как изменить ширину строки (столбца) электронной таблицы?

28. Как вставить группу строк (столбцов)?

29. Как объединить несколько ячеек таблицы в одну?

30. Как изменить формат таблицы?

31. Как выполняется копирование ячеек?

32. Как сохранить таблицу на диск?

33. Как загрузить таблицу на диск?

34. Как вывести на печать электронную таблицу?

35. Что такое *Строка формул*, какие поля она имеет?

36. Где можно увидеть адрес текущей ячейки?
37. Что такое абсолютный, смешанный и относительный адреса ячейки, как они записываются?
38. Какие операции над ячейками таблицы выполняют функции: ABS(), СРЗНАЧ(), СУММ(), МАКС(), МИН(), СЧЕТ()?
39. Назовите основные элементы диаграммы.
40. Что такое подписи данных на диаграмме?
41. Как подписать диаграмму (вывести название)?
42. Как убрать легенду на диаграмме и как ее восстановить?
43. Расскажите о назначении каждой кнопки ленты *Работа с диаграммой/Конструктор*.
44. Как изменить размеры диаграммы?
45. Каким образом диаграмма перемещается по рабочему листу?
46. Для чего используется команда *Промежуточные итоги*?
47. Что такое сводная таблица? Расскажите технологию создания сводной таблицы.
48. Для чего используют средство *Поиск решения*?
- Презентации Microsoft PowerPoint.**
49. Какие бывают типы презентаций?
50. Каковы основные этапы создания презентаций?
51. Какие существуют средства создания презентаций?
52. Как запустить программу PowerPoint?
53. Какие существуют режимы просмотра презентации? Для чего предназначен каждый режим просмотра?
54. Как создать новую презентацию?
55. Что такое шаблон дизайна? Как использовать шаблон презентации?
56. Как сохранить презентацию?
57. Как добавить новый слайд в презентацию? Как удалить слайд из презентации? Как переместить слайд в другое место? Как продублировать слайд?
58. Какие объекты можно размещать на слайде?
59. Что такое местозаполнитель?
60. Что такое дублирование объектов и какие эффекты можно применять для объектов?
61. Что такое анимация? Какие виды спецэффектов можно использовать в презентациях? Как добавить спецэффект? Как можно увидеть работу внесенных в презентацию эффектов при смене слайдов?

- 62. Какие эффекты анимации существуют в PowerPoint?
- 63. К каким объектам можно применить анимацию?
- 64. Какие виды демонстрации презентаций бывают?
- 65. Чем может завершиться демонстрация презентации?

#### ЛИТЕРАТУРА

1. Работа пользователя в Microsoft Excel 2010: учеб. пособие / Т. В. Зудилова [и др.]; под общ. ред. Т. В. Зудиловой. – СПб.: НИУ ИТМО, 2012. – 87 с.
2. Стригунов, В. В. Введение в электронные таблицы Excel: учеб. пособие / В. В. Стригунов; [науч. ред. Т. М. Попова]. – Хабаровск: Изд-во Тихоокеан. гос. ун-та, 2018. – 68 с.
3. Вишневецкий, В. Ю. Работа с Microsoft Office 2016: Excel: метод. руководство к выполнению лабораторных работ / В. Ю. Вишневецкий, И. Б. Старченко, В. С. Лебедева. – Ростов н/Д: Изд-во ЮФУ, 2016. – 36 с.
4. Речицкая, И. В. Microsoft Excel – универсальная система обработки данных: метод. указания [Электронный ресурс] / И. В. Речицкая; Дальневост. федерал. ун-т, Инженерная школа. – Владивосток: Издат. дом Дальневост. федерал. ун-та, 2013. – 52 с. – Режим доступа: <http://dvfu.ru/wcb/is/publikaciil>.
5. Кананович, А. В. Технология обработки числовых данных в электронных таблицах Excel / А. В. Кананович, Е. А. Кананович. – Минск: РИВШ, 2015. – 69 с.
6. Уокенбах, Д. Microsoft Office Excel 2007. Библия пользователя / Д. Уокенбах; пер. с англ. – М.: ООО «ИД «Вильямс», 2008. – 816 с.

## Тема 3. СЕТЕВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

3.1. Локальные сети.

3.2. Глобальная сеть Интернет.

3.3. Основы проектирования web-страниц.

### 3.1. Локальные сети

**Компьютерная (вычислительная) сеть** – это совокупность компьютеров и коммутационных устройств, соединенных с помощью каналов связи в единую систему, удовлетворяющую требованиям распределенной обработки данных.

**Распределенная обработка данных** – это обработка данных, выполняемая на независимых, но связанных между собой компьютерах, представляющих распределенную систему.

Компьютерные сети являются логическим результатом эволюции компьютерных и телекоммуникационных технологий. Постоянно возрастающие потребности пользователей в вычислительных ресурсах обуславливали попытки специалистов компьютерных технологий объединить в единую систему отдельные компьютеры.

Появление локальных компьютерных сетей внесло много нового в использование вычислительной техники. Появилась возможность быстрого доступа к разделяемым вычислительным ресурсам, к базе данных сразу несколькими пользователями, причем пользователь использовал на своем сетевом компьютере знакомые команды, как и при работе с отдельным компьютером. Задачу обработки этих команд и распределения задач между отдельными компьютерами взяла на себя сетевая операционная система.

В настоящее время разделение компьютерных сетей на глобальные и локальные происходит в первую очередь по признаку их территориального размещения, по механизму установления связей между компьютерами и скорости передачи данных.

К основным характеристикам сетей относятся:

– *пропускная способность* – максимальный объем данных, передаваемых сетью в единицу времени. Пропускная способность измеряется в Мбит/с;

– *время реакция сети* – время, затрачиваемое программным обеспечением и устройствами сети на подготовку к передаче информации по данному каналу. Время реакции сети измеряется в миллисекундах.

Базовые требования к компьютерным сетям:

- *открытость* – возможность включения дополнительных компьютеров, терминалов, узлов и линий связи без изменения технических и программных средств существующих компонентов;

- *надежность* – сохранение работоспособности при изменении структуры;

- *адаптивность* – допустимость изменения типов компьютеров, терминалов, линий связи, операционных систем;

- *эффективность* – обеспечение требуемого качества обслуживания пользователей при минимальных затратах;

- *безопасность* – способность сети обеспечить защиту информации от несанкционированного доступа и сохранить достоверность и целостность передаваемых данных.

Для передачи данных компьютеры используют самые разнообразные физические каналы (коаксиальный кабель, витая пара, оптоволоконный кабель), которые обычно называются *средой передачи*.

Если в сети имеется специальный компьютер, выделенный для совместного использования участниками сети, он называется *файловым сервером*.

Группы сотрудников, работающих над одним проектом в рамках локальной сети, называются *рабочими группами*. В рамках одной локальной сети могут работать несколько рабочих групп. У участников рабочих групп могут быть разные права для доступа к общим ресурсам сети. Совокупность приемов разделения и ограничения прав участников компьютерной сети называется *политикой сети*. Управление сетевыми политиками называется *администрированием сети*. Лицо, управляющее организацией работы участников локальной компьютерной сети, называется *системным администратором*.

По *территориальной распространенности* компьютерные сети подразделяются:

- на локальные сети (LAN, Local Area Network) в пределах предприятия, учреждения, одной организации;

- региональные сети (MAN, Metropolitan Area Network) в пределах города или области;

- глобальные сети (WAN, Wide Area Network) позволяют организовать взаимодействие между компьютерами на больших расстояниях, в пределах территории государства или группы государств.

По *скорости передачи информации* компьютерные сети подразделяются:

- на низкоскоростные сети – до 10 Мбит/с;
- среднескоростные сети – до 100 Мбит/с;
- высокоскоростные сети – свыше 100 Мбит/с.

По *типу среды передачи* компьютерные сети разделяются:

- на проводные (на коаксиальном кабеле, витой паре, оптоволоконные);
- беспроводные, с передачей информации по радиоканалам или в инфракрасном диапазоне.

По *способу организации взаимодействия компьютеров* сети делят на *одноранговые* и *с выделенным сервером* (иерархические сети).

*Одноранговая сеть.* Все компьютеры равноправны. Любой пользователь сети может получить доступ к данным, хранящимся на любом компьютере.

*Достоинство* – простота установки и эксплуатации.

*Недостаток* – затруднено решение вопросов защиты информации.

Такой способ организации используется для сетей с небольшим количеством компьютеров и там, где вопрос защиты данных не является принципиальным.

*Иерархическая сеть.* При установке заранее выделяются один или несколько серверов – компьютеров, управляющих обменом данных и распределением ресурсов сети. Сервер – это постоянное хранилище разделяемых ресурсов. Любой компьютер, имеющий доступ к услугам сервера, называют *клиентом сети*, или *рабочей станцией*. Сам сервер также может быть клиентом сервера более высокого уровня иерархии. Серверы обычно представляют собой высокопроизводительные компьютеры, возможно, с несколькими параллельно работающими процессорами, винчестерами большой емкости и высокоскоростной сетевой картой.

*Достоинство* – позволяет создать наиболее устойчивую структуру сети, более рационально распределить ресурсы и обеспечить более высокий уровень защиты данных.

*Недостатки:*

- необходимость дополнительной операционной системы для сервера;
- более высокая сложность установки и модернизации сети;
- необходимость выделения отдельного компьютера в качестве сервера.

По *технологии использования сервера* различают сети с архитектурой файл-сервер и архитектурой клиент-сервер.

**Файл-сервер.** На сервере хранится большинство программ и данных. По требованию пользователя ему пересылается необходимая программа и данные. Обработка информации выполняется на рабочей станции.

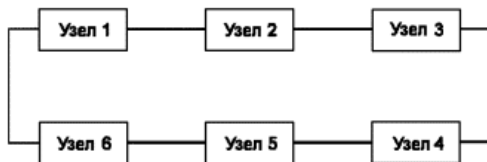
**Клиент-сервер.** Хранение данных и их обработка производится на сервере, который выполняет также контроль за доступом к ресурсам и данным. Рабочая станция получает только результаты запроса.

**Локальная вычислительная сеть (ЛВС).** Вычислительные машины, входящие в состав ЛВС, могут быть расположены самым случайным образом на территории, где создается вычислительная сеть.

Топология ЛВС – это усредненная геометрическая схема соединений узлов сети.

Топологии вычислительных сетей могут быть самыми различными, но для ЛВС сетей типичными являются три: *кольцевая, шинная, звездообразная.*

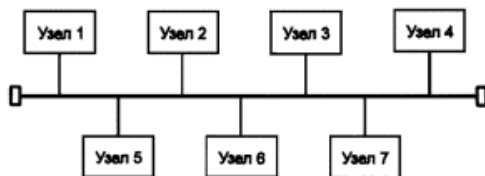
**Кольцевая топология** предусматривает соединение узлов сети замкнутой кривой – кабелем передающей среды. Выход одного узла сети соединяется со входом другого. Информация по кольцу передается от узла к узлу. Каждый промежуточный узел между передатчиком и приемником ретранслирует посланное сообщение. Принимающий узел распознает и получает только адресованные ему сообщения.



Кольцевая топология является идеальной для сетей, занимающих сравнительно небольшое пространство. В ней отсутствует центральный узел, что повышает надежность сети. В качестве *передающей среды используются любые типы кабелей.* Но последовательная дисциплина обслуживания узлов такой сети снижает ее быстродействие, а выход из строя одного из узлов нарушает целостность.

**Шинная топология** – одна из наиболее простых. Она связана с использованием в качестве *передающей среды коаксиального кабеля.* Данные от передающего узла сети распространяются по шине в обе стороны. Промежуточные узлы не транслируют поступающих сообщений. Информация поступает на все узлы, но принимает сообщение

только тот, которому оно адресовано. Дисциплина обслуживания параллельная.



Высокое быстродействие ЛВС. Сеть легко наращивать и конфигурировать, а также адаптировать к различным системам. Сеть устойчива к возможным неисправностям отдельных узлов, но имеет малую протяженность и не позволяет использовать различные типы кабеля в пределах одной сети. На концах сети устанавливают специальные устройства – *терминаторы*.

*Звездообразная топология* базируется на концепции центрального узла, называемого *концентратором*, к которому подключаются периферийные узлы. Каждый периферийный узел имеет свою отдельную линию связи с центральным узлом. Вся информация передается через центральный узел, который ретранслирует, переключает и маршрутизирует информационные потоки в сети.

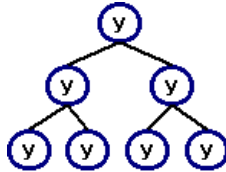


Звездообразная топология значительно упрощает взаимодействие узлов ЛВС друг с другом, позволяет использовать более простые сетевые адаптеры. В то же время работоспособность ЛВС со звездообразной топологией целиком зависит от центрального узла.

В реальных вычислительных сетях могут использоваться более сложные топологии, представляющие в некоторых случаях сочетания рассмотренных. Выбор той или иной топологии определяется областью применения сети, географическим расположением ее узлов и размерностью сети в целом. Например:

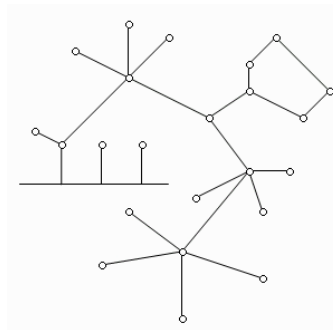


*Ячеистая топология.* Для нее характерна схема соединения узлов, при которой физические линии связи установлены со всеми рядом стоящими компьютерами:



В сети с ячеистой топологией непосредственно связываются только те компьютеры, между которыми происходит интенсивный обмен данными, а для обмена данными между компьютерами, несоединенными прямыми связями, используются транзитные передачи через промежуточные узлы. Ячеистая топология допускает соединение большого количества компьютеров и характерна, как правило, для глобальных сетей. Достоинства данной топологии в ее устойчивости к отказам и перегрузкам, так как имеется несколько способов обойти отдельные узлы.

*Смешанная топология.* В таких сетях можно выделить отдельные подсети, имеющие типовую топологию – звезду, кольцо или общую шину, которые для крупных сетей связываются произвольно.



**Построение локальных сетей.** Передающая среда является общим ресурсом для всех узлов сети. Чтобы получить возможность доступа к этому ресурсу из узла сети, необходимы специальные механизмы – методы доступа. *Метод доступа к передающей среде* – метод, обеспе-

чивающий выполнение совокупности правил, по которым узлы сети получают доступ к ресурсу.

*Маркерный доступ.* Компьютер-абонент получает от центрального компьютера сети маркер-сигнал на право ведения передачи в течение определенного времени, после чего маркер передается другому абоненту.

При *конкурентном методе доступа* абонент начинает передачу данных, если обнаруживает свободную линию.

*Сеть Ethernet.* Схема передачи данных конкурентная, элементы сети могут быть соединены по шинной или звездной топологии с использованием витых пар, коаксиальных и волоконно-оптических кабелей. Основное преимущество – быстродействие от 10 до 100 Мб/с.

*Сеть Token Ring.* Схема с маркерным доступом. Физически выполнена как звезда, но ведет себя как кольцевая. Данные передаются последовательно от станции к станции, но постоянно проходят через центральный узел. Используются витые пары и волоконно-оптические кабели. Скорость передачи 4 или 16 Мб/с.

*Сеть ARCnet.* Схема с маркерным доступом, может работать как с шинной, так и звездной топологией. Совместима с витой парой, коаксиальным и волоконно-оптическим кабелем. Скорость передачи 2.5 Мб/с.

**Беспроводные сети.** *Беспроводные сети* являются одной из главных тенденций развития компьютерных сетей. Они дают возможность развертывания компьютерной сети там, где прокладка традиционной кабельной системы невозможна или нежелательна.

Беспроводные сетевые технологии и построенные на их основе сети передачи данных (БСПД) – это технологии и сети, позволяющие организовать передачу информации без использования кабельной проводки. В качестве среды передачи в таких сетях обычно выступают радиоволны, в редких случаях – инфракрасное излучение.

Беспроводные сети передачи данных классифицируют:

*по географической протяженности:*

– персональные (WPAN, Wireless Personal Area Network); локальные, региональные (WLAN, Wireless Local Area Network, WMAN Wireless Metropolitan Area Networks); глобальные (WWAN, Wireless Wide Area Network);

*по применимой технологии связи:*

– радиорелейные (радиосвязь по линии, образованной цепочкой преопередающих радиостанций);

- спутниковые;
- атмосферные оптические (делают возможной передачу данных Интернет, голос, видео, телефония, телевидение между объектами в атмосфере, предоставляя оптическое соединение без использования оптоволокну);

- мобильные сотовые;
- транковые (подвижная радиосвязь) и т. д.;

*по каналу передачи данных:*

- радио; лазерные; инфракрасные и т. д.;

*по ширине полосы передачи* (чем шире полоса, тем выше скорость передачи данных):

- узкополосные; широкополосные; сверхширокополосные;

*по локализации абонентов:*

- фиксированные; подвижные.

Одним из важнейших факторов, влияющих на работоспособность БСПД, является обеспечение качества передачи и приема радиосигнала. При этом необходимо обеспечить не только требуемую скорость, но и защиту передаваемой информации от помех и посторонних вторжений. Это является одним из главных недостатков по сравнению с проводными сетями.

Одна из основных проблем построения БПСД – решение **задачи доступа многих пользователей к ограниченному ресурсу среды передачи**, которой является радиоканал, т. е. другими словами, как отличить в радиоканале сигнал одного пользователя от сигнала другого.

Сети WPAN обеспечивают обмен информацией между такими устройствами, как карманные и обычные персональные компьютеры, мобильные телефоны, ноутбуки, принтеры, цифровые фотоаппараты, мышки, клавиатуры, джойстики, наушники, гарнитуры на надежной, недорогой, повсеместно доступной радиочастоте для ближней связи. Сети работоспособны в радиусе от 10 до 100 м (дальность очень сильно зависит от преград и помех) даже в разных помещениях.

Наиболее известные сети WPAN – это сети, использующие технологии Bluetooth и ZigBee.

### 3.2. Глобальная сеть Интернет

В 1961 г. Defence Advanced Research Agency (DARPA – оборонное агентство передовых исследовательских проектов) по заданию министерства обороны США приступило к проекту по созданию экспери-

ментальной сети передачи пакетов. Эта сеть, названная ARPANET, первоначально предназначалась для изучения того, как поддерживать связь в случае ядерного нападения и для помощи исследователям в обмене информацией между разбросанными по всем штатам исследовательскими организациями оборонной промышленности.

В основу проекта были положены три основные идеи:

- каждый узел сети соединен с другими таким образом, что существует несколько различных путей от узла к узлу;
- все узлы и связи рассматриваются как ненадежные;
- существуют автоматически обновляемые таблицы перенаправления пакетов. Если нужный узел в данный момент недоступен, пакет, предназначенный для него, согласно этой таблице отправляется на ближайший к нему, затем на следующий и т. д.

Созданная по таким принципам система не имела централизованного узла управления и, следовательно, безболезненно могла изменять свою конфигурацию.

Эксперимент с ARPANET был настолько успешен, что многие организации захотели войти в нее с целью использования для ежедневной передачи данных. И в 1975 г. ARPANET превратилась из экспериментальной сети в рабочую сеть.

В конце 80-х гг. Россия подключилась к сети ARPANET. В 1990 г. сеть ARPANET перестала существовать, и на ее месте возник Интернет. Интернет сделала возможным свободный обмен информацией, невзирая на расстояния, государственные границы.

Слово *Интернет* происходит от выражения *interconnected networks* (связанные сети). В узком смысле – это *глобальное сообщество малых и больших сетей*, принадлежащих различным компаниям и предприятиям; работающих по самым разнообразным протоколам; связанных между собой различными линиями связи; передающих физически данные по телефонным проводам, оптоволокну, через спутники и радиомодемы. В более широком смысле – это *глобальное информационное пространство*, хранящее огромное количество информации на миллионах компьютерах, которые обмениваются данными. Каждый компьютер, подключенный к Интернету, – это часть Сети.

Интернет состоит из разнородных подсетей, соединенных друг с другом шлюзами. В качестве подсетей могут выступать разные локальные сети (Token Ring, Ethernet и т. п.), различные национальные, региональные и глобальные сети. К этим сетям могут подключаться

машины разных типов. Каждая из подсетей работает в соответствии со своими принципами и типом связи.

За Интернет никто централизованно не платит, каждый платит за свою часть. Представители сетей собираются вместе и решают, как им соединяться друг с другом и содержать эти взаимосвязи. Пользователь платит за подключение к некоторой региональной сети, которая в свою очередь платит за свой доступ сетевому владельцу государственного масштаба. Интернет не имеет никакого собственника, здесь нет и специального органа управления, который бы контролировал всю работу сети Интернет. Локальные сети различных стран финансируются и управляются местными органами согласно политике данной страны.

Существование Интернета стало возможным только после создания базовых протоколов TCP/IP, которые и предназначены для объединения сетей. Эти протоколы работают на двух уровнях: TCP – на транспортном, IP – на сетевом (адресном).

В основе принципа сети Интернет положен метод пакетной коммутации.

*Коммутация пакетов* – метод, специально разработанный для компьютерных сетей, где различные компьютеры сети могут иметь различное быстродействие. При коммутации пакетов все передаваемые сообщения разбиваются передающим компьютером на части (от 46 до 1500 байт), называемые пакетами. Каждый пакет снабжается заголовком, в котором указывается адресная информация, необходимая для доставки пакета к принимающему компьютеру, а также номер пакета, используемый для «сборки» сообщений на принимающем компьютере. Пакеты транспортируются в сети как независимые информационные блоки. Специальные устройства сети (коммутаторы) принимают пакеты от передающих компьютеров и на основании адресной информации передают их друг другу до конечного принимающего компьютера. За счет буферизации (задержки) пакета во внутренней памяти коммутатора (если требуемый участок сети занят передачей другой информации) выравнивается скорость передачи данных в сети в целом и повышается пропускная способность сети.

Поскольку компьютеры и сетевое оборудование могут быть от разных производителей, то возникает проблема их совместимости. Без принятия всеми производителями общепринятых правил построения оборудования создание компьютерной сети было бы невозможно. Поэтому разработка и создание компьютерных сетей может происходить только в рамках утвержденных стандартов.

В основу стандартизации компьютерных сетей положен принцип декомпозиции, т. е. разделения сложных задач на отдельные более простые подзадачи. При рассмотрении работы компьютера в сети можно выделить две основные подзадачи:

- взаимодействие программного обеспечения пользователя с физическим каналом связи (посредством сетевой карты) в пределах одного компьютера;
- взаимодействие компьютера через канал связи с другим компьютером.

Современное программное обеспечение компьютера имеет многоуровневую модульную структуру, т. е. программный код проходит несколько уровней обработки, прежде чем превратиться в электрический сигнал (модуль нижнего уровня), передаваемый в канал связи.

При взаимодействии компьютеров через канал связи оба компьютера должны выполнять ряд соглашений. Например, они должны согласовать величину и форму электрических сигналов, длину сообщений, методы контроля достоверности и т. д. На рис. 3.1 представлена модель взаимодействия двух компьютеров в сети. Процедура взаимодействия каждого уровня этих компьютеров может быть описана в виде набора правил взаимодействия каждой пары модулей соответствующих уровней.

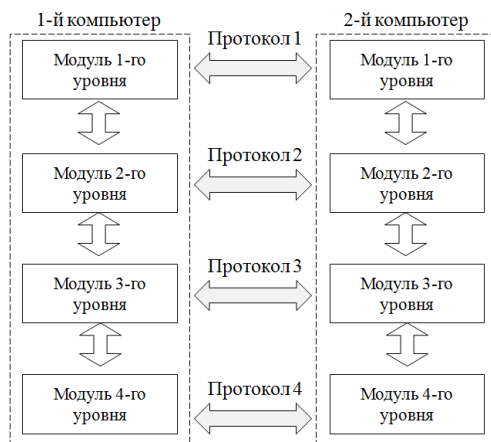


Рис. 3.1. Взаимодействие двух компьютеров в сети

Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются модули, лежащие на одном уровне, но в различных компьютерах, называются *протоколами*.

Суть работы многоуровневого протокола можно представить как «письмо в конверте». Каждый уровень протокола надписывает на «конверте» свою информацию. Сетям нужно только понимать «надпись» на «конверте», чтобы предать его в место назначения, а до содержания письма им дела нет.

Модули, реализующие протоколы соседнего уровня и находящиеся в одном компьютере, также взаимодействуют друг с другом в соответствии с четко определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила называются *интерфейсом* и определяют набор сервисов, предоставляемых данным уровнем соседнему уровню. Модули должны обрабатывать, во-первых, свой собственный протокол, а во-вторых, интерфейсы с соседними уровнями.

Иерархически организованный набор протоколов для взаимодействия компьютеров в сети называется *стеком коммуникационных протоколов*. Коммуникационные протоколы могут быть реализованы как программно, так и аппаратно. Протоколы нижних уровней, как правило, реализуются комбинацией программно-аппаратных средств, а протоколы верхних уровней – чисто программными средствами. Протоколы каждого уровня обладают независимостью друг от друга, т. е. протокол любого уровня может быть изменен, не оказывая при этом никакого влияния на протокол другого уровня. Независимость протоколов посредством интерфейсов является важнейшей предпосылкой для создания стандартных протоколов для компьютерных сетей.

Международной организацией по стандартизации (ISO – International Standards Organization) была предложена модель взаимодействия двух компьютеров в сети, названная OSI (Open system interconnection – взаимодействие открытых систем). В ней реализована семиуровневая архитектура, причем каждому из уровней соответствует определенная функция (рис. 3.2).

Взаимодействие компьютеров в сети начинается с того, что приложение (программа пользователя) одного компьютера обращается к прикладному уровню другого компьютера, например, к файловой системе. Приложение первого компьютера формирует с помощью операционной системы сообщение стандартного формата, состоящее из заголовка и поля данных.

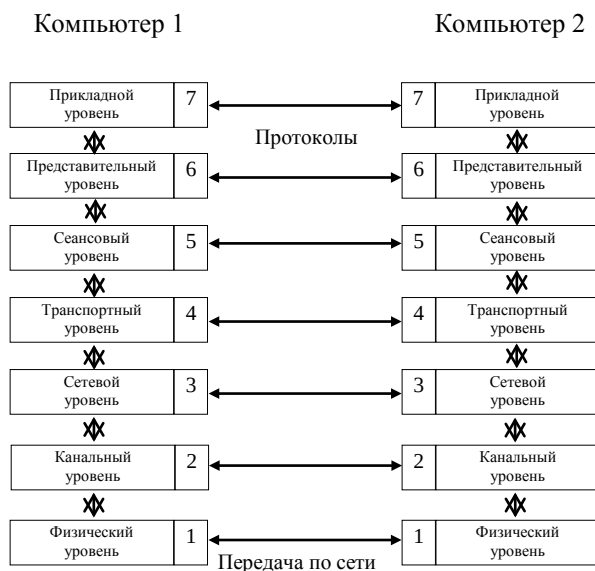


Рис. 3.2. Модель OSI

Заголовок содержит служебную информацию, которую необходимо передать через сеть прикладному уровню другого компьютера, чтобы сообщить ему, какую работу необходимо выполнить. В заголовке имеется информация для следующего нижнего уровня, чтобы он «знал», что делать с этим сообщением. В поле данных находится передаваемая информация. Сформировав сообщение, прикладной уровень направляет его представительному уровню, который, прочитав заголовок, выполняет требуемые действия над сообщением и добавляет к сообщению собственную служебную информацию (указания для протоколов представительного уровня второго компьютера). Полученное в результате сообщение передается сеансовому уровню, который в свою очередь добавляет свой заголовок и т. д. При достижении сообщением физического уровня у него имеется множество заголовков, добавленных на предыдущем уровне. В таком виде оно и передается по сети (рис. 3.3).

Второй компьютер принимает его на физическом уровне и последовательно перемещает вверх с уровня на уровень (рис. 3.3).



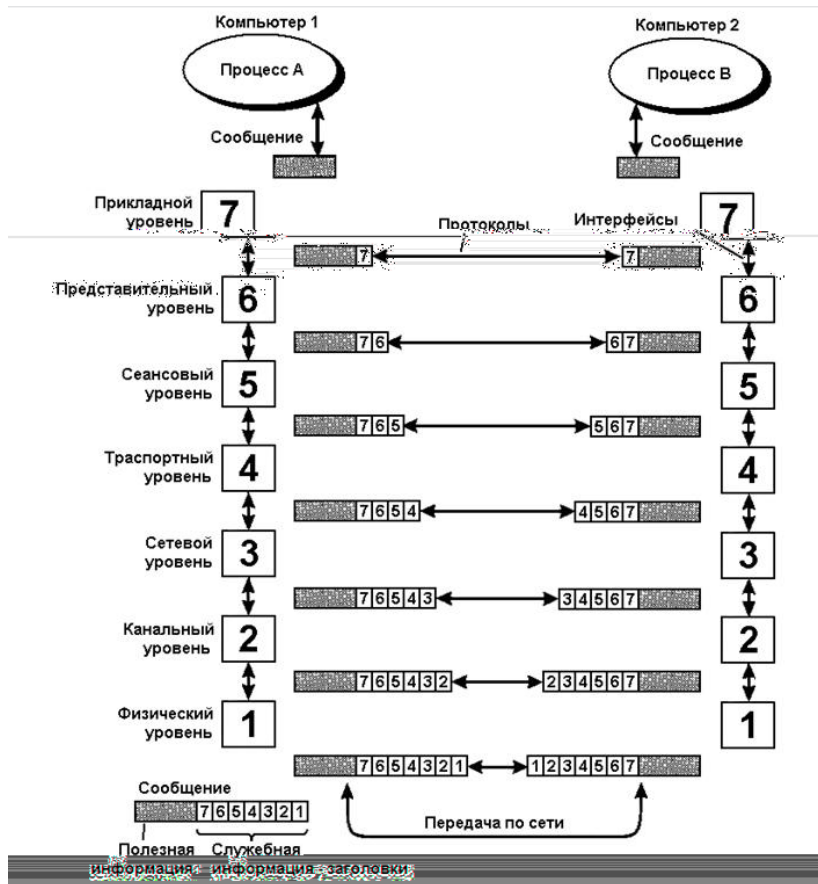


Рис. 3.3. Взаимодействие компьютеров в сети

Каждый уровень анализирует и обрабатывает заголовок своего уровня, выполняя соответствующие этому уровню функции, а затем удаляет этот заголовок и передает сообщение вышележащему уровню. В результате взаимодействия протоколов всех уровней и их единому стандарту на прикладном уровне второго компьютера получают данные, переданные первым компьютером.

В стандарте OSI для обозначения единиц данных, с которыми имеют дело протоколы различных уровней, используются специальные названия: *кадр*, *пакет*, *дейтаграмма*, *сегмент*.

*Физический уровень* имеет дело с передачей битов информации по физическим каналам связи (коаксиальный кабель, витая пара, оптоволоконный кабель). На этом уровне стандартизируются характеристики электрических сигналов, уровни напряжения и тока, тип кодировки информации, скорость передачи сигналов, а также типы разъемов и назначение каждого контакта. Пример стандарта физического уровня: спецификация 10Base-T технологии Ethernet, которая использует витую пару с волновым сопротивлением 100 Ом, разъем RJ45, максимальную длину 100 м.

*Канальный уровень* обеспечивает надежную передачу данных через физический канал. Канальный уровень оперирует блоками данных, называемых *кадрами*. Основной задачей канального уровня является прием кадра из сети и отправка его в сеть. Он осуществляет физическую адресацию передаваемых сообщений, контролирует соблюдение правил использования физического канала, выявляет неисправности, управляет потоками информации. Кадр может быть доставлен только в случае, если протокол соответствует той топологии, для которой он был разработан (общая шина, кольцо и звезда). Протоколы канального уровня для локальных сетей – Ethernet, Token Ring, FDDI, для глобальных – PPP, SLIP, LAP-B, LAP-D.

Для реализации протоколов канального уровня используется специальное оборудование – *коммутаторы*. Раньше использовались *концентраторы и мосты*, которые в настоящее время сняты с производства.

*Сетевой уровень* служит для образования единой системы, объединяющей несколько сетей, причем эти сети могут быть различной топологии, использовать совершенно различные принципы сообщений между конечными узлами и обладать произвольной структурой. Сети соединяются между собой специальными устройствами – *маршрутизаторами*. *Маршрутизатор* – устройство, которое собирает данные о топологии межсетевых соединений и на ее основании пересылает пакеты информации из одной сети в другую. Последовательность маршрутизаторов, через которые проходит пакет, называется *маршрутом*, а выбор маршрута называется *маршрутизацией*. На сетевом уровне действуют три протокола: *сетевой протокол* – для определения правил передачи пакетов от конечных узлов к маршрутизаторам и между маршрутизаторами; *протокол маршрутизации* – для сбора информации о топологии сети; *протокол разрешения адресов* – для отображения адреса узла, используемого на сетевом уровне в локальный адрес

сети. Примеры протоколов сетевого уровня: протокол межсетевого взаимодействия IP стека TCP/IP и протокол межсетевого обмена пакетами IPX стека Novell.

*Транспортный уровень* предназначен для оптимизации передачи данных от отправителя к получателю с той степенью надежности, которая требуется. Основная задача транспортного уровня – обнаружение и исправление ошибок в сообщениях. Начиная с транспортного уровня, все дальнейшие протоколы реализуются программным обеспечением компьютера, включаемого обычно в состав сетевой операционной системы. Примеры транспортных протоколов: протоколы TCP стека TCP/IP и протокол SPX Novell.

*Сеансовый уровень* управляет диалогом между двумя компьютерами и устанавливает правила начала и завершения взаимодействия, определяет, какая из сторон является активной в данный момент, а какая принимает данные. Приложение должно различать разные потоки данных в пределах одного соединения (например, приложение может одновременно запрашивать два файла с одного сервера, благодаря сеансовому уровню оно будет различать эти два потока).

*Представительный уровень* выполняет преобразование данных между устройствами с различными форматами данных, не меняя при этом содержания. Благодаря этому уровню информация, передаваемая прикладным уровнем одного компьютера, всегда понятна прикладному уровню другого компьютера. На этом уровне, как правило, происходит шифрование и дешифрование данных, благодаря которому обеспечивается секретность передаваемого сообщения.

*Прикладной уровень* является пользовательским интерфейсом для работы с сетью. Он непосредственно взаимодействует с пользовательскими прикладными программами, предоставляя им доступ в сеть. С помощью протоколов этого уровня пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры, гипертекстовые web-страницы, электронная почта.

Необходимо отметить, что три нижних уровня модели OSI – физический, канальный и сетевой – зависят от сети, т. е. их протоколы тесно связаны с технической реализацией сети и с используемым коммутационным оборудованием. Три верхних уровня – сеансовый, представления и прикладной – реализуются программным способом и мало зависят от особенностей построения сети. Транспортный уровень является промежуточным. Он скрывает детали функционирования нижних уровней от верхних уровней. Благодаря этому уровню можно раз-

рабатывать приложения, не зависящие от технических средств транспортировки сообщений.

Системы, взаимодействующие в соответствии с моделью OSI, называются *открытыми системами*. Для них опубликованы общедоступные спецификации и стандарты, принятые в результате достижения согласия многих разработчиков и пользователей после всестороннего обсуждения. Если две сети построены с соблюдением правил открытости, то у них есть возможность использования аппаратных и программных средств разных производителей, придерживающихся одного и того же стандарта, они легко сопрягаются друг с другом, просты в освоении и обслуживании. Примером открытой системы является глобальная компьютерная сеть Интернет. В результате сеть Интернет сумела объединить в себе самое разнообразное оборудование и программное обеспечение сетей, разбросанных по всему миру.

Стек протоколов TCP/IP (Transmission Control Protocol/Internet Protocol) является базовым протоколом сети Интернет.

**Базовые протоколы Интернета.** Применительно к Интернету протокол – это правило передачи информации в сети. Различают два типа протоколов: базовые (TCP/IP) и прикладные.

**Базовый протокол TCP** (протокол управления передачей) является протоколом *транспортного* уровня Интернета. Именно он нарезает файлы на пакеты определенной длины, маркирует их, следит за тем, чтобы все пакеты дошли до места назначения неповрежденными, в случае повреждения требует повторной пересылки пакета, а также собирает пакет, доставленный по месту назначения. Благодаря таким свойствам TCP Интернет обладает высокой устойчивостью и способностью самозалечивать повреждения.

**Базовый протокол IP** (протокол взаимодействия сетей) является протоколом *сетевого (адресного)* уровня Интернета. Если протокол TCP определяет, *как* происходит передача, то протокол IP определяет, *куда* она происходит.

На сетевом уровне протокол IP каждому TCP-пакету приписывает адрес отправителя и адрес получателя. По этому IP-заголовку маршрутизатор, через который проходит пакет, и определяет, куда нужно передать пакет дальше (к какому из «ближайших» соседей, понятие ближайший – не географическое), а если с пакетом что-то случается, куда обратиться за повторной передачей.

Каждый пакет даже одного сообщения путешествует самостоятельно, причем пакет, отправленный позже, может прийти раньше.

Но независимо от траектории в результате конечного числа пересылок TCP-пакеты достигают адресата, где затем собираются и передаются на более высокие уровни.

*Основные преимущества протокола TCP/IP:*

– *независимость от сетевой технологии отдельной сети* – определяет только элемент передачи – *дейтаграмму* – и описывает способ ее движения по сети;

– *всеобщая связанность сетей* – позволяет любой паре компьютеров взаимодействовать друг с другом, каждому компьютеру назначается логический адрес, а каждая дейтаграмма содержит адреса отправителей и получателей. Промежуточные маршрутизаторы используют адрес получателя для принятия решения о маршрутизации;

– *подтверждение* – обеспечивает подтверждение правильности прохождения информации при обмене между отправителем и получателем.

В стеке TCP/IP определены 4 уровня взаимодействия, каждый из которых берет на себя определенную функцию по организации надежной работы глобальной сети: прикладной уровень, основной (транспортный) уровень, уровень межсетевого взаимодействия, уровень сетевых интерфейсов.

*Хост* – объект сети, который может передавать и принимать IP-пакеты, например, компьютер или маршрутизатор.

По модели протокола TCP/IP – программные модули, соответствующие транспортному и сеансовому уровню, устанавливаются только на конечных компьютерах (хостах).

Программный модуль протокола TCP/IP реализуется в операционной системе компьютера в виде драйвера.

**Адресация в Интернет.** Для выхода каждого компьютера в Интернет устанавливаются два уникальных адреса: цифровой IP-адрес и доменный адрес. Оба эти адреса могут применяться равноценно. Цифровой адрес удобен для обработки на компьютере, а доменный адрес – для восприятия пользователем.

*Цифровой адрес* имеет длину 32 бита. Для удобства он разделяется на четыре блока по 8 бит:

XXXXXXXX.XXXXXXXXXX.XXXXXXXXXX.XXXXXXXXXX

Адрес содержит полную информацию, необходимую для идентификации компьютера, читается слева направо, т. е. вначале указывается адрес сети, а затем адрес компьютера в сети.

**Сеть типа А** – под адрес сети выделяется 1 блок, остальные под адрес компьютера. **Сеть типа В** – под адрес сети выделяется 2 блока. **Сеть типа С** – под адрес сети выделяется три блока, следовательно, на адрес компьютера в сети С остается только 8 бит, значит, максимальное количество компьютеров, которое можно подключить, – 255, из них два адреса зарезервированы, следовательно, максимально можно подключить 253 компьютера.

*Пример.* В двоичном коде цифровой адрес в сети С записан следующим образом: 11000000.00101101.00001001.11001000. В десятичном коде он имеет вид: 192.45.9.200. Адрес сети – 192.45.9; адрес компьютера – 200.

Человек не в состоянии запомнить такое большое количество цифровых адресов, с ними работает компьютер. Для пользователя были введены текстовые адреса – **доменные имена**. Например, адрес <http://www.microsoft.com/> – содержит доменное имя microsoft.com, E-mail адрес [alpron@aha.ru](mailto:alpron@aha.ru) – содержит доменное имя aha.ru.

В отличие от цифрового адреса, доменный адрес читается в обратном порядке. Вначале идет имя компьютера, затем имя сети, в которой он находится.

Например: <http://www.pc.dpt1.company.msk.ru>.

Первым в имени стоит название реального компьютера с адресом pc. Это имя создано и поддерживается группой dpt1. Группа входит в более крупное подразделение company, далее следует домен msk, определяющий имена московской части сети, а ru – российской.

За соответствием цифровых и доменных адресов в Интернете следит служба DNS.

**Прикладные протоколы и сервисы Интернета.** В Интернете существует множество разнообразных служб (сервисов), предоставляющих те или иные услуги. Каждая служба имеет свою прикладную программу и использует свой прикладной протокол.

| Сервис                   | Протокол               | Программа                                                        | Пояснение                          |
|--------------------------|------------------------|------------------------------------------------------------------|------------------------------------|
| Электронная почта e-mail | SMTP, POP2, POP3, IMAP | MS Outlook Express, The Bat, Eudora Pro                          | Почтовая служба                    |
| WWW                      | Http                   | Браузеры: Microsoft Internet Explorer, Netscape Navigator, Opera | Просмотр web-страниц (гипертекста) |

| Сервис                    | Протокол | Программа          | Пояснение                          |
|---------------------------|----------|--------------------|------------------------------------|
| Ftp                       | ftp      | Браузеры           | Передача файлов                    |
| UseNet                    | nntp     | MS Outlook Express | Телеконференции (группы новостей)  |
| IRC (Internet Relay Chat) |          | mIRC.exe           | Общение в режиме реального времени |
| Telnet                    |          | Telnet             | Удаленное управление компьютером   |

### 3.3. Основы проектирования web-страниц

HTML (от англ. *HyperText Markup Language* – язык гипертекстовой разметки) – стандартизированный язык гипертекстовой разметки документов для просмотра web-страниц в браузере. Web-браузеры получают документ HTML от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора. Web-страница представляет собой документ HTML. Этот документ может содержать:

- текст и описание его форматирования;
- определение структуры и формата страницы: может задаваться, например, деление текста на строки и абзацы, заголовок страницы, поля, цветовая разметка страницы, фоновое изображение и фоновый звук;
- указание путей к мультимедийным объектам (сами мультимедийные объекты хранятся в отдельных файлах), а также способ их форматирования;
- таблицы;
- гиперссылки;
- активные компоненты или ссылки на документы, содержащие активные компоненты.

Интерпретирует код HTML программа-браузер, поэтому при создании web-страниц на языке HTML необходимо как можно чаще тестировать созданный документ, открывая его в браузере.

Документ HTML содержит объекты всего двух типов: текст и так называемые теги (управляющие конструкции). Именно с помощью тегов задается форматирование текста, структура и формат страницы, вставляются и формируются мультимедийные объекты, таблицы, гиперссылки и прочие элементы web-страниц.

У каждого тега есть свое уникальное имя. Для того чтобы теги отличались от обычного текста, имена их заключают в угловые скобки:

**<имя\_тега>**. Имя тега может содержать латинские буквы и цифры. При записи имен тегов регистр не имеет значения.

Все теги HTML можно разделить на **парные** и **непарные**. Парные теги окружают тот фрагмент страницы, на который распространяется их действие.

Поясним это на примере. Пусть в предложении «Браузер – это программа для просмотра web-страниц.» слово «Браузер» необходимо выделить полужирным начертанием. Тег, задающий полужирное написание, носит имя **<b>**. Таким образом, соответствующий код HTML должен выглядеть так:

|                                                                              |
|------------------------------------------------------------------------------|
| <b>&lt;b&gt;Браузер&lt;/b&gt;</b> – это программа для просмотра web-страниц. |
|------------------------------------------------------------------------------|

Обратите внимание на то, что слово «Браузер» окружено парой тегов **<b>**. Первый тег (открывающий) указывает начало фрагмента, на который должно распространяться действие тега, второй тег (закрывающий) – указывает конец этого фрагмента. В закрывающем теге сразу после угловой скобки (**<**) необходимо ставить знак **</>**. Браузер интерпретирует этот код HTML так:

|                                                           |
|-----------------------------------------------------------|
| <b>Браузер</b> – это программа для просмотра web-страниц. |
|-----------------------------------------------------------|

Существуют и непарные теги, присутствие которых в определенном месте документа указывает на то, что здесь должен появиться некий объект. Для непарных тегов закрывающий тег отсутствует. Так, тег **<img>** указывает на то, что в данной области документа должно появиться графическое изображение.

Большинство тегов имеет свой набор **атрибутов** (параметров). Атрибуты уточняют действие тегов. Как правило, использование атрибутов необязательно. Для парных тегов атрибуты указывают в открывающем теге. Атрибуту может присваиваться заданное или произвольное значение. Для одного и того же тега допускается использование нескольких атрибутов, разделенных пробелами.

Так, тег **<img>** имеет ряд атрибутов: **src** – указывает путь к графическому файлу – источнику изображения, **width** и **height** – ширину и высоту изображения (в пикселах или процентах), **alt** – альтернативный текст и др. Например:

|                                                                    |
|--------------------------------------------------------------------|
| <b>&lt;img src="../../images/ris.jpg" width=200 height=150&gt;</b> |
|--------------------------------------------------------------------|

При интерпретации кода HTML браузер отобразит изображение из указанного графического файла шириной 200 пикселей и высотой 150 пикселей. Атрибут **alt** не задан, значит, альтернативный текст отсутствует.



Регистр имен тегов и атрибутов может быть любым. Так, тег вставки графического изображения можно записать как `<img>`, `<IMG>`, `<Img>` и т. д.

Наличие и количество пробелов между тегом и текстом или соседними тегами не имеет значения. Значения атрибутов необязательно заключать в кавычки, если они содержат только буквы, цифры, точки и дефисы.

На один и тот же фрагмент страницы может распространяться действие сразу нескольких тегов. Так, пусть фрагмент текста необходимо выделить полужирным начертанием и курсивом. Тег курсива – `<i>`. Тогда код HTML записывают следующим образом:

```
<b><i>фрагмент текста</i></b>
```

Необходимо соблюдать порядок вложенности тегов. Соответствующие конструкции должны вкладываться друг в друга, не пересекаясь. Запись, подобная приведенной ниже, является неверной.

```
<b><i>фрагмент текста</b><i>
```

Поэтому многие браузеры могут неправильно интерпретировать такой HTML-код.

Следует учитывать и тот факт, что браузеры не проверяют написания кода HTML. Неправильно записанный и нераспознанный браузером фрагмент кода игнорируется. При этом никакого сообщения об ошибке браузер выдавать не будет. Таким образом, еще раз напомним о необходимости периодического просмотра документа HTML в браузере.

Версии языка HTML разрабатываются Консорциумом Всемирной паутины (World Wide Web Consortium, W3C) – организацией, разрабатывающей для Интернета единые принципы и стандарты (рекомендации), которые внедряются производителями программных продуктов и оборудования.

На момент написания курса лекций действующим стандартом является HTML5 (был представлен в 2017 г.).

Для всех документов в HTML5:

```
<!DOCTYPE html>
```

**Структура документа HTML.** Общий вид документа HTML следующий:

```
<html>
<head>
  служебная область документа
</head>
```

```
<body>
содержательная область документа
</body>
</html>
```

Итак, документ HTML начинается и заканчивается тегами **<html>**. Служебная область документа ограничивается парным тегом **<head>**. Она содержит заголовок страницы (это единственные данные из этой области, которые увидит пользователь при просмотре страницы в браузере), а также информацию для браузеров и поисковых систем, которые будут индексировать страницу.

Собственно содержимое страницы располагается внутри парного тега **<body>**. Именно о содержательной части HTML-документа и пойдет речь далее.

### Цветовая разметка документа. Цвет в HTML.

Цвет в HTML можно задать двумя способами. Первый способ – указать название цвета. В HTML есть названия для 256 цветов, но не все браузеры распознают эти названия. Однако названия основных 16 цветов способны распознать все графические браузеры, и лучше использовать только эти цвета. Приведем их названия:

**black** – черный, **silver** – светло-серый, **gray** – серый, **white** – белый, **maroon** – бордовый, **red** – красный, **purple** – фиолетовый, **fuchsia** – лиловый, **green** – зеленый, **lime** – желто-зеленый, **olive** – оливковый, **yellow** – желтый, **navy** – темно-синий, **blue** – синий, **teal** – сине-зеленый, **aqua** – голубой.

Если ни один из перечисленных цветов не подходит, необходимо определить цвет, используя схему RGB.

**Схема RGB** (Red-Green-Blue – красный-зеленый-синий) предполагает, что цвет создается путем смешения в определенных пропорциях красного, зеленого и синего цветов. Интенсивность каждого компонента записывается как двузначное шестнадцатеричное число. Например, цвет A26720 означает, что взято достаточно много красного цвета, меньше зеленого, еще меньше синего. В результате получится терракотовый цвет.

Итак, фрагмент кода HTML

```
<body bgcolor=maroon text="#256AC1">
```

означает, что фон документа будет бордовым, а цвет текста – синим с примесью зеленого и небольшой примесью красного. Обратите внимание на то, что если цвет задается по схеме RGB, то перед значением цвета необходимо ставить символ **#** и обязательно заключать значение атрибута в кавычки.

Атрибуты цветовой схемы не рекомендуются к использованию в спецификации HTML 5, хотя и поддерживаются в настоящее время большинством браузеров. Для задания цветовой разметки рекомендуется использовать CSS (Cascading Style Sheets – каскадные таблицы стилей).

Заголовок web-страницы является единственным элементом в служебной области HTML-документа, который будет отображаться при просмотре страницы в браузере. Заголовок записывается внутри парного тега **<title>**.

В служебной области HTML-документа также могут находиться **Meta-теги**, которые содержат информацию для поисковых машин и для браузеров.

### **Каскадные таблицы стилей CSS. Преимущества использования.**

Для форматирования элементов web-страницы можно применять **каскадные таблицы стилей** (Cascading Style Sheets, CSS), описывающие способы представления (отображение и расположение) элемента (или группы элементов) web-страницы. Преимущества использования каскадных таблиц стилей состоят в следующем:

- расширяются возможности форматирования (по сравнению с HTML);
- в основной части документа отсутствуют теги физического форматирования текста, т. е. представлена лишь структурная разметка документа;
- сокращается объем HTML-документа;
- возможно единое стилевое оформление для группы web-страниц.

Стандарты CSS, как и HTML, разрабатывает и рекомендует Консорциум Всемирной паутины (World Wide Web Consortium, W3C).

**Синтаксис CSS.** Технология CSS предполагает создание *правил*, которые переопределяют отображение тегов в документе или задают пользовательский стиль. Правило состоит из *селектора* и *определения* – свойств и их значений – и записывается следующим образом:

**селектор {свойство1: значение1; свойство2: значение2;...}**

Селектор указывает элемент, который необходимо форматировать.

Это может быть тег (форматирование будет применено ко всем элементам данного типа), класс (форматирование можно применять выборочно к нескольким элементам) или идентификатор (форматирование можно применять только к одному элементу).

Определение указывает способ оформления. Оно заключается в фигурные скобки, внутри которых перечислены свойства и их значения. Свойства определяют параметры форматирования. Например, свойство **color** указывает цвет элемента. Свойствам присваивается заданное или произвольное (в зависимости от свойства) значение. Между свойством и значением ставят двоеточие. Регистр символов и разбиение правила на строки не имеют значения.

Например:

```
p {color: red}
```

Такое правило означает, что во всех абзацах цвет шрифта должен быть красным.

Для одного и того же селектора можно указать несколько свойств. В таком случае их нужно перечислить в фигурных скобках через точку с запятой, например:

```
p {color: red; text-weight: bold}
```

Это означает, что во всех абзацах шрифт должен быть красного цвета и полужирный. Порядок записи свойств не имеет значения.

Если для разных селекторов необходимо указать одинаковые параметры форматирования, селекторы можно сгруппировать, перечислив через запятую:

```
h1, h2, h3 {color: green; text-align: center}
```

Это означает, что заголовки 1, 2 и 3-го уровней необходимо отображать зеленым цветом и выравнивать их по центру.

Для улучшения восприятия CSS-кода принято записывать селектор и каждое свойство в отдельных строках:

```
селектор {  
    свойство1: значение1;  
    свойство2: значение2;  
    ...  
}
```

Например, приведенные ниже правила CSS для заголовков 1-го уровня являются идентичными и любой из представленных вариантов записи считается допустимым.

|                                    |                                    |
|------------------------------------|------------------------------------|
| h1 {color: green; font-size: 20px} | h1 {color: green; font-size: 20px} |
|------------------------------------|------------------------------------|

При создании правил CSS следует грамотно выбрать селектор. Если стилевые параметры планируется применить ко всем элементам

определенного типа, то в качестве селектора будет использоваться **имя тега**. Если стилевые параметры следует назначить только некоторой группе элементов (элементам, принадлежащим к определенному классу), то селектором будет **.имя\_класса**. Если же стилевые параметры задаются для одного элемента с определенным именем (идентификатором), то селектором будет **#имя\_идентификатора**.

**Классы.** Если необходимо изменить способ отображения всех элементов одного и того же типа, записывают правило, селектором которого является соответствующий тег. Если же форматирование нужно применить лишь к некоторым элементам, следует создать **класс**. Тогда селектором в правиле будет имя класса (оно должно содержать только латинские буквы, цифры, символы дефиса (-) и подчеркивания (\_)). При записи правила для класса перед селектором ставится точка. Например:

```
.special {color: blue; font-size: 14 pt; text-align: center}
```

В данном примере описан класс **special**, задающий синий цвет шрифта, размер 14 пунктов и выравнивание текста по центру. Чтобы применить его к конкретному элементу, в соответствующий тег добавляют атрибут **class** и присваивают ему значение, соответствующее имени класса (без точки в начале):

```
<p class="special">  
<h1 class="special">
```

Если предполагается, что класс будет применяться только к элементам определенного типа, т. е. к конкретному тегу, то запись правила будет выглядеть:

```
p.special {color: blue; font-size: 14 pt; text-align: center}
```

В таком случае класс **special** можно применять только к абзацам.

**Идентификаторы.** Идентификатор определяет уникальное имя элемента для изменения его стиля. С помощью идентификатора можно обратиться к элементу через скрипт. Имя идентификатора должно содержать только латинские буквы, цифры, символы дефиса (-) и подчеркивания (\_). При записи правила для идентификатора перед селектором ставится символ #:

```
#vvedenie {font-style: italic; text-align: right}
```

Этот идентификатор можно применять только к одному элементу на странице, используя атрибут **id**, например:

```
<p id="vvedenie">
```

## Добавление стилей в HTML-документ: внедрение и связывание. Универсальный атрибут style.

Существуют два способа добавления стилей в HTML-документ:

1) таблица стилей задается непосредственно в HTML-документе (**внедрение**);

2) таблица стилей хранится в отдельном файле \*.css и подключается к HTML-документу (**связывание**).

В первом случае используется тег **<style>**, который располагается в служебной области HTML-документа **<head>** и задает стилевую информацию для текущего документа. Кроме того, можно использовать атрибут **style** для определения стилевых параметров конкретного элемента документа.

Стилевая информация может храниться непосредственно в HTML-документе, в служебной области между тегами **<style>**:

```
<head>
<style>
h2 {color: green; font-size: 20 pt}
p.special {color: blue; font-size: 14 pt; text-align: center}
#vvedenie {font-style: italic; text-align: right}
</style>
</head>
```

Такой способ добавления стилей называется **внедрением**. Заметим, что в этом случае стилевое форматирование работает только для одной web-страницы. Чтобы применить такое же форматирование для других страниц, в служебных областях соответствующих HTML-документов также должен присутствовать тег **<style>** с описанием тех же правил.

Чтобы избежать многократного повторения одних и тех же правил в разных документах, таблицу стилей можно хранить в отдельном файле (с расширением **.css**). В этом случае таблица стилей присоединяется к HTML-документу с помощью тега **<link>** в служебной части документа:

```
<link href="адрес файла css" rel="stylesheet" type="text/css">
```

Такой способ добавления стилей называется **связыванием**. В этом случае одну и ту же таблицу стилей можно присоединить к нескольким.

## Контрольные вопросы

1. Какие объекты могут содержаться внутри документа HTML?
2. Чем отличаются парные и непарные теги?
3. Каковы основные правила записи тегов и их атрибутов?
4. Какие теги определяют служебную и содержательную области документа HTML?
5. Назовите способы указания цвета в HTML.
6. Какие теги применяются для физического форматирования текста? Укажите недостатки физического форматирования текста.
7. Что представляет собой логическое форматирование текста?
8. Для чего служит тег `<img>`? Почему атрибут `src` этого тега является обязательным?
9. Как задать таблицу в HTML?
10. Для чего используются Meta-теги и каковы их атрибуты?

## ЛИТЕРАТУРА

1. Мамаев, М. Технологии защиты информации в Интернете: специальный справочник / М. Мамаев, С. Петренко. – Санкт-Петербург: Питер, 2002. – 848 с.
2. Таненбаум, Э. Компьютерные сети / Э. Таненбаум. – 5-е изд. – Санкт-Петербург: Питер, 2019. – 960 с. ил. – (Классика Computer Science).
3. Олифер, В. Г. Компьютерные сети: принципы, технологии, протоколы: учеб. пособие / В. Г. Олифер, Н. А. Олифер. – 4-е изд. – Санкт-Петербург: Питер, 2012. – 944 с. ил. – (Учебник для вузов).

## Тема 4. ТЕХНОЛОГИИ БАЗ ДАННЫХ

- 4.1. Базы данных: понятие, назначение и архитектура.
- 4.2. Модели данных.
- 4.3. Системы управления базами данных (СУБД). Обзор современных СУБД и направления их развития.
- 4.4. Функциональные возможности СУБД на примере MS Access.

### 4.1. Базы данных: понятие, назначение и архитектура

**База данных** (БД) – это именованная совокупность данных, отображающая состояние объектов, их свойства и взаимоотношения в некоторой предметной области.

**Предметная область** – это часть реального мира, которая описывается и моделируется с помощью БД.

**Объект** – предмет, вещество, событие, лицо, явление, абстрактное понятие, т. е. все то, что может характеризоваться набором значений некоторой совокупности атрибутов.

**Атрибут** – это информационное отображение свойства объекта. *Преимущества БД* по сравнению с файловой организацией данных:

1) БД является *информационной моделью объекта*, от обоснованности, точности и достоверности которой во многом зависит эффективность управления объектом;

2) в БД информация *хранится централизованно*. Многие пользователи могут иметь возможность доступа, просмотра и изменения данных в одно и то же время, при этом пользуясь самой последней версией информации. Легче изменять и согласовывать данные. Экономится дисковое пространство;

3) организация данных в базах дает возможность *быстрого поиска, отображения и анализа данных*;

4) БД представляет собой новый подход к организации данных. Возможно *обращение к данным без знания физического расположения их в памяти компьютера*, вследствие чего доступ к данным и их обработка более просты. Разработка прикладных программ, использующих БД, становится проще, быстрее, дешевле.

Современные БД имеют большие объемы, измеряемые в Терабайтах (1 Тб = 1024 Гб) и даже Петабайтах (1 Пб = 1024 Тб). Для их названия появился специальный термин – «сверхбольшие БД».



*Приложения БД:*

**запросы** – требования пользователя на отбор данных из базы и (или) на выполнение определенных действий;

**формы** – чаще всего шаблоны, применяемые для ввода, просмотра и редактирования данных БД;

**отчеты** – представления информации из БД в виде, удобном для ее восприятия и анализа пользователем;

**web-страницы** – средства для публикации БД в сети Интернет;

**прикладные программы** – программы, работающие с БД, и написанные на языке программирования, встроенном в СУБД.

*Компоненты БД:*

1) *данные пользователей;*

2) *метаданные.* СУБД производит описание структуры БД (метаданные). **Метаданные** – данные о данных (хранятся в системных таблицах). По-другому метаданные называют *словарем данных* или *каталогом данных*;

3) *данные, призванные улучшить производительность и доступность БД.* Они состоят главным образом из индексов;

4) *метаданные приложений.* Это описания структуры и формата пользовательских запросов, форм, отчетов и других приложений, выполненные СУБД.

**Пользователь БД** – лицо или прикладная программа, которые могут обращаться с командами и (или) запросами к БД и получать от нее результаты обращений. Люди, работающие с БД, – это конечные пользователи БД и обслуживающий персонал. **Конечные пользователи БД** – это специалисты предметной области, которым требуется информация из БД для выполнения прямых служебных обязанностей (например, бухгалтер, руководитель, менеджер отдела продаж). **Обслуживающий персонал** – люди, ответственные за работу БД (*администраторы БД*) и разработку соответствующего прикладного программного обеспечения (*разработчики прикладного ПО*).

**Структуры данных.** Исследования в области БД прежде были направлены на разработку способов структуризации данных.

**Структура данных** – программная единица, позволяющая хранить и обрабатывать множество однотипных и (или) логически связанных данных.

Данные бывают двух типов:

*основной* (простой) – форма представления определяется архитектурой компьютера. Данные простого типа – это символы, числа, т. е. элементы, дробление которых не имеет смысла;

*сложный* – конструируемый пользователем для решения конкретных задач. Сложные типы данных (структуры данных) формируются из элементарных данных.

Некоторые структуры: **массив** – простая совокупность элементов данных одного типа; **запись** – совокупность элементов данных разного типа. В простейшем случае записи содержат одинаковое количество элементов, называемых полями. Записи имеют идентификатор (уникальное имя или номер), называемый **ключом**. Совокупность записей одинаковой структуры называется **файлом**.

Массив, запись – это *статические* структуры, так как занимают в памяти компьютера постоянный объем.

*Динамические* структуры – это дерево, список, ссылка. Существует большое количество данных, которые могут быть представлены как **деревья** – для их хранения требуется нелинейное адресное пространство.

**Список** – это упорядоченный набор значений, в котором некоторое значение может встречаться более одного раза.

**Ссылка** – это объект, указывающий на определенные данные, но не хранящий их (это может быть переменная, содержащая адрес ячейки).

Более сложные структуры могут включать массив, дерево, запись в качестве элементов данных.

Существует большое разнообразие сложных типов данных.

## 4.2. Модели данных

**Модель данных** – формальная теория представления и обработки данных в системах управления базами данных (СУБД), которая включает, по меньшей мере, аспекты:

- 1) *структуры*: методы описания типов и логических структур данных в БД (точка зрения пользователя на представление данных);
- 2) *манипуляции*: методы манипулирования данными (набор допустимых операций, выполняемых на структуре данных);
- 3) *целостности*: методы описания и поддержки целостности БД (механизм поддержания соответствия данных предметной области на основе формального описания правил).

**Модель данных** – это совокупность принципов организации БД. Важнейший из них – это принцип связи объектов в БД.

Классическими моделями данных являются иерархическая, сетевая и реляционная модели данных.

**Иерархическая модель.** В иерархической модели связи между данными можно представить с помощью дерева (рис. 4.1).

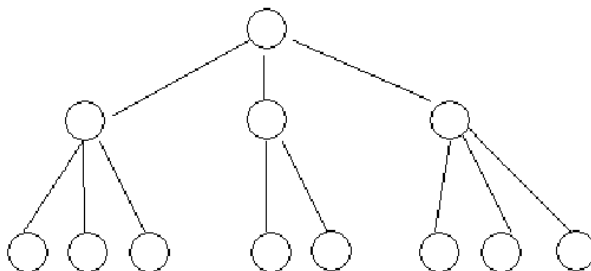


Рис. 4.1. Пример организации данных по иерархической модели

Данные расположены на разных иерархических уровнях и называются **сегментами**. Самый высокий сегмент – *корневой*. Сегменты на более низком уровне – *сегменты-потомки*. Сегменты на более высоком уровне – *сегменты-предки*.

Каждый сегмент может иметь только одного предка на более высоком уровне и одного или несколько потомков на более низком уровне.

Доступ к определенному сегменту осуществляется по цепочке, от сегмента-предка к сегменту-потомку, начиная слева.

*Достоинства иерархической модели:* эффективное использование памяти компьютера; неплохие показатели времени выполнения основных операций над данными.

*Недостатки:* громоздкость для обработки данных со сложными логическими связями; сложность понимания обычным пользователем; медленный доступ к данным нижних уровней иерархии; четкая ориентация на определенные типы запросов.

**Сетевая модель.** Сетевая модель – развитие иерархической модели. В ней потомок может иметь любое количество предков. Сегменты – наборы записей об объектах – связываются между собой не только по принципу сверху вниз, но и по горизонтали с помощью наборов записей о связях. Любой сегмент может быть связан с любым другим. На рис. 4.2 показан пример организации данных в сетевой модели.

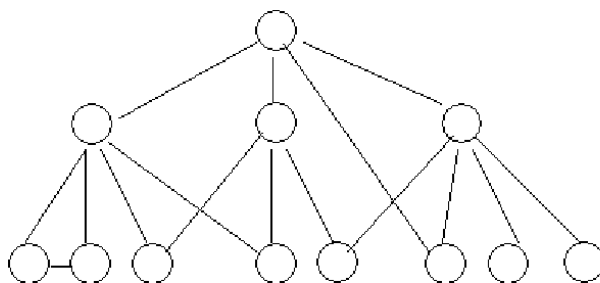


Рис. 4.2. Пример организации данных в сетевой модели

*Достоинство сетевой модели:* возможность эффективной реализации БД: по показателям затрат памяти; оперативности доступа к данным.

*Недостатки:* сложность и жесткость схемы БД, построенной на ее основе; сложность понимания и обработки информации в БД обычным пользователем; ослаблен контроль целостности связей вследствие допустимости установления произвольных связей между записями.

**Реляционная модель.** В основе реляционной модели данных (РМД) лежит понятие отношения. Отношение отображает некоторый объект. Объект характеризуется набором атрибутов  $x_1, x_2, \dots, x_n$ , а каждый атрибут – набором допустимых значений, называемым **доменом**. Пусть

$$\begin{aligned} D_1 &= \{x_1, x_2, \dots, x_k\}; \\ D_2 &= \{y_1, y_2, \dots, y_l\}; \\ &\dots\dots\dots \\ D_n &= \{z_1, z_2, \dots, z_m\}. \end{aligned}$$

Список имен атрибутов ( $x_1, x_2, \dots, x_n$ ) называется **схемой отношения**, а количество атрибутов в отношении – **степенью отношения**.

**Отношение** – это подмножество  $R$  декартова произведения

$$D_1 \times D_2 \times \dots \times D_n, \text{ т. е. } R \subseteq D_1 \times D_2 \times \dots \times D_n.$$

**Декартово произведение** – это набор всевозможных сочетаний из  $n$  значений, где каждое значение берется из своего домена.

Пусть атрибут *Номер заказа* имеет домен  $D_1 = \{1021, 1022, 1023\}$ , атрибут *Код клиента* – домен  $D_2 = \{AA, AC\}$ , атрибут *Стоимость заказа*, млн. руб. – домен  $D_3 = \{100, 300, 120\}$ .

Тогда отношение  $R$  есть декартово произведение  $D_1 \times D_2 \times D_3$  – набор из 18 троек значений. Его можно представить как таблицу вида:

| Номер заказа | Код клиента | Стоимость заказа, млн. руб. |
|--------------|-------------|-----------------------------|
| 1021         | AA          | 100                         |
| 1021         | AA          | 300                         |
| 1021         | AA          | 120                         |
| 1021         | AC          | 100                         |
| 1021         | AC          | 300                         |
| 1021         | AC          | 120                         |
| 1022         | AA          | 100                         |
| 1022         | AA          | 300                         |
| 1022         | AA          | 120                         |
| 1022         | AC          | 100                         |
| 1022         | AC          | 300                         |
| 1022         | AC          | 120                         |
| 1023         | AA          | 100                         |
| 1023         | AA          | 300                         |
| 1023         | AA          | 120                         |
| 1023         | AC          | 100                         |
| 1023         | AC          | 300                         |
| 1023         | AC          | 120                         |

Термин «отношение» – синоним слова «таблица».

Столбцы таблицы соответствуют атрибутам. Строки называются **кортежами**. Количество кортежей в отношении – **мощность отношения**.

**Реляционная модель данных** (РМД) – модель данных, основанная на математическом понятии отношения и представлении отношений в форме таблиц.

Реляционная таблица должна обладать *следующими свойствами*.

1. Каждое значение атрибута на пересечении строки и столбца должно быть **атомарным**.

2. Значения в столбце должны быть однородными.

3. Каждая строка уникальна.

4. Каждый столбец имеет уникальное имя.

5. Последовательность столбцов в таблице несущественна.

6. Последовательность строк в таблице несущественна.

Пример реляционной таблицы

КЛИЕНТ

| Код клиента | Клиент     | Адрес                       |
|-------------|------------|-----------------------------|
| АА          | МАЗ        | Минск, пр. Партизанский, 76 |
| АБ          | Элема      | Минск, ул. Тростенецкая, 5  |
| АС          | Коммунарка | Минск, ул. Аранская, 61     |
| АД          | МТЗ        | Минск, ул. Долгобродская, 6 |

В реляционной таблице *столбцы* называют **полями**, а *строки* – **записями**. Одно или несколько полей, значения которых в каждой записи таблицы однозначно ее идентифицируют, называют **ключевым полем**.

В таблице КЛИЕНТ таковым может быть поле «Код клиента» или поле «Клиент».

В реляционной БД между таблицами устанавливаются связи. **Связь** устанавливается посредством связи ключевых полей, содержащих общую информацию для обеих таблиц.

Одна запись главной таблицы может быть связана с одной или несколькими записями подчиненной таблицы. При этом значения первичного ключа уникальны, а внешнего – могут повторяться.

В общем случае РМД представляет собой **множество взаимосвязанных таблиц**. Графическое изображение связи между таблицами называется **схемой данных**.

Пусть имеется еще таблица ЗАКАЗ вида:

ЗАКАЗ

| Номер заказ | Код клиента | Дата заказ | Стоимость заказ, млн. руб. |
|-------------|-------------|------------|----------------------------|
| 1020        | АБ          | 01.02.15   | 400                        |
| 1021        | АА          | 01.02.15   | 100                        |
| 1022        | АС          | 12.02.15   | 300                        |
| 1023        | АА          | 20.06.15   | 120                        |
| 1024        | АБ          | 28.05.15   | 600                        |

Нужно связать таблицы КЛИЕНТ и ЗАКАЗ. Это можно сделать по полю «Код клиента». В таблице КЛИЕНТ оно играет роль *первич-*

ного ключа, а в таблице ЗАКАЗ – внешнего ключа. На рис. 4.3 показана схема данных.

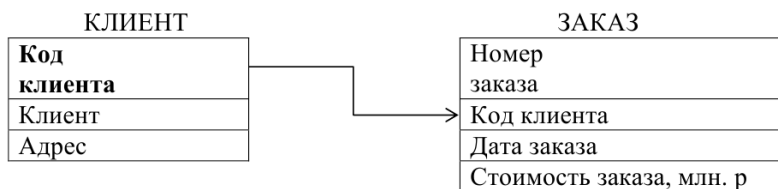


Рис. 4.3. Схема данных

В РМД должны выполняться условия целостности данных:

- 1) *целостность таблиц* накладывает ограничения на значения первичного ключа – они должны быть *уникальными* и *непустыми*.
- 2) *ссылочная целостность* предполагает, что *каждое значение внешнего ключа должно совпадать с одним из значений первичного ключа*.

В процессе обработки реляционных таблиц СУБД выполняет операции реляционной алгебры, такие как:

- традиционные операции над множествами: объединение, пересечение, декартово произведение, разность;
- специальные операции: проекция, выборка, соединение, деление.

*Достоинства РМД*: проста для понимания; наглядна; имеет строгое математическое обоснование.

*Недостатки*: не допускает представления объектов со сложной структурой, так как в ее рамках возможно моделирование лишь с помощью двумерных таблиц; данные об объектах содержатся, как правило, во многих таблицах. Извлечение информации о каждом объекте требует выполнения многих операций соединения с помощью первичных и внешних ключей, что значительно замедляет обработку данных.

В последнее время активно используются при разработке БД такие модели, как *постреляционная*, *объектно-ориентированная*, *объектно-реляционная* и *многомерная модели*.

**Постреляционная модель** – в общем случае представляет собой расширенную реляционную модель, снимающую ограничение неделимости значений полей, т. е. допускаются многозначные поля, значения которых состоят из подзначений. В случае постреляцион-

ной модели данные о заказах клиентов из двух связанных реляционных таблиц КЛИЕНТ и ЗАКАЗ можно представить одной таблицей:

| Код клиента | Номер заказа | Клиент     | Адрес                   | Дата заказ | Вес заказа |
|-------------|--------------|------------|-------------------------|------------|------------|
| АА          | 1021         | МАЗ        | Минск,                  | 01.02.15   | 100        |
|             | 1023         |            | пр. Партизанский, 76    | 20.06.15   | 120        |
| АБ          | 1020         | Элема      | Минск,                  | 01.02.15   | 400        |
|             | 1024         |            | ул. Тростенецкая, 5     | 28.05.15   | 600        |
| АС          | 1022         | Коммунарка | Минск, ул. Аранская, 61 | 12.02.15   | 300        |

*Достоинства* постреляционной модели данных: возможность представления связанных реляционных таблиц одной постреляционной таблицей, что обеспечивает высокую наглядность представления данных и повышение эффективности их обработки; отсутствие ограничений на длину полей и их количество в записях таблицы. *Недостаток* – сложность в обеспечении целостности данных.

**Объектно-ориентированная и объектно-реляционная модели** – используются для преодоления ограниченных возможностей РМД по хранению и обработке сложных объектов, как, например, документ, звук, видео, графическое изображение и др.

**Объектно-ориентированная модель** (ООМД) – представляет структуру, которую можно изобразить графически в виде дерева, узлами которого являются объекты. Каждый объект характеризуется уникальным *идентификатором, состоянием и поведением*.

**Состояние** объекта – определяется множеством значений его атрибутов.

**Поведение** объекта описывают *методы*, называемые **процедурами**, т. е. составной частью описания объекта являются процедуры, способные производить действия над атрибутами объекта в случае наступления тех или иных событий.

Объекты могут объединяться в **классы**. Экземпляры одного класса отличаются лишь значениями своих свойств. Методы устанавливаются для класса.

Для выполнения действий над объектами применяются объектно-ориентированные механизмы – *наследование, инкапсуляция, полиморфизм*.

**Наследование** – на основе существующего класса можно образовать новый класс объектов, наследующий свойства родительского класса.



**Инкапсуляция** – доступ к данным осуществляется только лишь в соответствии с правилами поведения объекта, описываемыми методами.

**Полиморфизм** – способность объектов по-разному реагировать на одно и то же событие в окружающем мире. Полиморфизм используется для унификации обработки разнородных объектов. Например, метод «Печать результата» может быть определен для многих классов объектов, но работать по-разному, в зависимости от класса, к которому он применяется.

**Достоинство** ООМД – способность отображать информацию о сложных объектах с исчерпывающим описанием взаимосвязей между ними и их динамического поведения. ООМД обычно применяется для сложных предметных областей, при моделировании которых не хватает функциональности РМД.

**Недостаток** – сложность понятийного аппарата, что усложняет ее применение и отрицательно сказывается на накоплении опыта создания и эксплуатации объектно-ориентированных БД.

**Объектно-реляционная модель данных (ОРМД)** – является гибридной моделью, сочетающей возможности реляционной модели с объектными свойствами данных.

Существует два подхода к ее созданию.

1. Объекты, видимые на внешнем интерфейсе, отображаются в таблицы реляционной БД. И наоборот, объекты воспроизводятся из их представления в табличной среде хранения, когда они запрашиваются пользователями или приложениями (*гибридный подход*).

2. Внутренние реляционные механизмы СУБД для управления данными расширяются объектно-ориентированными возможностями (*расширенный реляционный подход*). Этот подход технологически более продвинутый и предпочитаемый в настоящее время большинством разработчиков реляционных СУБД.

Отличительная особенность ОРМД от ООМД состоит в том, что она основана на стратегии реляционной модели.

**Многомерная модель** – означает *многомерное логическое представление структуры информации*, а не многомерность визуализации данных. Многомерная модель предназначена для *аналитической* обработки информации, в отличие от предыдущих моделей. В ней используются такие понятия, как агрегируемость, историчность, прогнозируемость данных.

**Агрегируемость** данных – возможность их рассмотрения с различным уровнем обобщения.

**Историчность** – высокий уровень статичности (неизменяемости) данных и их взаимосвязей, а также обязательная привязка данных к временным точкам.

**Прогнозируемость** данных – применение функций их прогнозирования к различным интервалам времени.

Представление данных о продажах автомобилей:

*по реляционной модели:*

| Марка автомобиля | Месяц   | Объем продаж, шт. |
|------------------|---------|-------------------|
| Ауди             | Январь  | 12                |
| Ауди             | Февраль | 24                |
| Ауди             | Март    | 5                 |
| Пежо             | Январь  | 2                 |
| Пежо             | Февраль | 18                |
| BMW              | Февраль | 19                |

*многомерной модели:*

| Марка автомобиля | Январь | Февраль | Март |
|------------------|--------|---------|------|
| Ауди             | 12     | 24      | 5    |
| Пежо             | 2      | 18      | 0    |
| BMW              | 0      | 19      | 0    |

Основные понятия в многомерной модели – *измерение* и *ячейка*.

**Измерение** – это множество однотипных данных, образующих одну из граней многомерного гиперкуба. Наиболее часто используются *временные* измерения – дни, месяцы, кварталы и годы. В качестве *географических* измерений широко употребляются города, районы, регионы и страны.

**Ячейка** – это поле, значение которого однозначно определяется фиксированным набором измерений. В вышеприведенном примере значение ячейки объема продаж однозначно определяется комбинацией временного измерения Месяц и измерения Марка автомобиля.

Многомерную модель, отображающую объемы продаж автомобилей менеджерами по годам, можно представить уже в виде трехмерного куба (рис. 4.4).

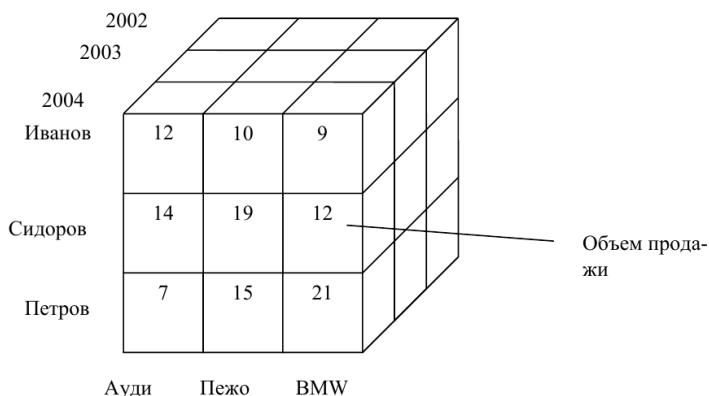


Рис. 4.4. Трехмерная модель данных о продажах автомобилей

В многомерной модели данных используется два варианта организации данных – гиперкубическая и поликубическая. В **гиперкубической** все кубы определяются одним и тем же набором измерений (максимально возможным). В **поликубической** – несколько гиперкубов различной размерности и различными измерениями в качестве граней.

Для извлечения данных из БД, организованной по многомерной модели, применяются специальные операции: срез, вращение, агрегация и детализация.

**Срез** – данные, полученные в результате фиксации одного или нескольких измерений.

**Вращение** – процедура изменения порядка следования измерений. Применяется в основном при двухмерном представлении данных.

**Агрегация и детализация** – соответственно переход к более или менее детальному представлению информации из гиперкуба.

**Достоинство** многомерной модели – удобство и эффективность аналитической обработки больших объемов данных, связанных с временными интервалами.

**Недостаток** – громоздкость для простейших задач оперативной обработки информации.

### **4.3. Системы управления базами данных (СУБД). Обзор современных СУБД и направления их развития**

СУБД существуют практически для всех аппаратных платформ (от карманных компьютеров до суперкомпьютеров) и используются во всех автоматизированных информационных системах как средство работы с БД.

**Система управления базами данными (СУБД)** – это совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями.

**Языковые средства:** язык описания данных (ЯОД), язык манипулирования данными (ЯМД), язык запросов, язык программирования. ЯОД – средства для описания структуры данных, их типов, свойств и ограничений для данных. ЯМД позволяет вставлять, обновлять, удалять и извлекать данные.

Современные языки запросов позволяют выполнять все операции над данными БД. Пример – язык SQL.

Некоторые СУБД располагают языками, которые используются и для обращения к данным БД, и для написания прикладных программ. Например, xBase. Для глубокой аналитической обработки данных могут создаваться программы на традиционных языках программирования, для которых конкретная СУБД обеспечивает доступ к БД.

**Программные средства** СУБД обеспечивают работу с физической БД (поиск, изменение, обработка, извлечение данных и др.) и выполнение всех ее функций.

#### **Функциональные возможности СУБД.**

Современные СУБД предоставляют пользователям следующие возможности:

- создание БД;
- обновление хранящихся в ней данных;
- быстрое извлечение из БД необходимых данных по запросам пользователей; выполнение вычислений над данными;
- создание форм для удобства работы с данными БД;
- создание отчетов на основе информации БД для ее анализа пользователями; разработка приложений;
- экспорт/импорт данных в БД из других приложений;
- публикация БД в Интернете;

- управление БД;
- защита данных от несанкционированного доступа, от сбоев в работе компьютерной системы;
- восстановление БД в случае ее повреждения.

**Архитектура СУБД** – это совокупность функциональных компонентов системы и их взаимосвязей. СУБД включает 3 компонента:

*подсистему средств проектирования* БД и ее приложений;

*подсистему обработки* компонентов приложений;

*ядро* СУБД, которое является посредником между данными подсистемами и БД, участвует в управлении транзакциями, блокировке, резервном копировании и восстановлении. **Транзакция** – совокупность операций с БД, которые должны быть выполнены обязательно до конца, чтобы БД осталась в непротиворечивом состоянии.

**Классификация СУБД.**

*По степени универсальности* различают СУБД:

- общего назначения;
- специального назначения.

*По типу поддерживаемой модели данных:*

– *иерархические*. Типичным представителем является Information Management System (IMS) фирмы IBM;

– *сетевые*. Типичным представителем является Integrated Database Management System (IDMS) компании General Electric;

– *реляционные*. Первые коммерческие реляционные СУБД – от компаний IBM, Oracle Corporation и др.;

– *постреляционные* – uniVers, Bubba, Dasdb и др.;

– *объектно-ориентированные* – ORION, IRIS, Vbase, PDM и др.;

– *объектно-реляционные* – Informix Universal Server (Informix Software), DB2 Universal Database (IBM) и др.;

– *многомерные* – Oracle Express Server (Oracle), Cache (InterSystems) и др.

*По принципу обработки запросов к БД:*

- настольные;
- серверные;
- распределенные.

**Режимы работы пользователя в СУБД.** Пользователи могут работать с БД в СУБД:

- *через меню системы;*
- *в командном режиме;*
- *программном режиме.*

Работа с локальной БД осуществляется в *однопользовательском* режиме, с централизованной БД – в *многопользовательском*.

СУБД является программной основой автоматизированной информационной системы. При ее выборе учитывают следующие факторы:

- скорость выполнения операций корректировки данных;
- время выполнения запросов (время отклика);
- время генерации отчета;
- максимальное число одновременных обращений к БД в многопользовательском режиме и др.

В 1988 г. была создана международная организация ТРС (Transaction Processing Performance Council), которая разработала ряд *эталонных тестов* для сравнительной оценки производительности коммерческих СУБД.

**Функции СУБД.** СУБД, работая с физической БД (с БД на устройстве внешней памяти), выполняют:

– **управление:**

- *данными во внешней памяти.* Эта функция дает возможность пользователям выполнять основные операции с данными – сохранение, извлечение, обновление;

- *транзакциями.* **Транзакция** – логическая единица работы, включающая несколько команд вставки, удаления или модификации, которая переводит БД из одного завершенного состояния в другое завершенное состояние, которое не нарушает целостности этих данных (все данные в таблицах БД правильны, а ссылки между таблицами корректны). В зависимости от того, какие команды используются, транзакции есть следующих *типов*: только для записи; только для модификации; только для чтения; только для удаления. Транзакции только для чтения называют *запросом*;

- *параллельным доступом*, чтобы избежать конфликта с нежелательными последствиями при попытке обновления данных;

- *буферами оперативной памяти*;

– **поддержку:**

- *языков баз данных*;

- *обмена данными* (экспорт/импорт);

- *целостности данных*;

- *независимости прикладных программ от данных.*

– **ведение системного каталога**;

– **контроль доступа к данным.**

Должен быть только санкционированный доступ.

**Направления развития СУБД.** В настоящее время можно говорить о следующих направлениях развития СУБД:

- расширение множества типов обрабатываемых данных;
- комбинирование технологий WWW и технологий БД;
- переход к активным БД.

**Настольные СУБД. Архитектура файл-сервер.**

*Настольные* СУБД использовались для работы с локальными БД на ПК, а с появлением компьютерных сетей – для работы с централизованными БД в архитектуре файл-сервер.

*Архитектурой файл-сервер* называется архитектура с совместным использованием файлов. В ней компьютеры объединены в сеть. На *файловом сервере* сети размещается БД, а на *рабочих станциях* устанавливаются настольные СУБД. СУБД на рабочей станции запрос пользователя или прикладной программы посылает к БД на сервере. По запросу из БД выбираются необходимые для его выполнения *таблицы целиком* и пересылаются на рабочую станцию, где СУБД выполняет запрос.

*Недостатки архитектуры файл-сервер:*

- по сети передается гораздо больший объем данных, чем реально нужно для выполнения запроса, и сеть сильно перегружается;
- выполнение запросов и управление целостностью БД ложится на СУБД пользователей, что является причиной невысокой безопасности работы. Секретность и конфиденциальность информации обеспечить также трудно; пользователи могут формировать запросы и на внесение изменений в БД. При этом блокируются записи, которые изменяются одним из пользователей, чтобы в это время другой пользователь не внес изменений в те же данные. Из-за этого настольные СУБД редко используются для обработки БД больших объемов, ориентированных на транзакции.

**Характеристика настольных СУБД.**

Популярные настольные СУБД: dBase, Paradox, MS FoxPro, MS Access. СУБД семейства **dBase** благодаря простоте в использовании, нетребовательности к ресурсам компьютера, комфортной среде разработки приложений БД, популярному формату данных и популярному языку программирования xBase имели немалую популярность. Последние версии dBase имеют средства, присущие современным СУБД. Тенденция такова, что dBase превращается в некоммерческий продукт с доступными исходными текстами программ.

Ранние версии **Paradox** предоставляли более широкие возможности, чем аналогичные версии dBase, – средства статистического анализа данных, создание приложений на языке PAL с возможностью визуального построения пользовательских интерфейсов. Windows-версии этой СУБД использовались как универсальное средство управления различными БД. Последние версии являются компонентом Corel Office Professional и содержит средства, свойственные современным СУБД. Популярность Paradox в настоящее время снизилась, хотя в мире эксплуатируется еще много информационных систем на ее основе.

**MS FoxPro** происходит от настольной СУБД FoxBase (Fox Software), которая впоследствии была приобретена Microsoft. Последние версии имеют средства визуального моделирования объектов, средства публикации данных в Интернете и др. С каждой новой версией этот продукт все более интегрируется с другими продуктами Microsoft. Тенденция развития этой СУБД – из настольной СУБД превращается в средство разработки приложений в архитектуре клиент/сервер и распределенных приложений.

**MS Access** была первой настольной реляционной СУБД для 16-разрядной версии Windows. Ее популярность значительно возросла после включения в состав MS Office. Эта СУБД ориентирована на непрофессиональных пользователей MS Office. В ней вся информация, относящаяся к конкретной БД, хранится в одном файле, что удобно для начинающих пользователей. Последние версии могут быть использованы, с одной стороны, в качестве настольной СУБД, а с другой – в качестве клиента MS SQL Server.

*Достоинства настольных СУБД:* просты для освоения и использования; имеют дружелюбный пользовательский интерфейс; ориентированы на класс самых распространенных компьютеров – персональных и на самую широкую категорию пользователей – непрофессионалов; обеспечивают хорошее быстродействие при работе с небольшими БД.

*Недостатки:* с увеличением объемов БД и числа их пользователей снижается производительность СУБД, и они дают сбои при обработке данных.

### **Серверные СУБД. Архитектура клиент-сервер.**

Серверные СУБД используются в архитектуре клиент/сервер. Базовый принцип этой архитектуры – централизация хранения и обработки данных. На *сервере сети* размещается база данных и



устанавливается мощная серверная СУБД или **сервер БД** – программный компонент, обеспечивающий хранение больших объемов информации, ее обработку и представление пользователям в сетевом режиме. На *клиентских компьютерах* устанавливаются клиентские приложения.

Выполнение запроса к БД в архитектуре клиент-сервер представлено на рис. 4.5.

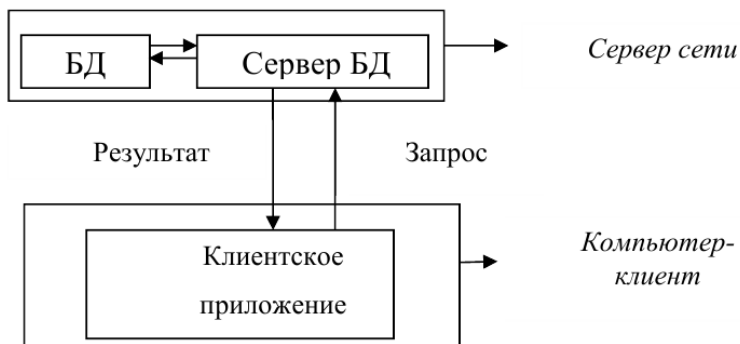


Рис. 4.5. Выполнение запроса к БД в архитектуре клиент-сервер

#### **Функции сервера БД:**

- интерпретирует, выполняет запрос, формирует его результат и пересылает по сети на клиентский компьютер;
- управляет целостностью БД;
- обеспечивает одновременную безопасную, отказоустойчивую многопользовательскую работу с одними и теми же данными;
- выполняет хранение и резервное копирование данных;
- реализует бизнес-правила (установки, принятые в предметной области).

#### **Преимущества архитектуры клиент-сервер:**

- уменьшается сетевой трафик;
- уменьшается потребность клиентских приложений в оперативной памяти;
- существенно повышается степень безопасности БД.

#### **Современные серверы БД. Характеристики современных серверов БД:**

- многоплатформенные;

- наличие нескольких версий для решения различных по масштабу задач; высокие показатели надежности и доступности;
- параллельная обработка данных в многопроцессорных системах; распределенные запросы и транзакции;
- стандартные механизмы доступа к данным (ODBC, JDBC, OLE DB, ADO.NET);
- средства разработки клиентских приложений;
- простота администрирования;
- несколько сценариев репликаций;
- создание хранилищ данных и OLAP.

### ***Популярные серверные СУБД:***

#### **ORACLE:**

- имеет большое количество различных версий и типов;
- поддерживает свыше 80 вариантов операционной среды;
- функционирует на большинстве компьютерных платформ (мэйнфреймы, Unix-серверы, ПК и др.);
- поддерживает все варианты архитектур компьютеров, в том числе многопроцессорные системы;
- хранит и обрабатывает различные типы данных (текстовых документов, графических образов более 20 форматов, видео- и аудио-информации);
- имеет язык запросов – PL-SQL;
- имеет множество различных компонентов и модулей.

#### **MS SQL Server:**

- реляционная СУБД;
- имеет язык запросов – Transact-SQL;
- используется для работы с БД от персональных до крупных масштабов предприятия.

#### **СУБД My SQL:**

- свободно распространяемая реляционная СУБД с открытой архитектурой. Поддержку осуществляет корпорация Oracle;
- обычно используется в качестве сервера, к которому обращаются локальные и удаленные клиенты;
- поддерживает большое количество таблиц, различающихся системой хранения данных;
- благодаря открытой архитектуре постоянно появляются новые типы таблиц.

## **Распределенные БД и СУБД.**

**Распределенная база данных (РаБД)** – совокупность логически взаимосвязанных БД, распределенных в компьютерной сети. В РаБД данные распределены по узлам компьютерной сети. Каждый узел имеет собственную БД и может обращаться к данным, хранящимся на других узлах. Пользователь распределенной БД не обязан знать, каким образом ее компоненты размещены в узлах сети и представляет себе эту БД как единое целое.

РаБД могут быть:

- *однородными* (гомогенными), имеющими в основе одну СУБД, обычно с единственным языком БД;
- *неоднородными* (гетерогенные), имеющими в основе две или более существенно различающихся СУБД.

### **4.4. Функциональные возможности СУБД на примере MS Access**

Microsoft Access 2007 – профессиональная программа управления базами данных. С ее помощью можно накапливать и систематизировать разнообразную информацию, искать и сортировать объекты согласно выбранным критериям, конструировать удобные формы для ввода данных и генерировать на основании имеющихся записей прекрасно оформленные отчеты.

**Типы данных.** Для того чтобы обеспечить возможность хранения в базе данных разнообразной информации, Access предлагает большой набор типов данных, представленных в табл. 4.1.

Таблица 4.1. **Типы данных MS Access**

| Название типа    | Назначение                                                                                                                    |
|------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Текстовый        | Текст длиной до 255 символов                                                                                                  |
| Поле МЕМО        | Текст длиной до 65 000 символов                                                                                               |
| Числовой         | Числа различных форматов                                                                                                      |
| Дата/время       | Дата и (или) время                                                                                                            |
| Денежный         | Денежные значения различных форматов                                                                                          |
| Счетчик          | Счетчик, который автоматически увеличивается на единицу с добавлением каждой новой записи                                     |
| Логический       | Величины, способные принимать только два значения: да/нет                                                                     |
| Поле объекта OLE | Поля, позволяющие вставлять рисунки, звуки и данные других типов                                                              |
| Гиперссылка      | Ссылки, дающие возможность открывать объект Access (таблицу, форму, запрос и т. д.), файл другого приложения или web-страницу |

**Основные понятия СУБД Microsoft Access.** База данных под управлением СУБД Access – это файл с расширением *.accdb*, содержащий следующие объекты: таблицы для хранения данных, формы для ввода и редактирования данных в интерактивном режиме, запросы для обработки таблиц и других запросов, отчеты для вывода результатов обработки данных, макросы и модули для автоматизации выполнения рутинных действий.

**Интерфейс пользователя.** Окно Microsoft Access (рис. 4.6) содержит элементы, стандартные для приложений Microsoft Office 2007: строку заголовка, ленту, рабочую область, строку состояния.

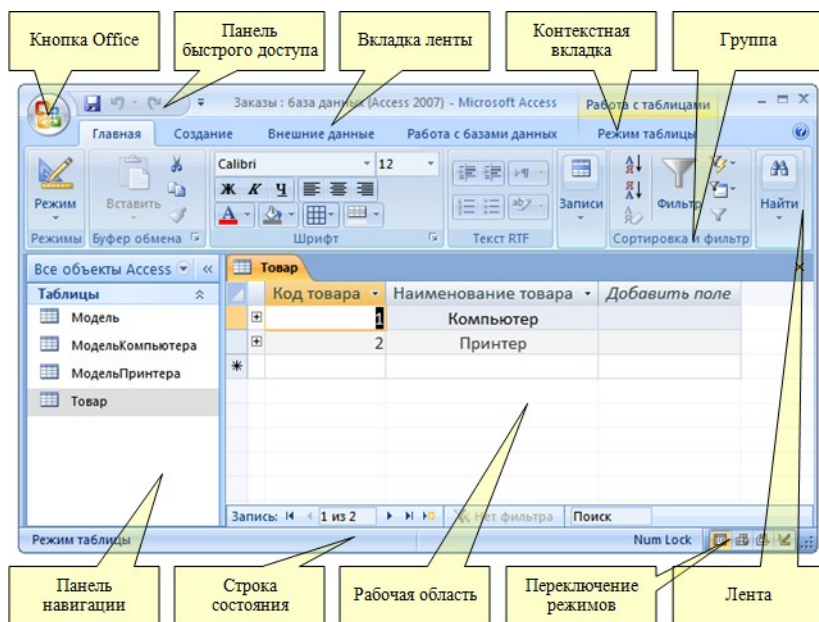


Рис. 4.6. Интерфейс Microsoft Access

В строке заголовка отображается имя открытой базы данных (дополнительно показываются ее формат и режим), а также кнопки управления окном программы. Строка заголовка расположена в верхней части окна и немного смещена вправо. В левой части расположены панель быстрого доступа и кнопка «Office».

**Кнопка «Office»** – открывает меню для выполнения основных операций с файлом базы данных, а также список последних документов (баз данных), с которыми работал пользователь.

**Панель быстрого доступа** – содержит кнопки для выполнения часто используемых команд. Эта панель настраиваемая. С помощью кнопки со стрелкой, направленной вниз, открывается ее меню, состоящее из команд, которые можно в нее добавить или убрать из нее.

**Лента** – элемент интерфейса, содержащий все элементы управления, с помощью которых выполняются команды и задания в Access 2007. Элементы расположены на тематических вкладках, вкладки бывают основные, или постоянные, и дополнительные, или контекстные.

1. Основные вкладки – **Главная, Создание, Внешние данные, Работа с базами данных** – содержат постоянно доступные команды и инструменты.

2. Контекстные вкладки – **Конструктор, Режим таблицы, Формат, Упорядочить** и др., появляются на ленте по мере необходимости. Количество отображаемых контекстных вкладок и их названия связаны с текущим объектом базы данных, его режимом или задачами, которые можно выполнить.

**Панель сообщения** – отображает важные сообщения системы, например, предупреждения системы безопасности.

**Панель навигации** (Область переходов) – отображает настраиваемые списки всех объектов базы данных, с ее помощью можно открывать объекты и выполнять определенные действия. Панель навигации прикреплена к левому краю окна Access. При необходимости ее можно свернуть, освобождая дополнительное место для объектов базы данных.

**Рабочая область** – в ней располагаются открытые объекты базы данных, с которыми может работать пользователь. Несколько открытых объектов располагаются на вкладках, которые «наложены» друг на друга. Для перехода между вкладками (объектами) используют ярлыки, воспроизводящие названия объектов.

**Строка состояния** – в ней отображается текущее состояние программы, а также содержатся кнопки для быстрого переключения режимов отображения активного объекта базы данных.

**Контекстная панель инструментов** – содержит кнопки для выполнения команд форматирования выделенного фрагмента текста.

**Режимы работы.** Приложение поддерживает три режима работы:

1. Режим запуска, в котором можно выполнять сжатие, преобразование, шифрование/дешифрование и другие операции без открытия базы данных.

2. Режим конструктора, в котором создаются и модифицируются объекты базы данных.

3. Режим выполнения, в котором отображаются окна объектов базы данных. При работе с таблицей этот режим называется – режимом таблицы, с формой – режимом формы, с отчетом – режимом предварительного просмотра.

**Создание базы данных.** При наличии проекта физической модели базы данных можно приступить к ее созданию в Microsoft Access. Сначала создается файл базы данных, затем – все входящие в нее объекты и, в первую очередь, таблицы. Все остальные объекты базы данных будут основаны на таблицах и установленных между ними связях.

СУБД Access позволяет создавать БД, используя различные средства: новый файл, новый файл на основе шаблона, конвертирование внешней базы данных (текстового формата, формата электронной таблицы, реляционной базы данных и т. п.).

#### 1. Создание новой базы данных.

1.1. Запускаем Microsoft Access, выполнив команду меню Пуск > Программы > Microsoft Office > Microsoft Office Access 2007.

1.2. В открывшемся окне (рис. 4.7) нажимаем на значке **Новая База данных**.

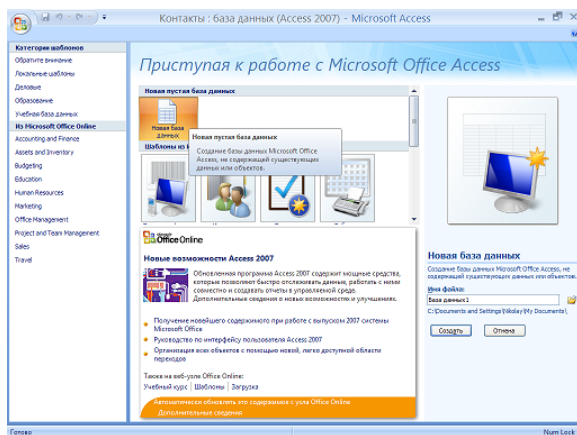


Рис. 4.7. Окно Microsoft Access

1.3. В поле **Имя файла** пишем название создаваемой базы, например «Библиотека».

1.4. По умолчанию база данных создается в папке **Мои документы** (My Documents), для изменения пути необходимо кликнуть на иконке



слева от поля **Имя файла**.

1.5. Нажимаем кнопку **Создать**.

2. *Создание базы данных с помощью шаблонов.*

2.1. Запускаем Microsoft Access, выполнив команду меню Пуск → Программы → Microsoft Office → Microsoft Office Access 2007.

2.2. В открывшемся окне (рис. 4.8), в области **Локальные шаблоны** выбираем нужный шаблон.

2.3. Нажимаем кнопку **Создать**.

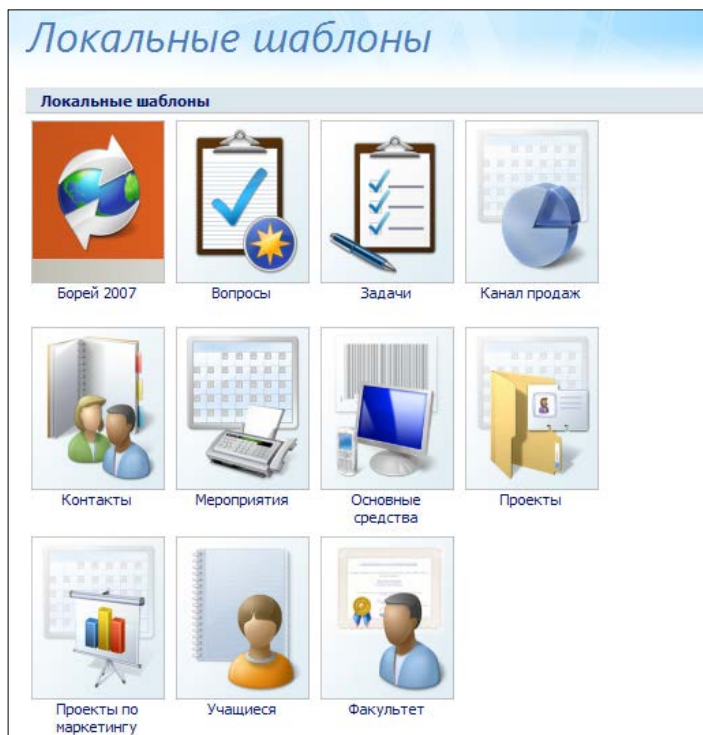


Рис. 4.8. Локальные шаблоны

**Таблицы.** После создания новой базы открывается окно **База данных** (рис. 4.9).

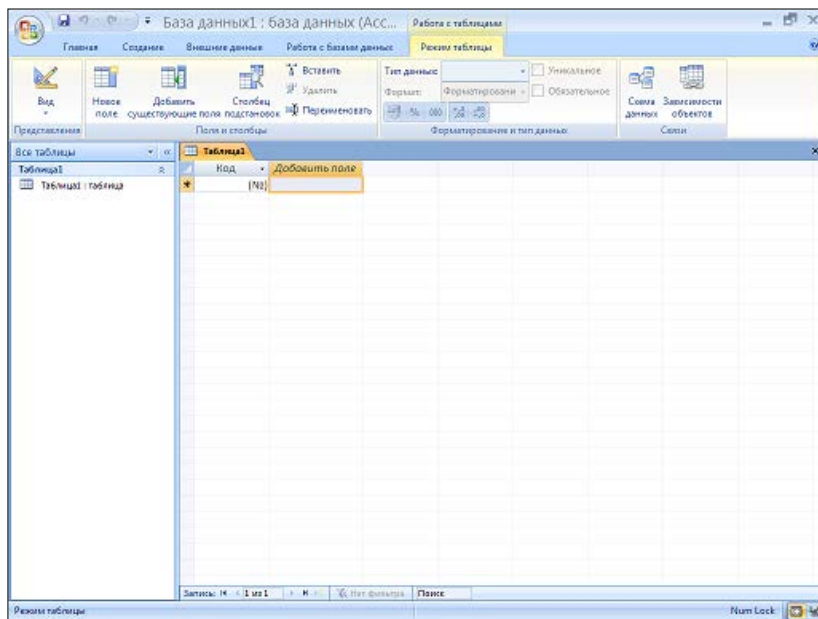


Рис. 4.9. Окно **Базы данных**

### 3. Создание таблицы с помощью шаблонов таблиц.

#### 3.1. Переходим на закладку **Создание** (рис. 4.10).

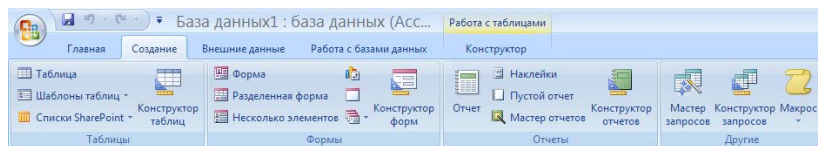


Рис. 4.10. Верхнее меню

#### 3.2. Нажимаем **Шаблоны таблиц**, выбираем **Контакты**.

3.3. В результате получили таблицу с готовыми именами столбцов, при необходимости можно добавить свои столбцы (двойной клик ЛКМ на последнем столбце **Добавить поле** и ввести имя нового столбца)



или удалить лишние столбцы (клик ПКМ на лишнем столбце, пункт контекстного меню – **Удалить столбец**) (рис. 4.11).

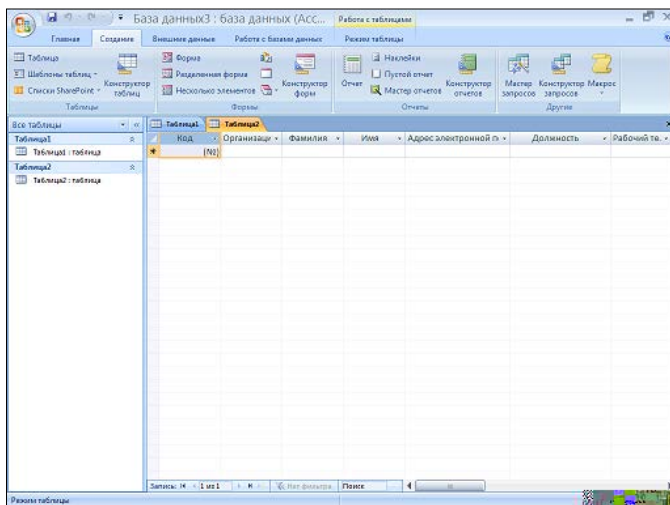


Рис. 4.11. Окно таблицы

3.4. Кликом в ячейку под названием **Организация**, введем название организации – **Школа**, тогда значению **Код** автоматически присваивается значение.

3.5. Заполним остальные ячейки таблицы (рис. 4.12).

3.6. Сохраняем таблицу: ПКМ на закладке **Таблица 2** вводим имя **Контакты**, нажимаем **ОК**.

| Код | Организация | Фамилия  | Имя  | Адрес электронной почты | Должность     | Рабочий телефон |
|-----|-------------|----------|------|-------------------------|---------------|-----------------|
| 1   | Школа       | Иванов   | Вася | 64231@mail.ru           | учитель       | 326598          |
| 2   | ТГУ         | Кузнецов | Петя | 45219@mail.ru           | студент       | 125463          |
| 3   | ТГУ         | Петров   | Ваня | 32681@mail.ru           | преподаватель | 635512          |
| *   | (Nz)        |          |      |                         |               |                 |

Рис. 4.12. Таблица данных

4. *Создание таблицы с помощью конструктора таблиц.*

4.1. Кликнем на закладку **Таблица 1**.

4.2. Переходим в режим конструктора (рис. 4.13).

4.3. Предлагается сохранить таблицу, вводим имя **Книги** и нажимаем **ОК** (рис. 4.14).

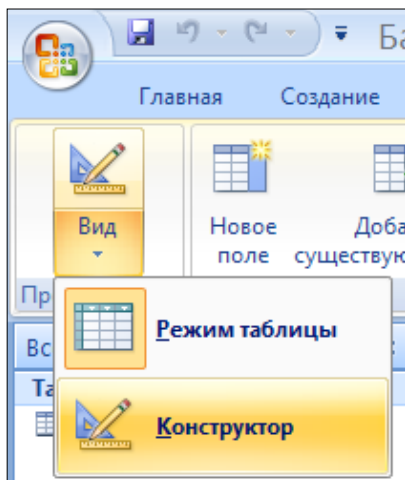


Рис. 4.13. Выбор режима отображения

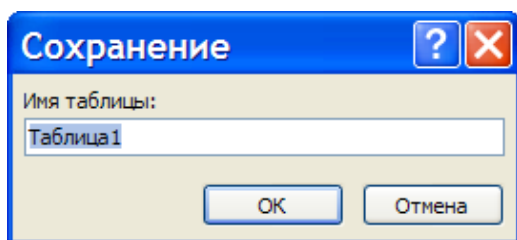


Рис. 4.14. Ввод имени таблицы

4.4. В открывшемся окне конструктора таблиц можно вводить имена полей новой таблицы или добавлять поля в уже созданную таблицу (рис. 4.15).

4.5. Кликнем ниже ячейки **Код**, введем имя столбца **Название**, тип данных – текстовый. Заполняем еще несколько ячеек (рис. 4.16).

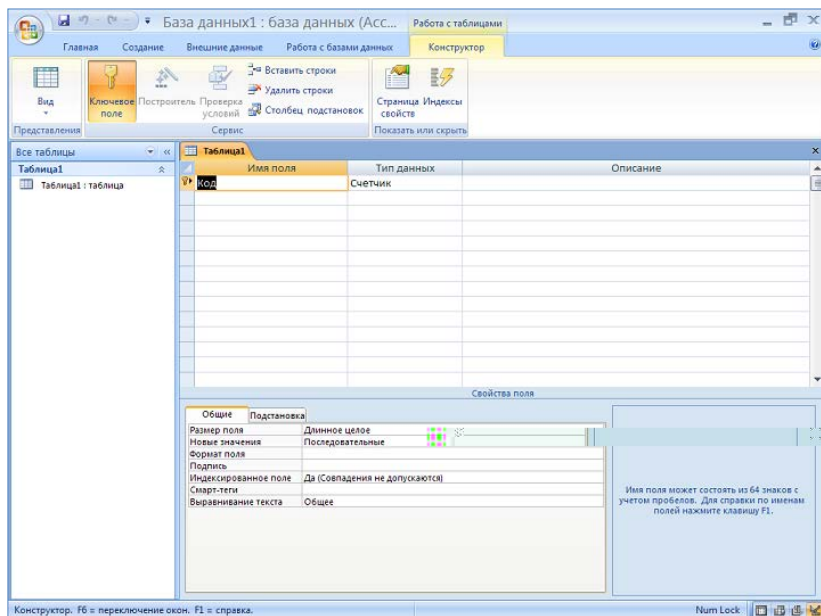


Рис. 4.15. Окно конструктора таблиц

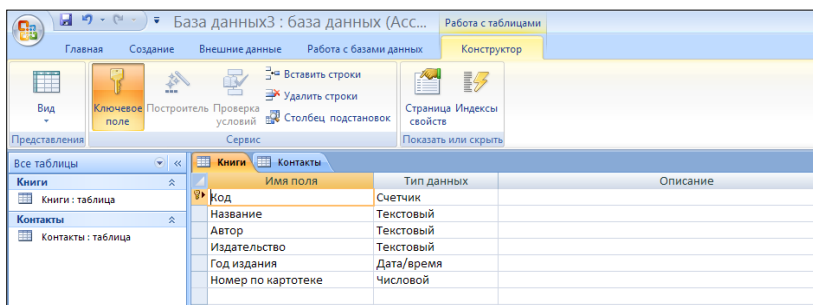


Рис. 4.16. Заполнение полей таблицы

4.6. Переходим в режим таблицы: **Вид → Режим таблицы.**

4.7. Заполняем таблицу аналогично п. 4.4. и 4.5 (рис. 4.17).

База данных: база данных (Асс...)

Режим таблицы

Вид: Представление, Вставить, Буфер обмена, Шрифт, Текст RTF

Создать, Сохранить, Обновить все, Удалить, Дополнительно, Записки, Фильтр, Найти

Все таблицы: Книги, Контакты

| Код | Название     | Автор         | Издательство | Год издания | Номер по к. | Добавить поле |
|-----|--------------|---------------|--------------|-------------|-------------|---------------|
| 1   | Война и Мир  | Толстой Л.Н.  | Москиздат    | 1981        | 643139      |               |
| 2   | Преступление | Достоевский < | Литиздат     | 1990        | 487266      |               |
| 3   | Ревизор      | Гоголь Н.В.   | Москиздат    | 1997        | 354471      |               |
| 4   | Мастер и Мар | Булгаков М.А. | Москиздат    | 1984        | 346864      |               |
| *   | (No)         |               |              |             |             |               |

Записи: 4 из 4, Нет фильтров, Поиск

Рис. 4.17. Таблица данных

**Связь таблиц.** Связь позволяет установить правила взаимодействия между таблицами. Различают два типа связей: один ко многим и многие ко многим. В нашем случае при создании базы данных библиотеки подходит связь – многие ко многим, т. е. у одного человека может быть несколько книг и экземпляры одной книги могут быть у разных людей.

При создании связи **один ко многим** ключ первой таблицы прописывается отдельным столбцом во второй таблице.

При создании связи **многие ко многим** ключи обеих таблиц прописывается в третьей таблице (вспомогательной).

Рассмотрим на примере.

#### 5. Создание связи.

5.1. Переходим на закладку **Создание** и создаем новую таблицу.

5.2. Затем переходим в режим конструктора, при этом сохраняем таблицу под именем **Записи**.

5.3. Вписываем названия столбцов по примеру (рис. 4.18).

5.4. Переходим в режим таблицы и заполняем ее по примеру (рис. 4.19).

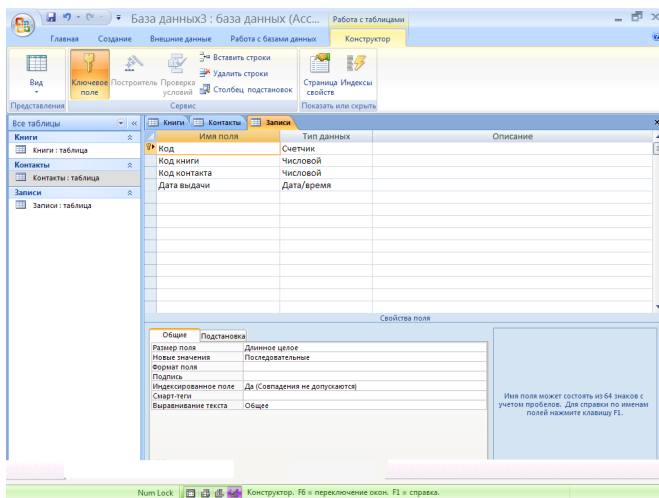


Рис. 4.18. Заполнение полей таблицы

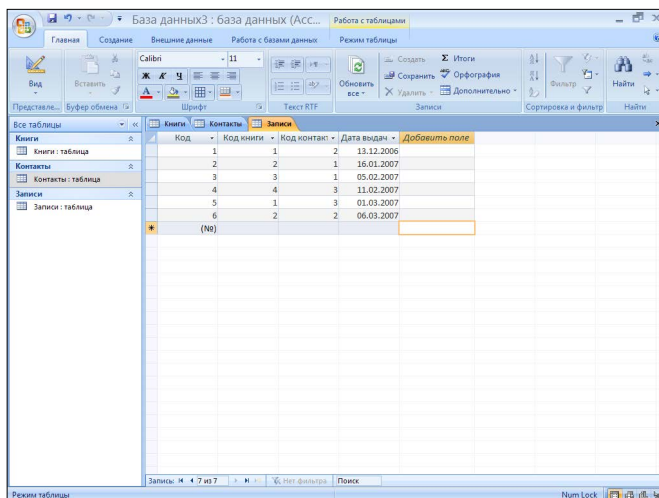


Рис. 4.19. Таблица данных

5.5. Переходим на закладку **Работа с базами данных** и нажимаем на кнопку **Схема данных**.

5.6. В открывшемся окне последовательно добавляем все три таблицы, в результате получим в окне **Схема данных** 3 таблицы (рис. 4.20).

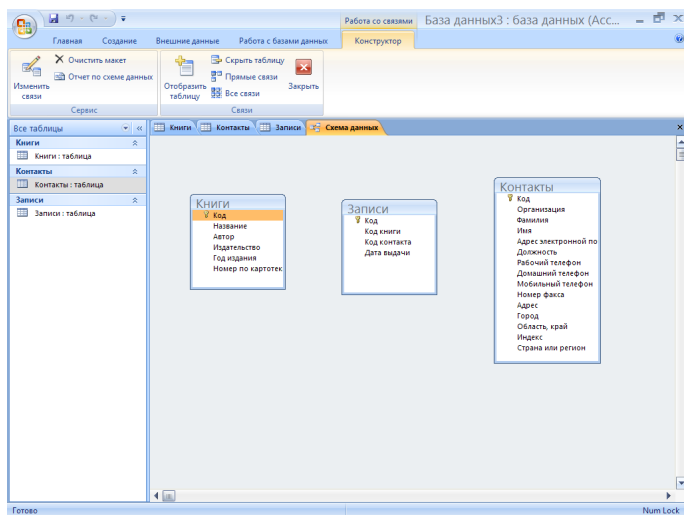


Рис. 4.20. Окно схемы данных

5.7. Создаем связь – помещаем указатель на пункт **Код** из таблицы **Книги**, нажимаем **ЛКМ** и, не отпуская ее, перетаскиваем в пункт **Код книги** из таблицы **Записи**.

5.8. В появившемся окне ставим флажок *Обеспечение целостности данных*. Этот режим не позволит Access оставлять в таблице **Записи** записи, для которых нельзя подобрать запись таблицы **Книги** с подходящим значением поля **Код**.

Установка флажка *Каскадное удаление связанных записей* приведет к тому, что при удалении записи таблицы **Книги** будут удалены все соответствующие записи таблицы **Записи**. Если указанный флажок сброшен, удаление тех записей таблицы **Книги**, на которые ссылается хотя бы одна запись таблицы **Записи**, запрещено.

Установка флажка *Каскадное обновление связанных полей* приведет к тому, что при обновлении поля **Код** таблицы **Книги** будут автоматически обновляться одноименные поля в соответствующих записях таблицы **Записи** (рис. 4.21).

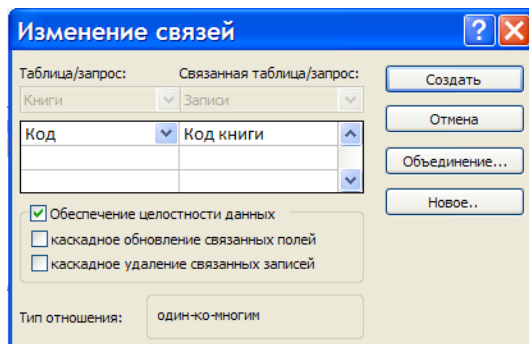


Рис. 4.21. Параметры связи

5.9. Нажимаем кнопку **Создать**.

5.10. Повторяем пункты 5.7, 5.8, 5.9 для таблицы **Контакты** (рис. 4.22).

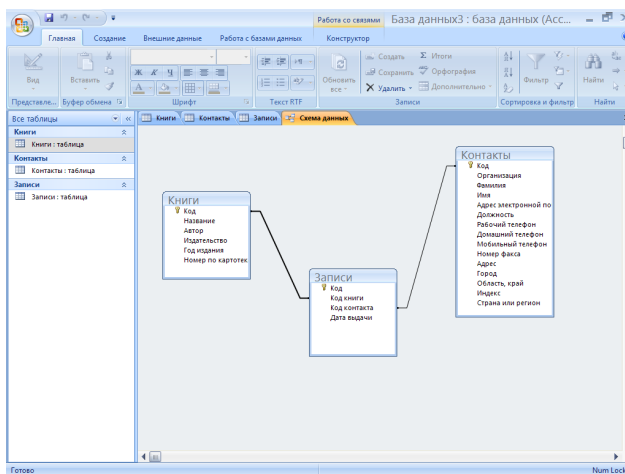


Рис. 4.22. Схема данных

## 6. Список подстановки.

При заполнении таблицы **Записи** новыми данными использование для этого **Кода** очень неудобно, для облегчения заполнения таблицы в Access есть список подстановки.

- 6.1. Откроем таблицу **Записи** в режиме конструктора.
- 6.2. Для имени поля **Код книги** в поле **Тип данных** выбираем пункт – **Мастер подстановок**.
- 6.3. В открывшемся окне **Создание подстановки** оставляем переключатель, как показано на рис. 4.23 и нажимаем **Далее >**.
- 6.4. Выбираем таблицу **Книги** (рис. 4.24) и нажимаем **Далее >**.

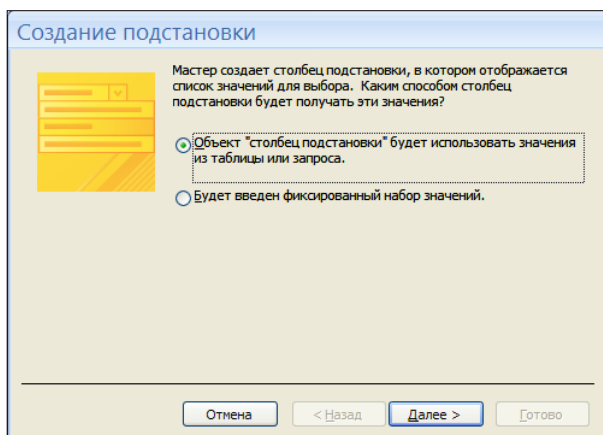


Рис. 4.23. Мастер подстановок (шаг 1)

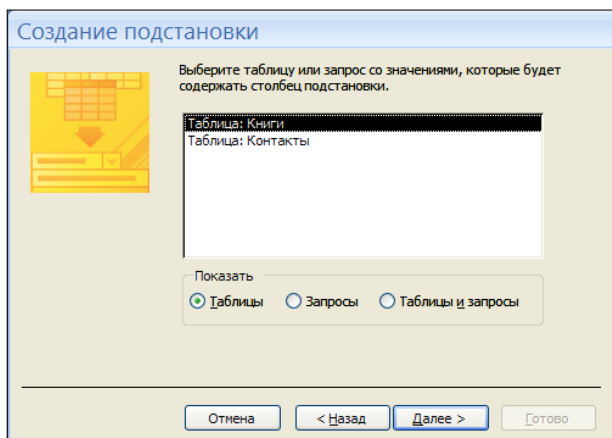


Рис. 4.24. Мастер подстановок (шаг 2)



6.5. Добавляем поля **Название** и **Автор** с помощью кнопки «>» и нажимаем **Далее >** (рис. 4.25).

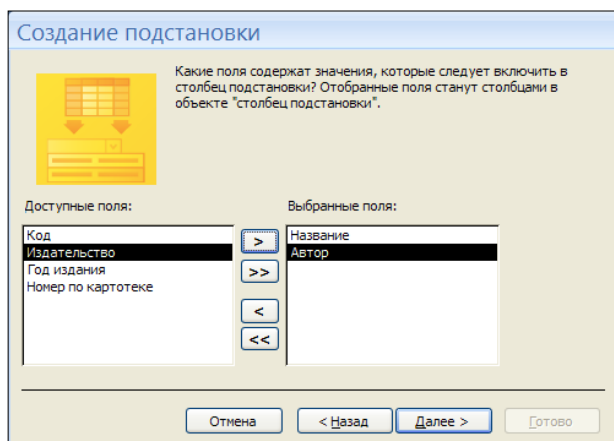


Рис. 4.25. Мастер подстановок (шаг 3)

6.6. Можно выполнить сортировку записей по возрастанию или по убыванию (рис. 4.26), выбрав название поля из списка, нажимаем **Далее >**.

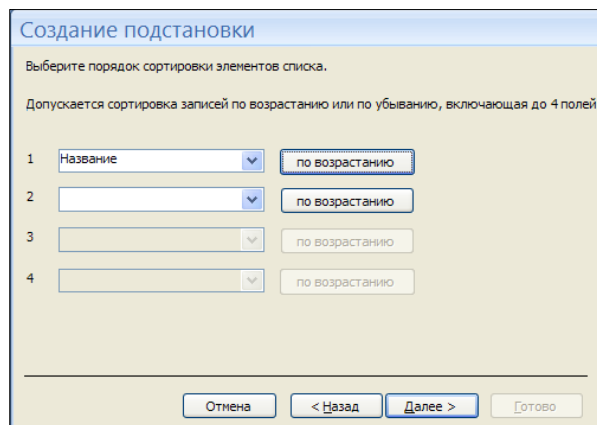


Рис. 4.26. Мастер подстановок (шаг 4)



| Записи |                    |                    |                |               |  |
|--------|--------------------|--------------------|----------------|---------------|--|
| Код    | Код книги          | Код контакта       | Дата выдачи    | Добавить поле |  |
| 7      | Война и Мир        | Кузнецов           | 13.12.2006     |               |  |
| 8      | Преступление       | Иванов             | 16.01.2007     |               |  |
| 9      | Ревизор            | Иванов             | 05.02.2007     |               |  |
| 10     | Мастер и Маргарита | Петров             | 11.02.2007     |               |  |
| 11     | Война и Мир        | Петров             | 01.03.2007     |               |  |
| 12     | Преступление       | Кузнецов           | 06.03.2007     |               |  |
| *      | (№)                |                    |                |               |  |
|        |                    | Война и Мир        | Толстой Л.Н.   |               |  |
|        |                    | Мастер и Маргарита | Булгаков М.А.  |               |  |
|        |                    | Преступление       | Достоевский Ф. |               |  |
|        |                    | Ревизор            | Гоголь Н.В.    |               |  |

Рис. 4.29. Заполнение таблицы с помощью **Мастера**

**Запросы.** Запросы являются инструментом поиска и структурирования данных. Запрос, адресованный одной или нескольким таблицам, инициирует выборку определенной части данных и их передачу в таблицу, формируемую самим запросом. В результате вы получаете подмножество информационного множества исходных таблиц, сформированное по определенному закону. Если обрабатываемый объем информации велик, выделение необходимых данных в такое подмножество позволяет существенно сократить время их обработки. В системах типа клиент-сервер, где основные базы данных хранятся на файловом сервере, система запросов позволяет уменьшить объем информации, передаваемой через локальную сеть.

#### 7. Мастер запросов.

Чтобы упростить задачу пользователя, в состав Access включен **Мастер запросов**, позволяющий автоматизировать процесс построения запроса.

7.1. Переходим на закладку **Создание** и нажимаем кнопку **Мастер запросов**.

7.2. Выбираем **Простой запрос** и нажимаем **ОК**.

7.3. В раскрывающемся списке **Таблицы и запросы** (рис. 4.30) выбираем таблицу **Контакты**, из списка **Доступные поля** выбираем: **Фамилия**, **Имя** и **Рабочий телефон**. Затем из таблицы **Книги** выбираем: **Название** и **Автор**, а из таблицы **Записи** – **Дата выдачи**. И нажимаем **Далее** >.

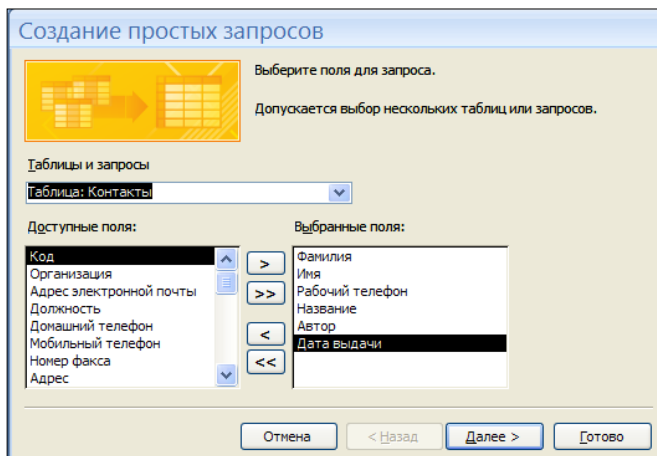


Рис. 4.30. Мастер запросов (шаг 1)

7.4. Выбираем **подробный отчет** (рис. 4.31) и нажимаем **Далее >**.

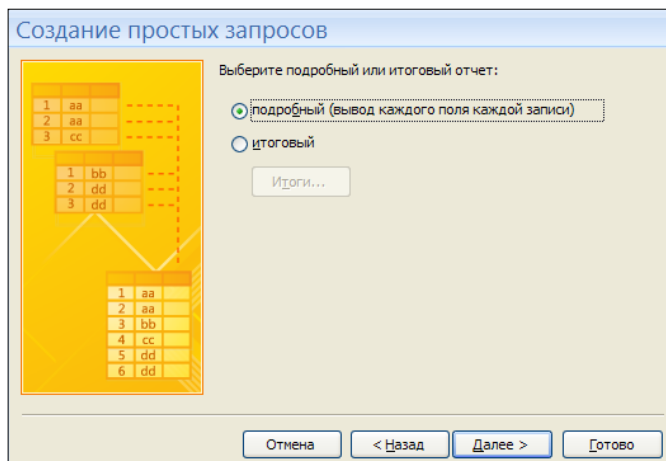


Рис. 4.31. Мастер запросов (шаг 2)

7.5. Вводим имя запроса, ставим переключатель на **Открыть запрос для просмотра данных** (рис. 4.32) и нажимаем **Готово**.

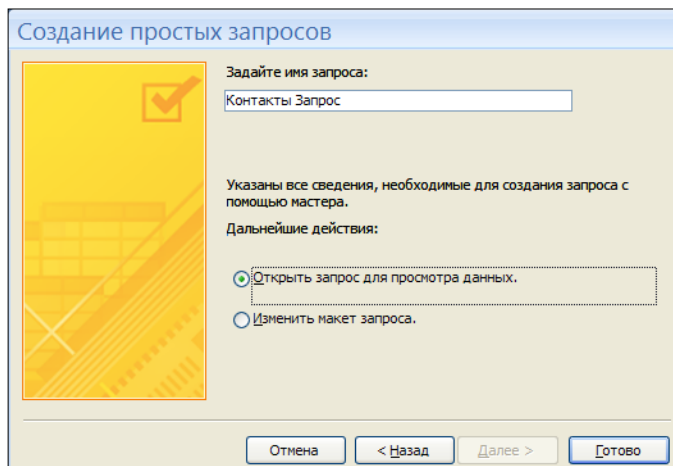


Рис. 4.32. Мастер запросов (шаг 3)

7.6. В результате получаем таблицу (рис. 4.33).

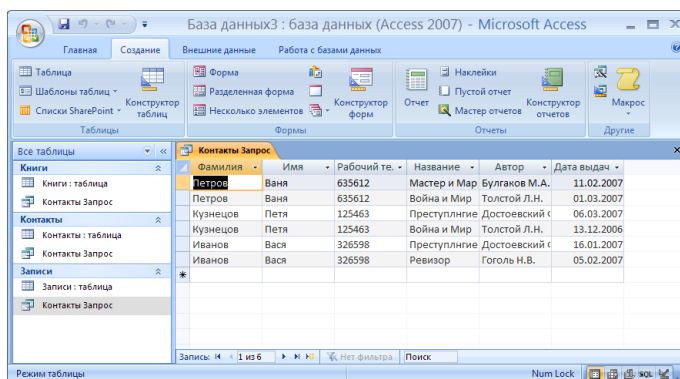


Рис. 4.33. Таблица запроса

## 8. Конструктор запросов.

Мастер запросов умеет конструировать только простые условия отбора. Чтобы наложить дополнительные ограничения, следует пользоваться конструктором запросов, обеспечивающим полное управление параметрами запроса и построение сложных условий отбора данных.

8.1. Переходим на закладку **Создание** и нажимаем кнопку **Конструктор запросов**.

8.2. Появляется диалоговое окно **Добавление таблицы** (рис. 4.34). Добавляем все три таблицы и закрываем окно.

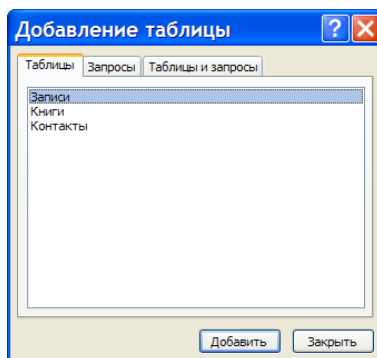


Рис. 4.34. Добавление таблицы

8.3. Из раскрывающегося списка выбираем таблицы и поля для отображения в запросе (рис. 4.35).

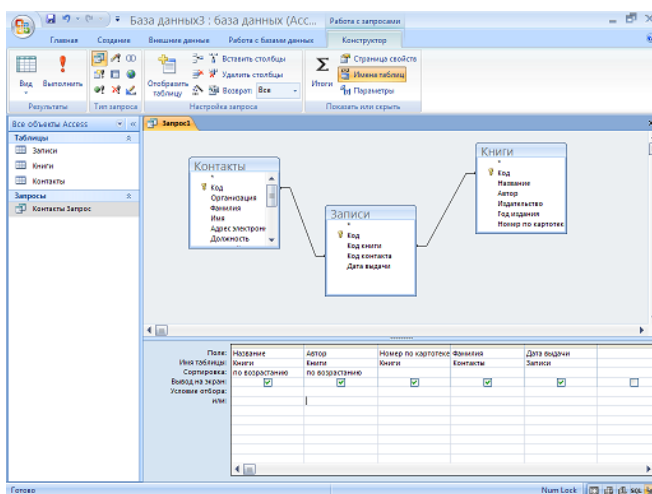


Рис. 4.35. Окно **Конструктор запросов**

## 9. Условие отбора.

Добавление в запрос условия отбора позволяет выбирать из таблицы не все записи, а лишь те, которые удовлетворяют определенным критериям. Например, нам нужны записи, приходящиеся на февраль 2007 г.

9.1. В бланке запроса щелкаем на ячейке **Условие отбора пятого столбца** правой кнопкой мыши и выбираем в контекстном меню команду **Построить**. Откроется окно построителя выражений (рис. 4.36).

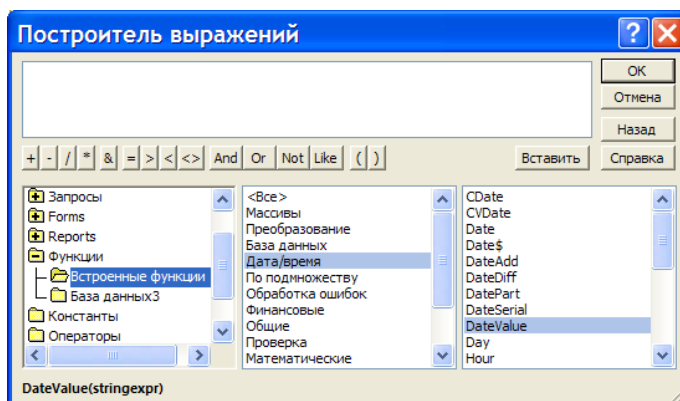


Рис. 4.36. Окно построителя выражений

9.2. В левом списке построителя щелкаем на папке **Операторы**.

9.3. В среднем списке выбираем категорию **Сравнения**.

9.4. В правом списке дважды щелкаем на пункте **Between**.

9.5. Щелчком выделяем в поле формулы первый местозаполнитель **Выражение**.

9.6. В левом списке построителя выражений двойным щелчком открываем папку **Функции**.

9.7. Щелкаем на папке **Встроенные функции**, содержащей стандартные функции Access.

9.8. В среднем списке построителя выражений щелкаем на пункте **Дата/время**.

9.9. В правом списке дважды щелкаем на функции **DateValue**, чтобы заменить ею местозаполнитель **Выражение**.

9.10. Выделив местозаполнитель **stringexpr**, вводим текст **01.02.2007**.

9.11. Повторяя шаги, заменяем второй местозаполнитель **Выражение** на выражение **DateValue (31.02.2007)**.

В результате у нас получилась формула **Between DateValue (01.02.2007) and DateValue (31.02.2007)**. Она проверяет условие нахождения даты в интервале от 1 до 31 февраля 2007 г.

9.12. Нажимаем **ОК**.



9.13. Для выполнения запроса нажимаем . Результат выполнения запроса представлен на рис. 4.37.

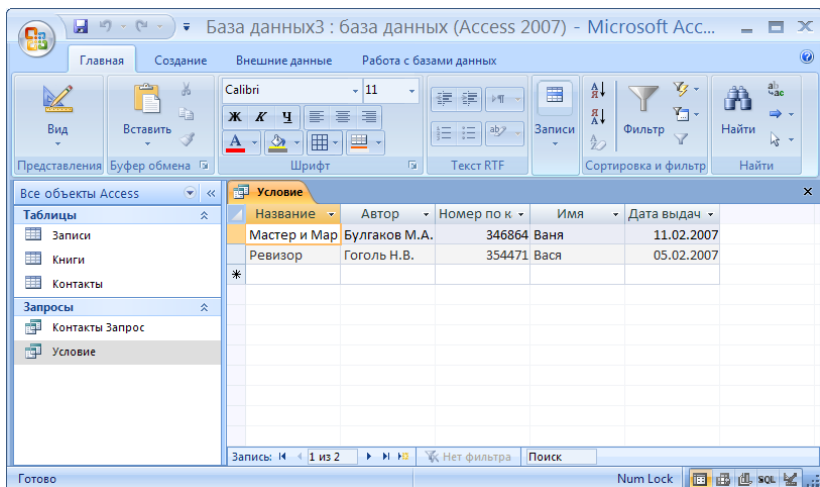


Рис. 4.37. Результат выполнения запроса

## 10. Запрос действия.

Запросы могут применяться также для добавления, удаления и обновления группы записей таблицы. Такие запросы являются мощным инструментом преобразования данных, они называются запросами действия. К примеру, нам нужно изменить в таблице **Книги** номер по картотеке. Заменить первую цифру – 4 на 6. Подобную операцию трудно провести вручную, если в таблице больше тысячи записей.

10.1. Создаем новый запрос в режиме конструктора.

10.2. В окне конструктора открываем таблицу **Книги**.

10.3. В значение поле выбираем **Номер по картотеке** (рис. 4.38).



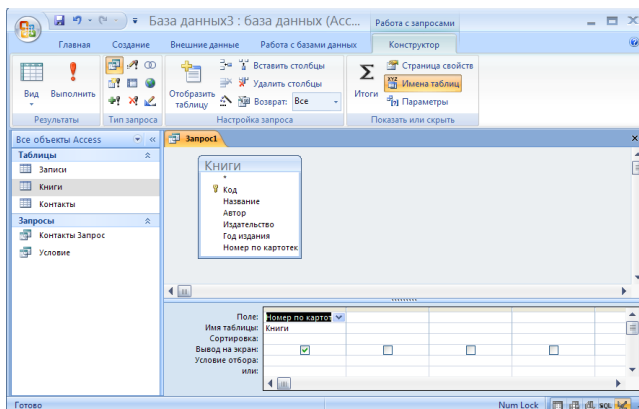


Рис. 4.38. Окно конструктора запроса

10.4. В поле **Тип запроса** выбираем запрос на обновление, при этом в бланке запроса появится еще одно поле **Обновление**, в которое нужно ввести новое значение поля.

10.5. В поле **Обновление** вызываем контекстное меню, щелкаем на пункте **Построить**.

10.6. В окне построителя выражений пишем формулу (рис. 4.39).

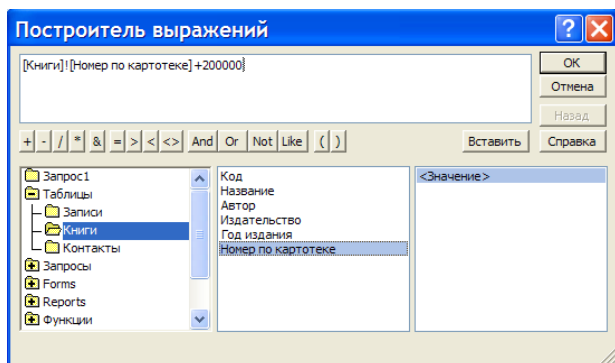


Рис. 4.39. Окно построителя выражений

10.7. В поле **Условие отбора** вызываем контекстное меню, щелкаем на пункте **Построить**.

10.8. В окне построителя выражений пишем формулу (рис. 4.40).

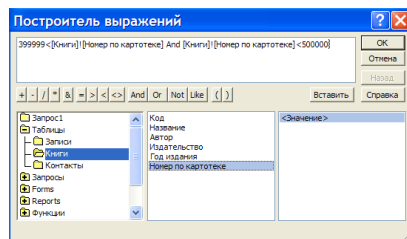


Рис. 4.40. Окно построителя выражений

10.9. Закрываем запрос, сохранив под именем **Обновление**.

10.10. Двойным кликом выполняем запрос **Обновление**, Access проинформирует об обнаружении одной записи, удовлетворяющей условию отбора, и попросит подтвердить необходимость ее изменения.

Виды запросов действия представлены в табл. 4.2.

Таблица 4.2. Виды запросов действия

| Тип              |  | Описание                                                                                                                              |
|------------------|--|---------------------------------------------------------------------------------------------------------------------------------------|
| Выбор            |  | Выполняет отбор записей из базы данных и показывает их                                                                                |
| Обновление       |  | Обновляет данные в существующей таблице                                                                                               |
| Объединение      |  | Запрос в режиме SQL                                                                                                                   |
| Создание таблицы |  | Выполняет отбор записей из базы данных и сохраняет их как новую таблицу                                                               |
| Перекрестный     |  | Выполняет сведение данных по двум наборам значений, один из которых отображается в левой части таблицы, а другой – в верхней ее части |
| Запрос к серверу |  | SQL запрос на сервер                                                                                                                  |
| Добавление       |  | Добавляет данные в существующую таблицу                                                                                               |
| Удаление         |  | Удаляет данные, соответствующие указанному условию из указанной таблицы                                                               |
| Управление       |  |                                                                                                                                       |

**Формы.** В то время как таблицы и запросы позволяют отобразить на экране длинные списки записей, формы дают возможность сосредоточиться на конкретной записи. Они облегчают ввод, редактирование и восприятие информации, могут содержать вспомогательные подписи и элементы оформления.

#### 11. *Мастер форм.*

11.1. Переходим на закладку **Создание** и запускаем **Мастер форм** (рис. 4.41).

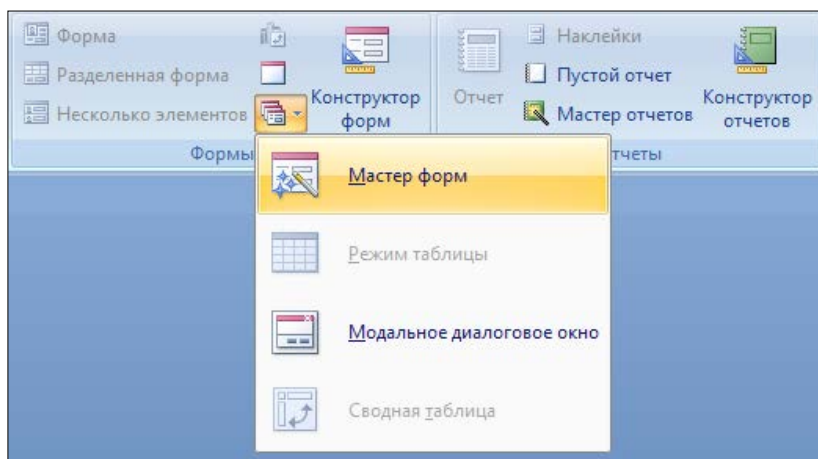


Рис. 4.41. Запуск **Мастера форм**

11.2. В списке **Таблицы и запросы** выберите **Таблица: Контакты**.

11.3. Кликом на кнопке «>>» добавьте все поля в список выбранных полей.

11.4. Выделите поле **Код** и кликом на кнопке «<» уберите из списка выбранных полей, так как значение этого поля является кодом, оно неважно для пользователя и поэтому его не следует включать в форму (рис. 4.42).

11.5. Кликните на кнопке **Далее >**.

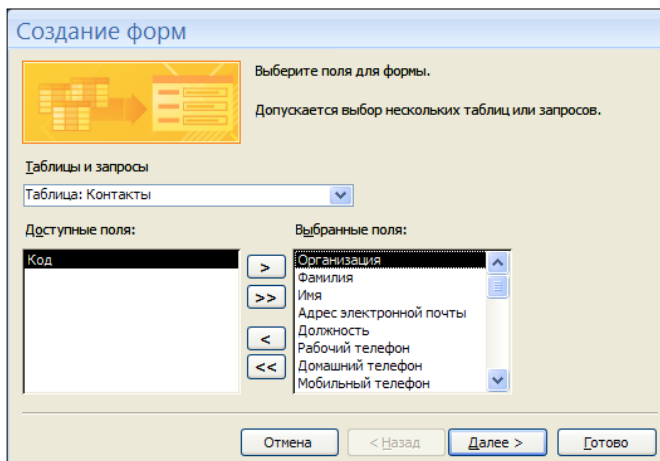


Рис. 4.42. Мастер форм

11.6. В следующем окне выберите внешний вид формы – выровненный – и кликните на кнопке **Далее >**.

11.7. В третьем окне мастера выберите стиль оформления формы (какая понравится) и нажмите на кнопку **Далее >**.

11.8. В четвертом окне нажмите на кнопку **Готово**, ничего не изменяя. В результате получим форму, представленную на рис. 4.43.

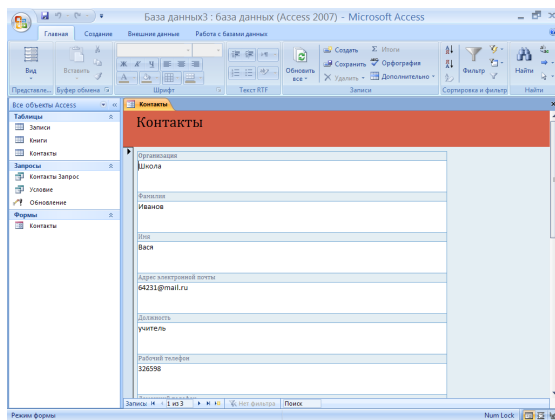
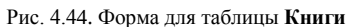


Рис. 4.43. Форма для таблицы **Контакты**

**Конструктор форм.** Недостатком форм, создаваемых мастером, является то, что они однообразны и не содержат пояснительных надписей. Чтобы приукрасить форму и расположить поля более удобным образом, следует воспользоваться конструктором форм, который позволяет передвигать и масштабировать элементы формы, связывать их с источником данных и настраивать любые другие параметры формы.

12.1. Создадим, как показано в подразделе 4.1, форму на таблицу **ИГГИ** (рис. 4.44).



197

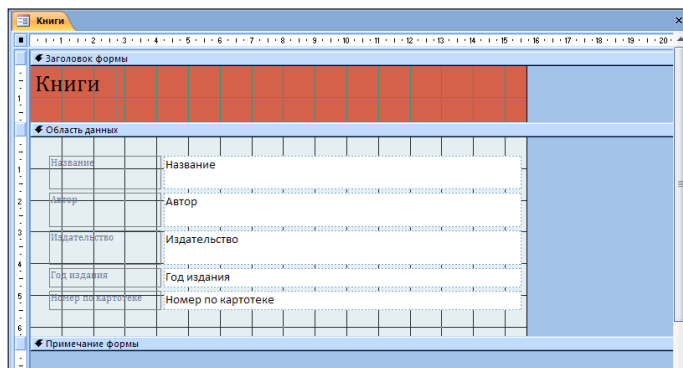


Рис. 4.45. Окно конструктора форм

В окне конструктора присутствует разметочная сетка с вертикальной и горизонтальной линейкой, помогающая позиционировать объекты.

При нажатии ПКМ в поле **Заголовок формы** или **Область данных**, пункт **Цвет заливки/фона** можно изменить цвет поля.

Кликнув на любом элементе формы, вокруг него появляется рамка, потянув за края которой можно изменить форму и размер элемента или переместить его в другое место формы.

Каждый элемент формы имеет свои свойства, окно свойств открывается нажатием ПКМ на элементе, пункт **Свойства** (рис. 4.46).

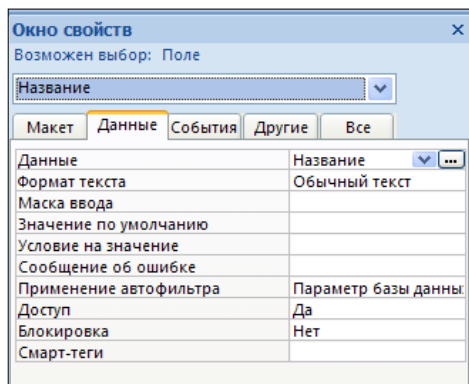


Рис. 4.46. Окно свойств

При открытии конструктора форм на экране появляется панель элементов. С помощью кнопок панели элементов в форму можно добавлять различные объекты, типы которых перечислены в табл. 4.3.

Таблица 4.3. Элементы управления форм

| Название элемента            |                                                                                     | Назначение                                                                                                                                                                                               |
|------------------------------|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1                            | 2                                                                                   | 3                                                                                                                                                                                                        |
| Поле                         |    | Используется для ввода и отображения информации полей таблиц и запросов, а также для вывода результатов вычислений                                                                                       |
| Подпись                      |    | Подписи создаются автоматически вместе с элементами типа текстовое поле, поле со списком и т. п. Они разъясняют смысл поля ввода. Дополнительные надписи могут использоваться для заголовков и пояснений |
| Кнопка                       |    | Щелчок на такой кнопке выполняет команду, с помощью которой можно перейти в другую форму, переместиться по записям и выполнить многие другие операции                                                    |
| Поле со списком              |    | Разрешает как непосредственный ввод значения в поле, так и его выбор в раскрывающемся списке предопределенных значений                                                                                   |
| Список                       |    | Позволяет выбирать данные из набора пунктов и не разрешает непосредственный ввод значений                                                                                                                |
| Подчиненная форма/отчет      |    | Вставляет в форму (или отчет) область с другой формой (или отчетом), связанной с главной                                                                                                                 |
| Линия                        |  | Добавляет прямую линию                                                                                                                                                                                   |
| Прямоугольник                |  | Рисует прозрачный или непрозрачный прямоугольник с цветной границей любой толщины                                                                                                                        |
| Присоединенная рамка объекта |  | Объект OLE, связанный с полем данных таблицы или запроса                                                                                                                                                 |
| Группа переключателей        |  | Группа переключателей ссылается на некоторое поле таблицы. Каждый переключатель группы соответствует определенному целочисленному значению этого поля                                                    |

| 1                       | 2                                                                                   | 3                                                                                                                                                                              |
|-------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Флажок                  |    | Предназначен для представления полей типа да/нет. Отмеченный квадратик соответствует величине «да», а пустой – «нет»                                                           |
| Переключатель           |    | Элемент группы переключателей, в которой может быть отмечен только один из них (его кружок выглядит зачерненным)                                                               |
| Выключатель             |    | Позволяет вводить информацию типа да/нет. Величине «да» или «истина» соответствует утопленное положение выключателя                                                            |
| Вкладка                 |    | Позволяет разместить на одном и том же пространстве экрана несколько наборов элементов управления. Удобен в тех случаях, когда элементы легко разделяются на логические группы |
| Вставить диаграмму      |    |                                                                                                                                                                                |
| Свободная рамка объекта |    | Объект OLE, не связанный ни с каким источником данных                                                                                                                          |
| Рисунок                 |    | Рисунок любого графического формата с рамкой                                                                                                                                   |
| Разрыв страниц          |    | Линия, по которой формируется перевод страницы при выводе формы или отчета на принтер                                                                                          |
| Вставить гиперссылку    |   | Создание ссылки на web-страницу, рисунок, адрес электронной почты или программу                                                                                                |
| Вложение                |  | Присоединенные файлы                                                                                                                                                           |

**Отчеты.** В целом отчеты похожи на формы, но они, как правило, предназначаются для вывода информации из базы данных на принтер. Поэтому в отчетах данные форматируют так, чтобы их было удобно размещать на отдельных страницах. Отчеты поддерживают самые разнообразные способы оформления и позволяют группировать данные, разбивая их на логически цельные блоки.



### 13. *Мастер отчетов.*

Чтобы облегчить работу пользователя, в Access имеется специальный мастер, который при недостатке времени позволяет быстро создавать довольно привлекательные отчеты.

13.1. Переходим на закладку **Создание** и запускаем **Мастер отчетов** (рис. 4.47).

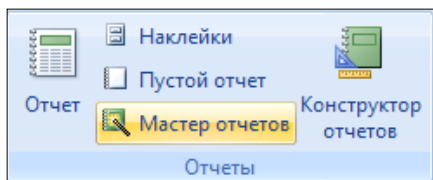


Рис. 4.47. Запуск **Мастера отчетов**

13.2. В открывшемся окне **Мастера**, выбираем **Таблица: Контакты**, добавляем все поля кроме поля **Код**, нажимаем **Далее >** (рис. 4.48).

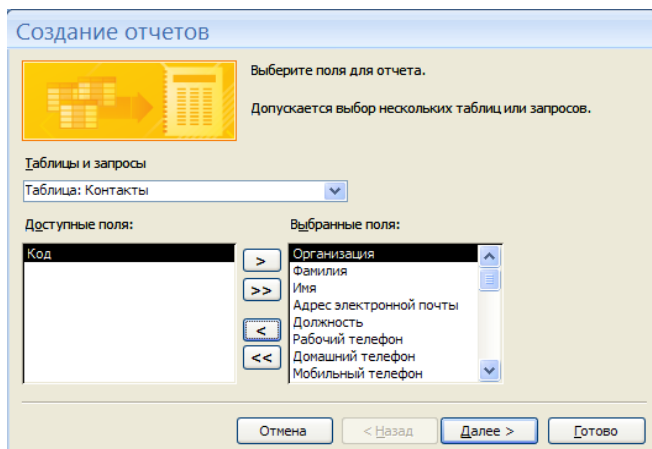


Рис. 4.48. Мастер отчетов (шаг 1)

13.3. В следующем окне **Мастера** оставляем все без изменения и нажимаем **Далее >**.

13.4. Выбираем сортировку по фамилии и нажимаем **Далее >** (рис. 4.49).

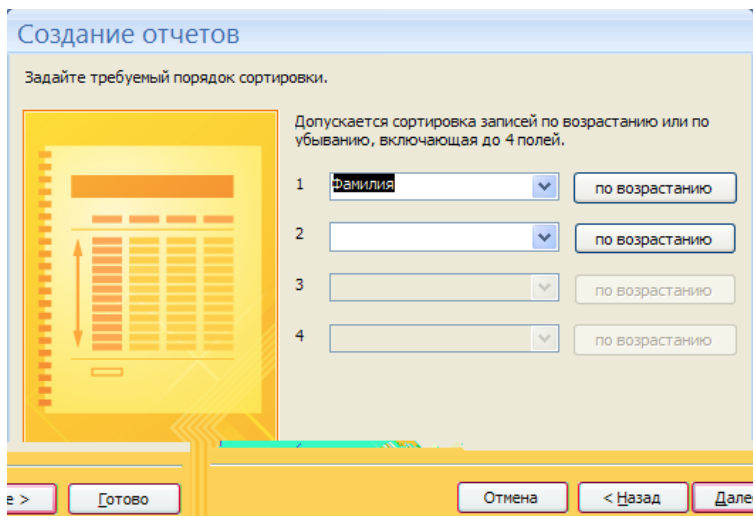


Рис. 4.49. Мастер отчетов (шаг 2)

13.5. Ставим переключатели, как показано на рис. 4.50, и нажимаем **Далее >**.

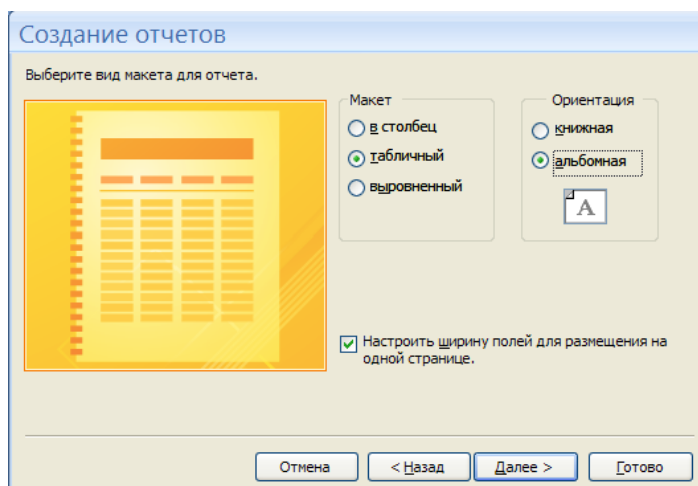


Рис. 4.50. Мастер отчетов (шаг 3)

13.6. Выбираем стиль оформления отчета – Access 2007 – и нажимаем **Готово** (рис. 4.51).

| Фамилия  | Организация | Имя  | Адрес электронной почты | Должность     | Рабочий телефон | Домашний телефон | Мобильный телефон |
|----------|-------------|------|-------------------------|---------------|-----------------|------------------|-------------------|
| Иванов   | Школа       | Вася | 64231@mail.ru           | учитель       | 326598          | 568 151          | 865 451 3165      |
| Кузнецов | ТТУ         | Петя | 45219@mail.ru           | студент       | 125 463         | 654 964          | 816513 1655       |
| Петров   | ТТУ         | Боря | 32 681@mail.ru          | преподаватель | 655 612         | 135 484          | 852 132 6543      |

1 апреля 2007 г.

Страница 1 из 1

Страница: 1 | 1 | Нет фильтров

Рис. 4.51. Отчет **Контакты**

#### 14. *Конструктор отчетов.*

Отчет, представленный на рис. 4.52, имеет ряд недостатков, таких как: одни поля занимают слишком много места, а другие поля не помещаются; отчет занимает больше одной страницы и др.

Эти недостатки можно исправить с помощью **Конструктора отчетов**.

14.1. Переходим в режим конструктора, щелкая ПКМ на отчете, пункт – Конструктор (рис. 4.52).

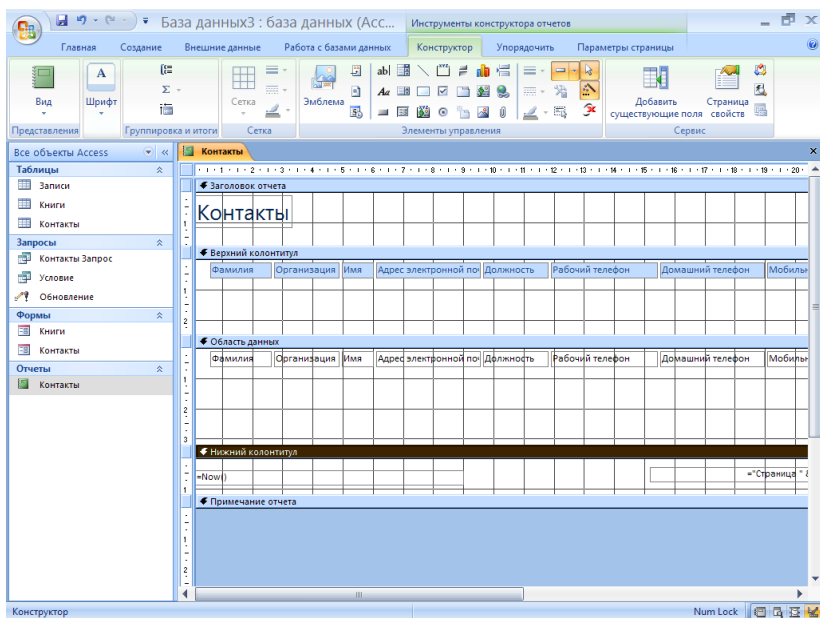


Рис. 4.52. Конструктор отчетов

14.2. Конструктор отчетов напоминает конструктор форм, в нем также можно изменять размеры, форму и положение элементов. Переместим элементы так, чтобы они не выходили за пределы одного листа и занимали место соответственно содержанию. После редактирования переходим в режим предварительного просмотра.

### Контрольные вопросы

1. Что представляет собой база данных?
2. В чем преимущества БД по сравнению с файловой организацией данных?
3. Назовите приложения и компоненты БД.
4. Кто является пользователем БД?
5. Дайте определение понятию «структура данных».
6. Назовите статические и динамические структуры данных.
7. Какие три аспекта включает понятие «модели данных»?

8. Каков принцип связи данных в иерархической и сетевой моделях данных?
9. Как определяют реляционную модель данных?
10. Какими свойствами должна обладать реляционная таблица?
11. Какие условия целостности должны выполняться в реляционной модели данных?
12. Дайте определение постреляционной модели данных.
13. Назовите основные понятия, на которых базируется объектно-ориентированная модель данных.
14. Как определяют объектно-реляционную модель данных?
15. Для чего предназначена многомерная модель данных?
16. Дайте определение понятию «система управления базами данных».
17. Каковы функциональные возможности современных СУБД?
18. Какие три компонента включает архитектура СУБД?
19. Дайте классификацию СУБД по принципу обработки запросов к БД.
20. Каковы направления развития СУБД?
21. Как выполняется запрос к БД в архитектуре файл-сервер?
22. Какие популярные настольные СУБД вы знаете?
23. Дайте определение понятию «сервер БД».
24. Каковы преимущества архитектуры клиент-сервер?
25. Охарактеризуйте современные серверы БД.
26. Дайте краткую характеристику СУБД Oracle, MS SQL Server, My SQL.
27. Дайте определение понятиям «распределенная БД» и «распределенная СУБД».

#### ЛИТЕРАТУРА

1. Компьютерные информационные технологии. Система управления базами данных: метод. указания по выполнению лабораторных работ в СУБД Access / М. С. Лапушкина [и др.]. – Горки: БГСХА, 2014. – 40 с.

## Тема 5. ИНФОРМАЦИОННЫЕ СИСТЕМЫ

- 5.1. Корпоративные информационные системы.
- 5.2. Системы искусственного интеллекта.
- 5.3. Экспертные системы: структура и классификация.

### 5.1. Корпоративные информационные системы

*Представление о корпоративных информационных системах.* До недавнего времени информационное обеспечение предприятий укладывалось в понятие учетно-отчетных информационных систем. Действительно, первым объектом автоматизации и информатизации стала область бухгалтерского учета и отчетности. Эволюция информационных технологий, внедрение web-технологий, экономическое развитие предприятий и расширение масштаба бизнеса, в том числе географического, привело к необходимости комплексной автоматизации деятельности и создания корпоративных информационных систем (КИС).

КИС отличаются от информационных систем (ИС) как развитием функциональности, так и расширением областей автоматизированной информационной поддержки. В то же время в ряде случаев нет четкого понимания, какие системы относятся к классу КИС, и порой система поддержки распределенного электронного документооборота компании, которая обеспечивает постановку документов на учет, их защиту и сохранность, или корпоративный портал, являющийся средством внутренней и внешней коммуникации, рассматриваются как КИС предприятия. Подавляющее число разработчиков финансово-экономического программного обеспечения также считает, что их продукты относятся к классу КИС.

Поэтому рассмотрим подробнее главные отличия КИС от ИС.

1. Как уже было сказано, основное отличие КИС от ИС состоит в расширении функциональности и класса задач, решаемых системой.

КИС – система информационного обеспечения управления предприятием, включающая, помимо учетно-отчетной компоненты, функциональность, обеспечивающую планирование и управление деятельностью предприятия.

2. Понятие корпоративной информационной системы (КИС) не обязательно связывать только с крупномасштабными компаниями. Систему автоматизации управления компании можно называть КИС в

том случае, если система охватывает все необходимые сферы управления компанией – от управления производственной деятельностью до управления финансами.

3. Автоматизация деятельности предприятия, как правило, начинается с внедрения учетных систем, в частности, систем бухгалтерского и кадрового учета, построения системы электронного документооборота, создания систем поддержки и ведения договорной деятельности. В этом случае мы говорим о наличии на предприятии нескольких информационных систем, которые могут работать автономно и являются составляющими «лоскутной» автоматизации предприятия. В основе построения КИС лежит концепция единого информационного пространства, все подсистемы КИС должны работать с единой базой данных, содержащей основные данные.

4. Лоскутная автоматизация компаний формируется, как правило, на базе собственных разработок с добавлением некоторого количества готового программного обеспечения, которое может поддерживаться разными операционными системами. КИС создается на базе единой интегрированной платформы. Создание КИС в рамках единой инструментальной среды, например SAP BS, существенно повышает эффективность работы системы.

5. Сегодня широко используется процессный подход к управлению деятельностью организации. Эффективность работы предприятия определяется степенью автоматизации основных и обеспечивающих бизнес-процессов компании. В основе работы ИС предприятия лежит функциональный подход, в то время как КИС – это интегрированный комплекс программ или информационных систем, поддерживающих основные и обеспечивающие бизнес-процессы компании.

6. Корпоративная информационная система – это не просто совокупность программ автоматизации бизнес-процессов компании: управление производством, ресурсами, финансово-хозяйственной деятельностью. Характерной чертой КИС является сквозная интеграция, при которой каждому модулю системы, отвечающему за свой бизнес-процесс, в режиме реального времени доступна вся необходимая информация, вырабатываемая другими модулями без дополнительного и, тем более, двойного ввода информации.

7. Информационные системы предприятия нацелены на решение частных задач, в то время как КИС – это инструмент повышения эффективности бизнеса компании.

Таким образом, можно дать следующее определение: корпоративная информационная система – это открытая интегрированная система реального времени, автоматизирующая бизнес-процессы компании всех уровней и направлений деятельности, в том числе бизнес-процессы принятия управленческих решений. При этом степень автоматизации бизнес-процессов определяется исходя из обеспечения максимальной прибыли компании.

**Структура корпоративной информационной системы. Виды обеспечения КИС.** Современные банки, предприятия и организации представляют собой совокупность подразделений, филиалов, отделов и офисов, обменивающихся между собой информацией и выполняющих отдельные части общей работы.

Любая организация – это совокупность взаимодействующих элементов (подразделений), каждый из которых может иметь свою структуру. Элементы связаны между собой функционально, т. е. они выполняют отдельные виды работ в рамках единого бизнес-процесса, а также информационно, обмениваясь документами, факсами, письменными и устными распоряжениями и т. д. Кроме того, эти элементы взаимодействуют с внешними системами, причем их взаимодействие может быть как информационным, так и функциональным. Такая ситуация справедлива практически для всех организаций, каким бы видом деятельности они не занимались (органы государственного управления, банки, промышленные предприятия, коммерческие фирмы и т. д.).

Такой общий взгляд на организацию позволяет сформулировать некоторые общие принципы построения корпоративных информационных систем, т. е. информационных систем в масштабе всей организации:

1) информационная модель, представляющая собой отражение реальной информационной базы предприятия и описывающая все существующие информационные потоки, совокупность правил и алгоритмов функционирования информационной системы;

2) техническое обеспечение (суперкомпьютеры, имеющие перспективные архитектуры и технологии организации вычислительного процесса);

3) средства коммуникации (сетевые компьютерные технологии, технологии Internet/Intranet, технологии клиент-сервер);

4) системное и сетевое программное обеспечение, обеспечивающее работу коммуникационных средств;



5) прикладное программное обеспечение, необходимое для выполнения прикладных задач в каждом подразделении банка;

6) средства обеспечения безопасности (разграничение доступа к ресурсам, обеспечение надежности функционирования корпоративной системы в целом).

**Требования, предъявляемые к КИС.** Исторически сформировался ряд следующих требований к корпоративным информационным системам:

- 1) системность;
- 2) комплексность;
- 3) модульность;
- 4) открытость;
- 5) адаптивность;
- 6) надежность;
- 7) безопасность;
- 8) масштабируемость;
- 9) мобильность;
- 10) простота в изучении;

11) поддержка на этапе внедрения и сопровождение со стороны разработчика.

Рассмотрим каждое из этих требований. КИС прежде всего должна отвечать требованиям *комплексности* и *системности*. Она охватывает все уровни управления, от корпорации в целом с учетом филиалов, дочерних фирм, сервисных центров и представительств до цеха, участка и конкретного рабочего места, а весь процесс производства с точки зрения информатики представляет собой непрерывную совокупность порождения, обработки, изменения, хранения и распространения информации. Все такие узлы связаны между собой потоками данных, овеществленными в виде документов, сообщений, приказов, действий и т. п.

Следующим требованием, предъявляемым к КИС, является *модульность* построения. Это требование также очень важно с точки зрения внедрения системы, поскольку позволяет распараллелить, облегчить и, соответственно, ускорить процесс инсталляции (установки приложения на рабочие места пользователей), подготовки персонала и запуска системы в промышленную эксплуатацию.

Поскольку ни одна типовая информационная система, внедряемая на реальном предприятии, не может быть исчерпывающе полной, а также в силу того, что на функционирующем предприятии могут быть

уже работающие и доказавшие свою полезность компоненты других КИС, то следующим определяющим требованием является *открытость*.

Любое современное предприятие существует не в изолированном пространстве, а в мире постоянно меняющегося спроса и предложения, заставляющем гибко реагировать на рыночную ситуацию, что может быть иногда связано с существенным изменением структуры предприятия и номенклатуры выпускаемых изделий или оказываемых услуг. Это означает, что КИС должна обладать свойством *адаптивности*, то есть гибко настраиваться на различные законодательства, иметь разноязычные интерфейсы, уметь работать с несколькими различными валютами одновременно. Желательно, чтобы кроме средств настройки система обладала и средствами развития (расширения функционала).

Одним из важнейших требований к такой системе является ее *надежность*, подразумевающая непрерывность функционирования системы в целом даже в условиях частичного выхода из строя отдельных ее элементов (модулей) вследствие непредвиденных и непреодолимых причин (ошибок в работе).

Чрезвычайно большое значение для любой крупномасштабной системы, содержащей большое количество информации, имеет *безопасность*. Требование безопасности включает в себя несколько аспектов.

#### *1. Защита данных от потери.*

Это требование реализуется в основном на организационном, аппаратном и системном уровнях. Прикладная система, какой является автоматизированная система управления (АСУ), необязательно должна содержать средства резервного копирования и восстановления данных. Эти задачи решаются на уровне операционной системы или СУБД.

#### *2. Сохранение целостности и непротиворечивости данных.*

Прикладная система должна отслеживать изменения во взаимозависимых документах и обеспечивать управление версиями и поколениями наборов данных.

#### *3. Предотвращение несанкционированного доступа к данным внутри системы.*

Эти задачи решаются комплексно как с помощью организационных мероприятий, так и на уровне операционных и прикладных систем. В частности, прикладные компоненты должны иметь развитые средства администрирования, позволяющие ограничивать доступ к данным и функциональным возможностям системы в зависимости от статуса

пользователя, а также вести мониторинг действий пользователей в системе.

*4. Предотвращение несанкционированного доступа к данным извне.*

Решение этой части проблемы ложится в основном на аппаратную и операционную среду функционирования КИС и требует ряда административно-организационных мероприятий.

Успешно функционирующее и получающее достаточную прибыль предприятие характеризуется тенденцией к росту, образованию дочерних фирм и филиалов, что в процессе эксплуатации КИС может потребовать увеличения количества автоматизированных рабочих мест и объема хранимой (обрабатываемой) информации. Такой подход к управлению информацией выдвигает требование масштабируемости.

На определенном этапе развития предприятия рост требований к производительности и ресурсам системы может определить переход на более новую программно-аппаратную платформу. Чтобы такой переход не повлек за собой кардинального изменения управленческого процесса и неоправданных финансовых затрат на приобретение более мощных прикладных компонентов, необходимо выполнение требования мобильности.

*Простота в изучении* – это требование включает в себя не только наличие интуитивно понятного графического интерфейса программ, но и наличие подробной, грамотной и хорошо структурированной документации, возможности обучения персонала на специализированных курсах и прохождения ответственными специалистами стажировки на предприятиях родственного профиля, где данная система успешно внедрена и эксплуатируется.

Следующее требование – это *поддержка со стороны разработчика*. Данное понятие включает в себя целый ряд предоставляемых услуг, таких как получение новых версий программного обеспечения бесплатно или с существенной скидкой; получение дополнительной методической литературы, круглосуточной консультации по горячей линии; получение информации о других программных продуктах разработчика; возможность участия в семинарах, научно-практических конференциях пользователей и других мероприятиях, проводимых разработчиком или группами пользователей, и т. д.

Еще одним требованием является *сопровождение*. В процессе эксплуатации сложных программно-технических комплексов могут возникать ситуации, требующие на месте оперативного вмешательства

квалифицированного персонала фирмы-разработчика или ее представителя. Сопровождение включает в себя выезд специалиста на объект заказчика для устранения последствий аварийных ситуаций, методическую и практическую помощь при внесении изменений в систему, не носящих характер радикальной реструктуризации или новой разработки.

**Международные стандарты в области КИС.** Современный рынок требует, чтобы вся продукция удовлетворяла общепризнанным стандартам качества, которые касаются не только качества конечного продукта, выставляемого на рынок, но и всего процесса производства этого изделия, начиная от выбора поставщиков комплектующих и заканчивая сервисным обслуживанием готового продукта. В настоящее время всемирное распространение получил комплекс стандартов на систему качества предприятия, разработанный ISO (International Standards Organization). Этот комплекс стандартов имеет общее название ISO9000 (ИСО 9000). Часть стандартов этого комплекса регламентирует функциональные элементы, присутствующие в КИС.

Внедрение и поддержание на предприятии системы контроля качества в соответствии со стандартами серии ИСО 9000 предполагает использование программных продуктов, по крайней мере, трех классов:

1. Комплексные системы управления предприятием (автоматизированные информационные системы поддержки принятия управленческих решений), АИСППР.

2. Системы электронного документооборота.

3. Продукты, позволяющие создавать функциональные модели организации, проводить анализ и оптимизацию ее деятельности.

Наличие корпоративной информационной системы позволяет поддерживать требуемый ИСО 9000 уровень качества с меньшими затратами на ведение документооборота и на принятие решений. Таким образом, внедрение на производстве системы контроля качества выпускаемой продукции, соответствующей стандарту ИСО 9000, и внедрение корпоративной информационной системы на предприятии взаимосвязаны. Это позволяет дать следующее (функциональное) определение корпоративной информационной системы.

Корпоративная информационная система (КИС) – это совокупность информационных систем отдельных подразделений предприятия, объединенных общим документооборотом, в котором каждая из систем выполняет часть задач по управлению принятием решений, а все системы в целом обеспечивают функционирование предприятия в соответствии с требованиями стандартов качества ИСО 9000.

За всю историю существования корпоративных информационных систем разработано множество стандартов, регламентирующих их. Та или иная система, отвечающая по своему функционалу определенному стандарту, относится к классу этого стандарта и автоматически называется так же, как и он.

На рис. 5.1 представлен эволюционный путь развития наиболее известных стандартов.

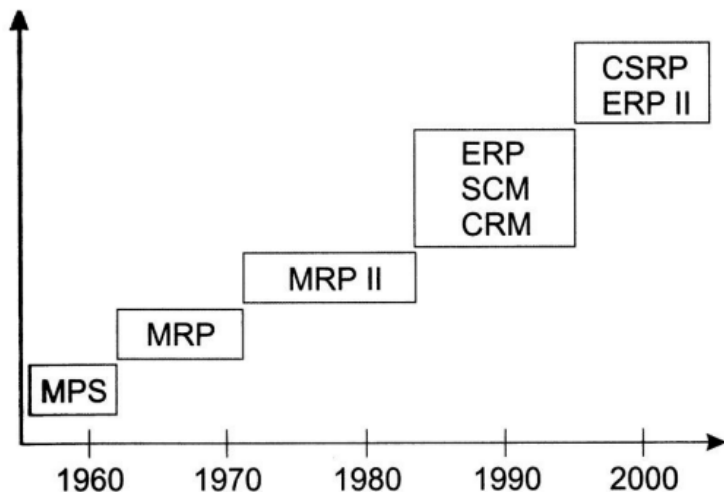


Рис. 5.1. Эволюционный путь развития стандартов КИС

Исходным стандартом, появившимся в конце 50-х – начале 60-х гг. XX в., был стандарт MPS (Master Planning Scheduling – управление календарным планированием), предназначенный для составления основного плана производства. Планы выпуска конечной продукции разрабатывались на основании данных о состоянии спроса.

С целью оптимального управления производством в середине 60-х гг. были сформулированы принципы управления материальными запасами предприятия. Эти принципы легли в основу систем класса MRP (Material Requirement Planning – планирование материальных потребностей). Эти системы предназначены для расчёта возможности выполнения нового заказа к указанному сроку при заданной загрузке производства. При условиях невозможности выполнения данного зака-

за конкретному сроку система способна ответить на вопрос, во что обойдется выполнение нового заказа, если заказчик все же настаивает на первоначальном сроке.

Затем появились системы класса MRPII (Manufacturing Resource Planning – планирование производственных ресурсов), основным назначением которых было прогнозирование, планирование и контроль производства, которые осуществлялись по всему циклу, начиная от закупки сырья и заканчивая отгрузкой конечного продукта потребителю. В общем случае они обеспечивали решение задач планирования деятельности предприятия в натуральных единицах и финансовое планирование в денежном выражении.

Следующим этапом в развитии КИС стало появление с конца 80-х гг. систем класса ERP (Enterprise Resource Planning – планирование ресурсов предприятия). Эти системы охватывают всю финансово-хозяйственную и производственную деятельность предприятия. К ним предъявляются такие требования, как централизация данных в единой базе; режим работы, близкий к реальному времени; сохранение общей модели управления для предприятий любых отраслей; поддержка территориально распределенных структур; работа в широком круге аппаратно-программных платформ и СУБД. Другими важными требованиями, предъявляемыми к ERP-системам, являются: применение графики; использование CASE-технологий для дальнейшего развития системы; поддержка архитектуры типа «клиент-сервер» и реализация их как открытых систем. При правильном внедрении и эксплуатации таких систем эффективность бизнес-процессов предприятия повышается, что дает конкурентное преимущество для дальнейшего развития. Однако занимаясь улучшением внутренней структуры, предприятие не повышает степень эффективности взаимодействия с контрагентами (внешними организациями и фирмами).

Следующий этап развития КИС ориентирован на интеграцию деятельности заказчиков и партнеров предприятия в его внутреннюю систему и называется ERPPII (Enterprise Resource and Relationship Processing – обработка данных по ресурсам и взаимоотношениям предприятия). Развитие Интернета и внутрикорпоративных сетей предоставляет предприятию возможность взаимодействовать со всеми его контрагентами в совершенно новой среде, позволяющей контактировать напрямую с потребителем по типу B2C (Business-to-Consumer – «бизнес-клиент») и (или) с партнерами по бизнесу по типу B2B (Business-to-Business – «бизнес-бизнес»).

Для использования ERP-системы в электронной коммерции и бизнесе необходимо создать приложения по управлению связями с заказчиком CRM (Customer Relationship Management – управление взаимоотношениями с клиентом), а также дополнительное программное обеспечение (ПО) промежуточного слоя. Такое ПО носит название EAI (Enterprise Application Integration – интеграция приложений масштаба предприятия). EAI-системы обеспечивают следующие функции:

- 1) электронная коммерция;
- 2) управление цепочкой поставок;
- 3) услуги доступа к приложениям;
- 4) виртуальные торговые площадки.

ERP-система, снабженная CRM- и EAI-продуктами, получила название XRP-система, т. е. расширенная (Extended) ERP-система. Она позволяет в режиме реального времени разделять данные, используемые различными корпоративными приложениями. По классификации эта система приближается к следующему поколению КИС – системам стандарта CSRP (Customer Synhronized Resource Planning – планирование ресурсов совместно с потребителем). Системы такого класса позволяют интегрировать в единое целое процессы как внутри одной корпорации, так и за ее пределами.

*Стандарт MPS.* В конце 50-х – начале 60-х гг. XX в. в связи с ростом популярности вычислительных систем возникла идея использовать их возможности для планирования деятельности предприятия, в том числе для планирования производственных процессов. Необходимость планирования обусловлена тем, что основная масса задержек в процессе производства связана с запаздыванием поставок отдельных комплектующих. В результате этого параллельно с уменьшением эффективности производства на складах появляется избыток материалов, поступивших в срок или ранее намеченного срока. Кроме того, вследствие нарушения баланса поставок комплектующих возникают дополнительные осложнения с учетом и отслеживанием их состояния в процессе производства, т. е. невозможно определить, к какой партии принадлежит данный составляющий элемент в уже собранном готовом продукте (изделии).

*Схема функционирования MPS-системы.* Первым стандартом управления бизнесом был MPS (Master Planning Scheduling), или стандарт объемно-календарного планирования. Идея стандарта очевидна и представлена на рис. 5.2. Сначала формируется план продаж, т. е. определяется объем продаж с распределением по календарным

периодам. На основе этого плана продаж составляется план пополнения запасов за счет производства или закупки, и оцениваются финансовые результаты за различные периоды, в качестве которых используются периоды планирования или финансовые периоды.



Рис. 5.2. Схема функционирования КИС по стандарту MPS

Данный стандарт оптимально описывает модель функционирования небольшого торгового предприятия с довольно простой схемой производства.

Если предприятие динамично развивается, то возникает ряд сложностей. Первые проблемы появляются с управлением логистикой, так как сформированный заказ на необходимые материалы может поступить не в тот срок, который был намечен ранее.

Одной из наиболее сложных проблем, возникающих при формировании заказа, была проблема прогнозирования необходимого объема и срока поставки, который во многом зависит от возможностей поставщика. Следовательно, нужно прогнозировать спрос на длительное время вперед, учитывать длительность (а часто и сезон) производственного процесса и потребности в складских площадях. При этом объем заказа материалов также не может быть выражен в произвольных цифрах. Такие проблемы возникают при оптовых (крупных) продажах.

*Стандарт MRP.* В результате частичного решения обозначенных проблем возникла методология планирования производств (в основном сборочных или дискретных), которая предлагала формировать заказ на комплектующие и узлы, опираясь на потребности объемно-календарного плана производства, получившая название MRP (Material Requirement Planning).

КИС, работающая в соответствии с методологией MRP, представляет собой компьютерную программу, позволяющую оптимально ре-



гулировать поставки комплектующих для производственного процесса, контролируя складские запасы и саму технологию производства. Главной задачей MRP является обеспечение необходимого количества всех требуемых материалов и комплектующих в любой момент времени в рамках срока планирования, наряду с возможным уменьшением постоянных запасов, а следовательно, разгрузкой склада. Прежде чем описывать саму структуру MRP, следует ввести краткий список основных ее понятий.

Материалами называют все сырье и отдельные комплектующие, составляющие конечный продукт.

MRP-программа – компьютерная программа, работающая по алгоритму, регламентированному MRP-методологией. Как и любая программа, она обрабатывает файлы данных (входные элементы) и формирует на их основе файлы-результаты.

Статус материала является основным указателем на текущее состояние материала. Каждый отдельный материал в конкретный момент времени имеет статус в рамках MRP-системы, который определяет, имеется ли данный материал в наличии на складе, зарезервирован ли он для других целей, присутствует ли в текущих заказах или заказ на него только планируется.

Страховой запас материала необходим для поддержания процесса производства в случае возникновения непредвиденных и неустраняемых задержек в его поставках.

Потребность в материале в MRP-программе представляет собой определенную количественную единицу, отображающую возникшую в некоторый момент времени (в течение периода планирования) необходимость в заказе данного материала. Различают понятия полной потребности в материале, которая определяет необходимое количество, требуемое для отправки в производство, и чистой потребности, при вычислении которой учитывается наличие всех страховых и зарезервированных запасов данного материала. Заказ в системе автоматически формируется при возникновении отличной от нуля чистой потребности.

Процесс планирования предусматривает автоматическое создание проектов заказов на закупку и (или) внутреннее производство необходимых материалов. Другими словами, MRP-система оптимизирует время поставки комплектующих, тем самым уменьшая затраты на производство и повышая его эффективность. Основными преимуществами использования MRP-системы в производстве являются:

- гарантия наличия на складе необходимых комплектующих и уменьшение временных задержек их поставки;
- уменьшение производственного брака в процессе сборки готовой продукции, возникающего из-за использования некачественных комплектующих;
- упорядочение производства ввиду контроля статуса каждого материала, позволяющего однозначно отслеживать весь его конвейерный путь, начиная от создания заказа до его положения в уже собранном готовом изделии. Благодаря этому достигается полная достоверность и эффективность производственного учета.

*Стандарт MRP II.* MRP-системы на основе утвержденной производственной программы формировали план заказов на определенный период, что было явно недостаточным для многих предприятий. С целью увеличения эффективности планирования в конце 1970-х гг. была предложена концепция реализации замкнутого цикла (Closed Loop) в MRP-системах. Идея заключалась в рассмотрении более широкого спектра факторов при проведении планирования путем введения дополнительных функций. К базовым функциям планирования производственных мощностей (CRP, Capacity Requirements Planning) и планирования потребностей в материалах (MRP) было предложено добавить ряд дополнительных, таких как контроль соответствия количества произведенной продукции количеству использованных в процессе сборки комплектующих, составление регулярных отчетов о задержках заказов, об объемах и динамике продаж продукции, о поставщиках и т. д.

В процессе дальнейшего анализа ситуации, сложившейся в бизнесе, и ее развития оказалось, что большую часть себестоимости продукции составляют затраты, несвязанные напрямую с процессом и объемом производства. Из-за постоянно растущей конкуренции конечные потребители продукции становятся все более разборчивыми, ощутимо увеличиваются затраты на рекламу и маркетинг, уменьшается жизненный цикл изделий. Все это требует пересмотра взглядов на планирование коммерческой деятельности. Исходя из этих предпосылок, зародилась новая концепция корпоративного планирования, названная MRP II.

*Стандарт CSRP.* В начале нового века появились более мощные инструменты управления производством, которые построены на твердом фундаменте ERP и направлены на интеграцию с покупателями. Система планирования производства этого десятилетия имеет два фо-

куса: на производственной эффективности и на создании покупательской ценности. Эта новая парадигма-стратегия получила название планирование ресурсов, синхронизированное с покупателем CSRP (Customer Synchronized Resource Planning).

*Основные преимущества CSRP-систем.* При использовании модели бизнеса CSRP традиционные бизнес-процессы переадресуются на обслуживание покупателей и создание продуктов, удовлетворяющих их потребностям. Внедрение приложений CSRP заставляет руководителей предприятия гибко реагировать на изменения. Внутренняя сфокусированность традиционных производственных структур, сегментированная по отделам и функциональности, меняет фокусировку. CSRP позволяет построить двунаправленный свободный поток информации между покупателем и производителем.

Рассмотренная концепция построения информационных систем позволяет реализовать более грамотное управление производственными графиками в условиях ограниченных мощностей (так называемую APS-задачу – Advanced Planning and Scheduling – расширенное управление производственными графиками).

Системы типа APS решают такие задачи, как выполнение срочного заказа в рамках производственного графика, распределение заданий с учетом приоритетов и ограничений, перепланирование с использованием полноценного графического интерфейса. Благодаря принципиально новому математическому аппарату, применяемому в CSRP-системах, расчет типовых MRP-задач осуществляется значительно быстрее, чем раньше. Типичный пример ситуации, при которой подчеркивается эффективность применения APS-систем, – дополнительный срочный заказ на предприятии, где уже сформирована и выполняется производственная программа. При всей привлекательности выполнение нового заказа может повлечь за собой серьезные последствия: несвоевременное выполнение ранее принятых заказов, сбой в производственных циклах и в конечном итоге финансовые потери для предприятия. В этом случае необходимо принять решение, соглашаться ли на выполнение заказа, и если да, то какова должна быть его стоимость для покупателя.

Реализация концепции CSRP на конкретном предприятии позволяет управлять заказами клиентов и всей работой с ними на порядок точнее, чем при использовании предыдущих систем. Стало возможным ежечасное изменение производственного графика – при приеме каждого нового заказа можно полностью пересчитать производственную

программу с учетом приоритетных стратегий предприятия. При использовании классической ERP-системы это было практически непосильной задачей. Расчет детальной стоимости заказа или отдельных его составляющих стал возможен уже на этапе его формирования. Появилась возможность учесть вариации спецификации изделия или технологической цепочки, что часто требуется в полиграфической и других отраслях промышленности. Также учитываются все дополнительные операции по тестированию и административному обслуживанию заказа, не говоря уже о послепродажном обслуживании.

В рамках CSRP и аналогичных ей методик очень важна интеграция с системой управления ресурсами предприятия программных продуктов третьих фирм, реализующих специфические задачи управления и расчета ресурсов, характерных для конкретного предприятия. В целом применение новых методик управления ресурсами предприятия позволяет компании чувствовать себя уверенно даже в условиях нестабильности рынка и быстро меняющегося макроэкономического окружения. CSRP заставляет пересмотреть всю бизнеспрактику, фокусируя внимание на рыночной активности, а не на производственной деятельности. Бизнес-процессы синхронизируются с деятельностью покупателей. Для примера рассмотрим процесс обработки заказов. Планирование ресурсов, синхронизированное с потребностями покупателей (CSRP), предлагает модель бизнеса и набор инструментов, которые способны сделать партнерство с покупателем возможным и привычным.

Планирование ресурсов, синхронизированное с покупателем (CSRP), предлагает новый набор бизнес-правил, которые позволяют производителю создать покупательскую ценность – разработать решения и услуги, которые сделают их (производителей) востребованными на рынке. Бизнес-функции, предлагаемые CSRP-системами, позволяют производителю индивидуализировать продукцию, разработать решения и услуги, которые подчеркнут его ценность для клиентов. Все чаще конкурентные преимущества определяются как способность производителей в любое время удовлетворить уникальные потребности каждого отдельного покупателя и делать это постоянно.

*Стандарт ERP.* Системы ERP (Enterprise Resource Planning System – система планирования ресурсов предприятия) предназначены для управления финансовой и хозяйственной деятельностью предприятий. Это «верхний уровень» в иерархии систем управления предприятием, затрагивающий ключевые аспекты его производственной и ком-

мерческой деятельности. Такие системы создаются для предоставления руководству информации, способствующей принятию управленческих решений, а также для создания инфраструктуры электронного обмена данными предприятия с поставщиками и потребителями.

Все предприятия уникальны в своей финансовой и хозяйственной деятельности. Однако прогресс в области разработки программных решений для задач ERP обусловлен тем, что наряду со спецификой удастся выделить задачи, общие для предприятий самых разных сфер деятельности (различные отрасли промышленности, сфера услуг, телекоммуникации, банки, государственные учреждения и др.). К таким общим задачам можно отнести управление материальными и финансовыми ресурсами, закупками, сбытом, заказами потребителей и поставками, управление кадрами, основными фондами, складами, бизнес-планирование и учет, бухгалтерию, расчеты с покупателями и поставщиками, ведение банковских счетов и прочее.

Со временем MRPII-система превратилась в систему планирования ресурсов предприятия ERP (Enterprise Resource Planning), также называемую иногда системой планирования ресурсов в масштабе предприятия. В основе ERP лежит принцип создания единого хранилища данных, содержащего всю деловую информацию, накопленную организацией в процессе ведения деловых операций, включая финансовую информацию, данные, связанные с производством, управлением персоналом, или различные другие сведения.

Такой подход устраняет необходимость передачи данных между отдельными системами. Кроме того, любая часть информации, которой располагает организация, становится доступной одновременно для всех работников, обладающих соответствующими полномочиями. После того как была доказана применимость концепции ERP в условиях производства, стало возможным создание единого информационного ресурса, используемого коммерческой организацией в целом.

Системы ERP, в отличие от MRPII, ориентированы на управление «виртуальным предприятием», отражающим взаимодействие производства, поставщиков, партнеров и потребителей, представляющих собой автономно работающие предприятия или корпорации.

В ERP добавляются механизмы управления транснациональными корпорациями, включая поддержку нескольких часовых поясов, языков, валют, систем бухгалтерского учета и отчетности.

Увеличиваются требования к интеграции систем ERP с приложениями, уже используемыми на предприятии. В новых системах ERP

большое внимание уделяется средствам поддержки принятия решений и средствам интеграции с хранилищами данных (иногда включаемых в систему как новый модуль).

*Стандарт ERP II.* Авторитетная консалтинговая компания Gartner Group заявила о завершении эпохи ERP-систем в 1999 г. На смену была предложена концепция управления внутренними ресурсами и внешними связями предприятия ERP II (Enterprise Resource and Relationship Processing). Основная идея ERP II-системы заключается в выходе за рамки задач по оптимизации внутренних процессов организации.

В настоящее время ERP II-системы являются новейшей разработкой в сфере информационных управленческих решений. Исторически развитие новых концепций протекало в русле поглощения отработанных и зарекомендовавших себя стандартов и формирования на их основе новых, проверенных решений (рис. 5.3).

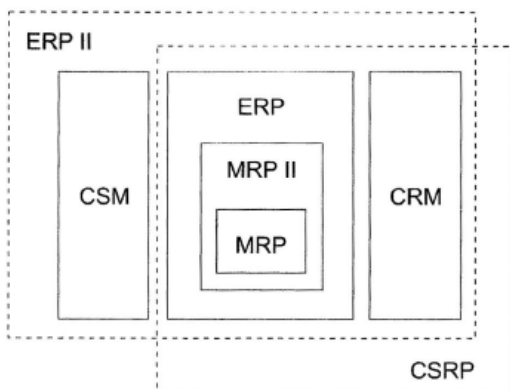


Рис. 5.3. Стандарты управления предприятием

Концепция ERP II предполагает развитие базовой ERP-функциональности путем совершенствования существующих модулей, т. е. в системах от версии к версии появляются все новые дополнительные возможности в финансовом, логистическом и производственном контурах, реализуется ведение раздельного учета по разным юридическим лицам в рамках одного приложения с общей базой данных.

**Отличия ERP II от ERP-систем.** Gartner Group определяет ERP II как бизнес-стратегию и набор специфичных для отрасли приложений, которые позволяют осуществлять внутренние и внешние бизнес-

процессы, совместные операционные и финансовые инициативы и дают возможность их оптимизации. Ключевые финансовые сферы ERP-II – это бухгалтерский учет, купля-продажа продукции и материалов, ввод заказов и калькуляция себестоимости.

Назначение ERP-II заключается не только в оптимизации ресурсов и обработке транзакций традиционных ERP-систем, но и в использовании информации. Данные функции ERP включает в процесс сотрудничества между предприятиями. Таким образом, роль ERP не ограничивается только осуществлением купли-продажи в рамках электронной коммерции. Предметная область ERP-II распространяется за пределы ERP и затрагивает непроизводственные отрасли.

Внутренние функции этих отраслей выходят за пределы широкого понимания производства, распространения, финансов и объединяют специфические для отраслевого сектора или какой-либо отдельной отрасли действия.

Сконцентрированные в сети Интернет, рассчитанные на интеграцию архитектуры продуктов ERP-II, они настолько отличаются от монолитных архитектур ERP, что требуют полной трансформации. ERP-II расширяют функции ERP, такие как хранение всех данных внутри предприятия, до возможностей работы с данными, распределенными по торговому сообществу. В табл. 5.1 представлены основные отличия ERP-II от ERP.

Таблица 5.1. Основные концепции CRM-стратегии

| Параметр                          | ERP-система                                                      | ERP-II-система                                                                                  |
|-----------------------------------|------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| Роль системы на предприятии       | Оптимизация процессов                                            | Участие в цепочке, обеспечивающей увеличение прибыли, создание условий для совместной коммерции |
| Область применения                | Производство и дистрибуция                                       | Все сегменты и секторы                                                                          |
| Функции                           | Производство, торговля (дистрибуция) и финансовые процессы       | Межотраслевые и отраслевые сектора, специфические производственные процессы                     |
| Типы автоматизированных процессов | Внутренние, скрытые                                              | Связаны на внешнем уровне                                                                       |
| Архитектура                       | Монолитная, закрытая, как правило, основанная на веб-технологиях | Открытая, компонентная, интернет-ориентированная                                                |
| Данные                            | Генерируемые и используемые внутри предприятия                   | Предназначены для внутреннего и внешнего использования                                          |

Таким образом, ERP II – это результат развития методики и технологии ERP в направлении более тесного взаимодействия предприятия с его клиентами и контрагентами. При этом управленческая информация компании не только используется для внутренних целей, но и служит для развития отношений сотрудничества с другими организациями.

Помимо новой управленческой ориентации системы ERP II характеризуются и некоторыми технологическими особенностями. Здесь имеется в виду интернет-ориентированная архитектура, которая существенно отличается от архитектуры традиционных ERP-систем. Это обусловлено тем, что управленческая информация, ранее хранимая и применяемая только внутри предприятия, теперь должна быть доступной для информационных систем клиентов и партнеров.

Таким образом, традиционная клиент-серверная архитектура начинает уступать место веб-клиентам и распределенным компонентным технологиям.

В концептуальном построении традиционных систем управления ресурсами предприятия ERP клиент рассматривается как элемент внешнего мира, не интегрированный в обслуживаемые ERP-системой бизнес-процессы. Смысл такого устройства систем для управления компанией заключается в стратегическом фокусировании бизнеса на оптимизации только внутренней деятельности самого предприятия, что в настоящее время устарело. Многие подразделения предприятия, контактирующие с внешним миром, разобщены, хотя взаимодействуют с одними и теми же контрагентами. Отсутствие единого подхода в работе с клиентом оказывает негативное влияние на эффективность работы на рынке – компания теряет возможность увеличить продажи и повысить уровень лояльности клиентов.

Интегрировать клиента внутрь компании, предоставить ему индивидуальное обслуживание – основная задача бизнес-планирования. В рамках этой задачи появилась целая стратегия, направленная на смещение усилий по наведению порядка внутри компании в сторону обслуживания клиентов, названная CRM (Customer Relationship Management).

CRM – это стратегия компании, определяющая взаимодействие с клиентами во всех организационных аспектах: она касается рекламы, продажи, доставки и обслуживания клиентов, дизайна и производства новых продуктов, выставления счетов и т. п.

Стратегия основана на выполнении следующих условий:



- наличие единого хранилища информации, содержащего сведения обо всех случаях взаимодействия с клиентами;
- синхронизация управления множественными каналами взаимодействия (т. е. существуют организационные процедуры, которые регламентируют использование этой системы и информации в каждом подразделении компании);
- постоянный анализ собранной информации о клиентах и принятие соответствующих организационных решений, например о ранжировании клиентов, исходя из их значимости для компании, выработке индивидуального подхода к клиентам согласно их специфическим потребностям и запросам.

## **5.2. Системы искусственного интеллекта**

**Данные и знания. Базы знаний.** При изучении интеллектуальных систем традиционно возникает вопрос – что же такое знания и чем они отличаются от обычных данных, десятилетиями обрабатываемых ЭВМ. Можно предложить несколько рабочих определений, в рамках которых это становится очевидным.

Данные – это отдельные факты, характеризующие объекты, процессы и явления в предметной области, а также их свойства.

При обработке на ЭВМ данные трансформируются, условно проходя следующие этапы:

- данные как результат измерений и наблюдений;
- данные на материальных носителях информации (таблицы, протоколы, справочники);
- модели (структуры) данных в виде диаграмм, графиков, функций;
- данные в компьютере на языке описания данных;
- базы данных на машинных носителях.

Знания связаны с данными, основываются на них, но представляют результат мыслительной деятельности человека, обобщают его опыт, полученный в ходе выполнения какой-либо практической деятельности. Они получают эмпирическим путем.

Знания – это выявленные закономерности предметной области (принципы, связи, законы), позволяющие решать задачи в этой области.

При обработке на ЭВМ знания трансформируются аналогично данным:

- знания в памяти человека как результат мышления;
- материальные носители знаний (учебники, методические пособия);

- поле знаний – условное описание основных объектов предметной области, их атрибутов и закономерностей, их связывающих;
- знания, описанные на языках представления знаний (продукционные языки, семантические сети, фреймы – см. далее);
- базы знаний.

Часто используется такое определение знаний:

– знания – это хорошо структурированные данные, или данные о данных, или метаданные.

Для хранения данных используются базы данных (для них характерны большой объем и относительно небольшая удельная стоимость информации), для хранения знаний – базы знаний (небольшого объема, но исключительно дорогие информационные массивы). База знаний – основа любой интеллектуальной системы.

Знания могут быть классифицированы по следующим категориям:

- поверхностные – знания о видимых взаимосвязях между отдельными событиями и фактами в предметной области;
- глубинные – абстракции, аналогии, схемы, отображающие структуру и процессы в предметной области.

Современные экспертные системы работают в основном с поверхностными знаниями. Это связано с тем, что на данный момент нет адекватных моделей, позволяющих работать с глубинными знаниями.

Кроме того, знания можно разделить на процедурные и декларативные. Исторически первичными были процедурные знания, т. е. знания, «растворенные» в алгоритмах. Они управляли данными. Для их изменения требовалось изменять программы. Однако с развитием искусственного интеллекта приоритет данных постепенно изменялся, и все большая часть знаний сосредоточивалась в структурах данных (таблицы, списки, абстрактные типы данных), т. е. увеличивалась роль декларативных знаний.

Сегодня знания приобрели чисто декларативную форму, т. е. знаниями считаются предложения, записанные на языках представления знаний, приближенных к естественному и понятных неспециалистам.

Существуют десятки моделей (или языков) представления знаний для различных предметных областей. Большинство из них может быть сведено к следующим классам:

- продукционные;
- семантические сети;
- фреймы;
- формальные логические модели.

### ***Модели представления знаний.***

***Продукционная модель.*** Продукционная модель, или модель, основанная на правилах, позволяет представить знания в виде предложений типа: ***Если (условие), то (действие).***

Под условием понимается некоторое предложение-образец, по которому осуществляется поиск в базе знаний, а под действием – действия, выполняемые при успешном исходе поиска (они могут быть промежуточными, выступающими далее как условия, и терминальными или целевыми, завершающими работу системы).

При использовании продукционной модели база знаний состоит из набора правил. Программа, управляющая перебором правил, называется машиной вывода. Чаще всего вывод бывает прямой (от данных к поиску цели) или обратный (от цели для ее подтверждения – к данным). Данные – это исходные факты, на основании которых запускается машина вывода – программа, перебирающая правила из базы.

**Пример 1.** Имеется фрагмент базы знаний из двух правил: П1: Если «отдых – летом» и «человек – активный», то «ехать в горы». П2: Если «любит солнце», то «отдых летом».

Предположим, в систему поступили данные – «человек активный» и «любит солнце». Прямой вывод – исходя из данных, получить ответ.

1-й проход.

Шаг 1. Пробуем П1, не работает (не хватает данных «отдых – летом»).

Шаг 2. Пробуем П2, работает, в базу поступает факт «отдых летом».

2-й проход.

Шаг 3. Пробуем П1, работает, активируется цель «ехать в горы», которая и выступает как совет, который дает ЭС.

Обратный вывод – подтвердить выбранную цель с помощью имеющихся правил и данных.

1-й проход.

Шаг 1. Цель – «ехать в горы»: пробуем П1 – данных «отдых – летом» нет, они становятся новой целью, и ищется правило, где она в правой части.

Шаг 2. Цель «отдых – летом»: правило П2 подтверждает цель и активирует ее.

2-й проход.

Шаг 3. Пробуем П1, подтверждается искомая цель.

Продукционная модель чаще всего применяется в промышленных экспертных системах. Она привлекает разработчиков своей наглядностью, высокой модульностью, легкостью внесения дополнений и изменений и простотой механизма логического вывода.

Имеется большое число программных средств, реализующих продукционный подход (язык OPS 5; «оболочки» или «пустые» ЭС – EXSYS, ЭКСПЕРТ; инструментальные системы ПИЭС и СПЭИС и др.), а также промышленных ЭС на его основе (ФИАКР) и др.

**Семантические сети.** Термин семантическая означает смысловая, а сама семантика – это наука, устанавливающая отношения между символами и объектами, которые они обозначают, т. е. наука, определяющая смысл знаков.

Семантическая сеть – это ориентированный граф, вершины которого – понятия, а дуги – отношения между ними.

Понятиями обычно выступают абстрактные или конкретные объекты, а отношения – это связи типа: «это» («is»), «имеет частью» («has part»), «принадлежит», «любит». Характерной особенностью семантических сетей является обязательное наличие трех типов отношений:

- класс – элемент класса;
- свойство – значение;
- пример элемента класса.

Можно ввести несколько классификаций семантических сетей. Например, *по количеству типов отношений*:

- однородные (с единственным типом отношений);
- неоднородные (с различными типами отношений).

*По типам отношений*:

- бинарные (в которых отношения связывают два объекта);
- *n*-арные (в которых есть специальные отношения, связывающие более двух понятий). Наиболее часто в семантических сетях используются следующие отношения:

- связи типа «часть-целое» («класс-подкласс», «элемент-множество» и т. п.);
- функциональные связи (определяемые обычно глаголами «производит», «влияет» ...);
- количественные (больше, меньше, равно ...);
- пространственные (далеко от, близко от, за, под, над ...);
- временные (раньше, позже, в течение ...);
- атрибутивные связи (иметь свойство, иметь значение ...);
- логические связи (и, или, не) и др.

Проблема поиска решения в базе знаний типа семантической сети сводится к задаче поиска фрагмента сети, соответствующего некоторой подсети, соответствующей поставленному вопросу.

**Пример.** На рис. 5.4 изображена семантическая сеть. В качестве вершин – понятия: Человек, Иванов, Волга, Автомобиль, Вид транспорта, Двигатель.

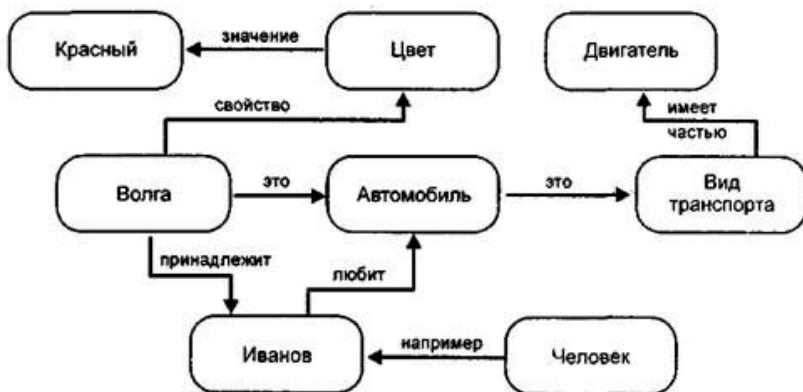


Рис. 5.4. Семантическая сеть

Основное преимущество этой модели – в соответствии с современными представлениями об организации долговременной памяти человека. Недостаток модели – сложность поиска вывода на семантической сети.

Для реализации семантических сетей существуют специальные сетевые языки, например NET и др. Широко известны экспертные системы, использующие семантические сети в качестве языка представления знаний – PROSPECTOR, CASNET, TORUS.

**Фреймы.** Фрейм (англ. *frame* – каркас или рамка) предложен М. Минским в 70-е гг. как структура знаний для восприятия пространственных сцен. Эта модель, как и семантическая сеть, имеет глубокое психологическое обоснование.

Под фреймом понимается абстрактный образ или ситуация. В психологии и философии известно понятие абстрактного образа. Например, слово «комната» вызывает у слушающих образ комнаты: жилое помещение с четырьмя стенами, полом, потолком, окнами и дверью, площадью 6–20 м<sup>2</sup>». Из этого описания ничего нельзя убрать (напри-

мер, убрав окна, мы получим уже чулан, а не комнату), но в нем есть «дырки», или «слоты», – это незаполненные значения некоторых атрибутов – количество окон, цвет стен, высота потолка, покрытие пола и др.

В теории фреймов такой образ называется фреймом. Фреймом называется также и формализованная модель для отображения образа.

Структуру фрейма можно представить так:

– *имя фрейма*: (имя 1-го слота: значение 1-го слота), (имя 2-го слота: значение 2-го слота), (имя N-го слота: значение N-го слота).

Ту же запись представим в виде схемы, дополнив двумя столбцами.

| Имя фрейма |           |                |                          |
|------------|-----------|----------------|--------------------------|
| имя слота  | тип слота | значение слота | присоединенная процедура |

В схеме дополнительные столбцы предназначены для описания типа слота и возможного присоединения к тому или иному слоту специальных процедур, что допускается в теории фреймов. В качестве значения слота может выступать имя другого фрейма; так образуют сети фреймов.

Различают фреймы-образцы, или прототипы, хранящиеся в базе знаний, и фреймы-экземпляры, которые создаются для отображения реальных ситуаций на основе поступающих данных.

Модель фрейма является достаточно универсальной, поскольку позволяет отобразить все многообразие знаний о мире:

- через фреймы-структуры, для обозначения объектов и понятий (заем, залог, вексель);

- фреймы-роли (менеджер, кассир, клиент);

- фреймы-сценарии (банкротство, собрание акционеров, празднование именин);

- фреймы-ситуации (тревога, авария, рабочий режим устройства) и др. Важнейшим свойством теории фреймов является заимствованное из теории семантических сетей наследование свойств. И во фреймах, и в семантических сетях наследование происходит по АКО-связям (A-Kind-Of = это). Слот АКО указывает на фрейм более высокого уровня иерархии, откуда неявно наследуются, т. е. переносятся, значения аналогичных слотов.

**Пример 2.** Например, в сети фреймов, представленной на рис. 5.5, понятие «ученик» наследует свойства фреймов «ребенок» и «человек», которые находятся на более высоком уровне иерархии. Так, на вопрос: «Любят ли ученики сладкое?» Следует ответ: «Да», так как этим свой-

ством обладают все дети, что указано во фрейме «ребенок». Наследование свойств может быть частичным, так, возраст для учеников не наследуется из фрейма «ребенок», поскольку указан явно в своем собственном фрейме.

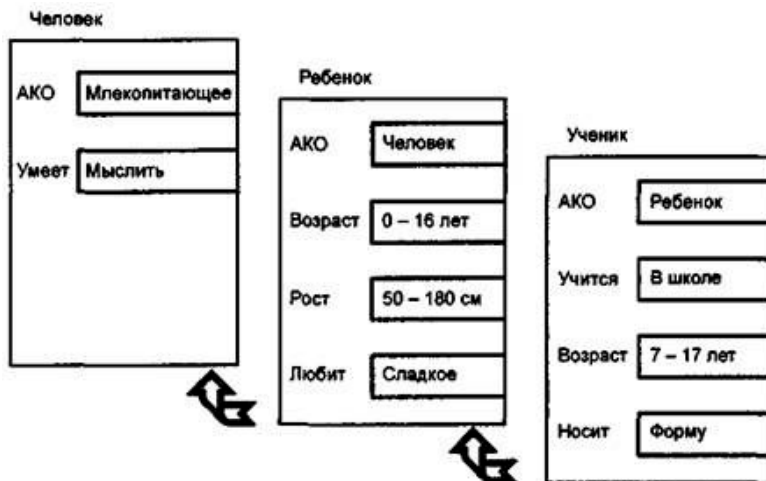


Рис. 5.5. Сеть фреймов

Основным преимуществом фреймов как модели представления знаний является способность отражать концептуальную основу организации памяти человека, а также ее гибкость и наглядность.

Специальные языки представления знаний в сетях фреймов FRL (Frame Representation Language) и другие позволяют эффективно строить промышленные ЭС. Широко известны такие фреймо-ориентированные экспертные системы, как ANALYST, МОДИС.

**Формальные логические модели.** Традиционно в представлении знаний выделяют формальные логические модели, основанные на классическом исчислении предикатов I порядка, когда предметная область или задача описывается в виде набора аксиом. Мы опустим описание этих моделей, так как эта логическая модель применима в основном в исследовательских «игрушечных» системах и предъявляет очень высокие требования и ограничения к предметной области.

**История развития искусственного интеллекта за рубежом.** Идея создания искусственного подобия человеческого разума для решения

сложных задач и моделирования мыслительной способности волновала людей с древнейших времен. Впервые ее выразил Р. Луллий (ок. 1235 – ок. 1315), который еще в XIV в. пытался создать машину для решения различных задач на основе всеобщей классификации понятий.

В XVIII в. Г. Лейбниц (1646–1716) и Р. Декарт (1596–1650) независимо друг от друга развили эту идею, предложив универсальные языки классификации всех наук. Эти идеи легли в основу теоретических работ в области создания искусственного интеллекта.

Развитие искусственного интеллекта как научного направления стало возможным только после создания ЭВМ. Это произошло в 40-х гг. XX в. В это же время Н. Винер (1894–1964) создал свои основополагающие работы по новой науке – кибернетике.

Термин искусственный интеллект (*artificial intelligence*) предложен в 1956 г. на семинаре с аналогичным названием в Станфордском университете (США). Семинар был посвящен разработке логических, а не вычислительных задач. Вскоре после признания искусственного интеллекта самостоятельной отраслью науки произошло разделение на два основных направления: нейрокибернетику и кибернетику «черного ящика». И только в настоящее время стали заметны тенденции к объединению этих частей вновь в единое целое.

Основную идею нейрокибернетики можно сформулировать следующим образом. Единственный объект, способный мыслить, – это человеческий мозг. Поэтому любое «мыслящее» устройство должно каким-то образом воспроизводить его структуру.

Таким образом, нейрокибернетика ориентирована на аппаратное моделирование структур, подобных структуре мозга. Физиологами давно установлено, что основой человеческого мозга является большое количество (до 1021) связанных между собой и взаимодействующих нервных клеток – нейронов. Поэтому усилия нейрокибернетики были сосредоточены на создании элементов, аналогичных нейронам, и их объединении в функционирующие системы. Эти системы принято называть нейронными сетями, или нейросетями.

Первые нейросети были созданы в конце 50-х гг. американскими учеными Г. Розенблаттом и П. Мак Каллоком. Это были попытки создать системы, моделирующие человеческий глаз и его взаимодействие с мозгом. Устройство, созданное ими, получило название перцептрона. Оно умело различать буквы алфавита, но было чувствительно к их написанию, например, буквы А, А и А для этого устройства



были тремя разными знаками. Постепенно в 70–80 гг. количество работ по этому направлению искусственного интеллекта стало снижаться. Слишком неутешительными оказались первые результаты. Авторы объясняли неудачи малой памятью и низким быстродействием существующих в то время компьютеров.

Однако в середине 80-х гг. в Японии в рамках проекта разработки компьютера V поколения, основанного на знаниях, был создан компьютер VI поколения, или нейрокомпьютер. К этому времени ограничения по памяти и быстродействию были практически сняты. Появились транспьютеры – параллельные компьютеры с большим количеством процессоров. От транспьютеров был один шаг до нейрокомпьютеров, моделирующих структуру мозга человека. Основная область применения нейрокомпьютеров – распознавание образов.

В настоящее время используются три подхода к созданию нейросетей:

- аппаратный – создание специальных компьютеров, плат расширения, наборов микросхем, реализующих все необходимые алгоритмы;
- программный – создание программ и инструментариев, рассчитанных на высокопроизводительные компьютеры. Сети создаются в памяти компьютера, всю работу выполняют его собственные процессоры;
- гибридный – комбинация первых двух. Часть вычислений выполняют специальные платы расширения (сопроцессоры), часть – программные средства.

В основу кибернетики «черного ящика» лег принцип, противоположный нейрокибернетике. Не имеет значения, как устроено «мыслящее» устройство. Главное, чтобы на заданные входные воздействия оно реагировало так же, как человеческий мозг.

Это направление искусственного интеллекта было ориентировано на поиски алгоритмов решения интеллектуальных задач на существующих моделях компьютеров. В 1956–1963 гг. велись интенсивные поиски моделей и алгоритма человеческого мышления и разработка первых программ. Оказалось, что ни одна из существующих наук – философия, психология, лингвистика – не может предложить такого алгоритма. Тогда кибернетики предложили создать собственные модели. Были созданы и опробованы различные подходы.

В конце 50-х гг. родилась модель лабиринтного поиска. Этот подход представляет задачу как некоторый граф, отражающий простран-

ство состояний, и в этом графе проводится поиск оптимального пути от входных данных к результирующим. Была проделана большая работа по разработке этой модели, но в решении практических задач идея большого распространения не получила.

Начало 60-х гг. – эпоха эвристического программирования. Эвристика – правило, теоретически не обоснованное, но позволяющее сократить количество переборov в пространстве поиска. Эвристическое программирование – разработка стратегии действий на основе известных, заранее заданных эвристик.

В 1963–1970 гг. к решению задач стали подключать методы математической логики. На основе метода резолюций, позволившего автоматически доказывать теоремы при наличии набора исходных аксиом, в 1973 г. создается язык Пролог.

Существенный прорыв в практических приложениях искусственного интеллекта произошел в середине 70-х гг., когда на смену поискам универсального алгоритма мышления пришла идея моделировать конкретные знания специалистов-экспертов. В США появились первые коммерческие системы, основанные на знаниях, или экспертные системы. Пришел новый подход к решению задач искусственного интеллекта – представление знаний. Созданы MYCIN и DENDRAL – ставшие уже классическими экспертные системы для медицины и химии. Объявлено несколько глобальных программ развития интеллектуальных технологий – ESPRIT (Европейский Союз), DARPA (Министерство обороны США), японский проект машин V поколения.

Начиная с середины 80-х гг. происходит коммерциализация искусственного интеллекта. Растут ежегодные капиталовложения, создаются промышленные экспертные системы. Растет интерес к самообучающимся системам.

**История развития искусственного интеллекта в России.** В 1954 г. в МГУ под руководством профессора Л. Л. Ляпунова (1911–1973) начал свою работу семинар «Автоматы и мышление». В этом семинаре принимали участие крупнейшие физиологи, лингвисты, психологи, математики. Принято считать, что именно в это время родился искусственный интеллект в России. Как и за рубежом, выделились направления нейрокибернетики и кибернетики «черного ящика».

Среди наиболее значимых результатов, полученных отечественными учеными, следует отметить алгоритм «Кора» М. Бонгарда, моделирующий деятельность человеческого мозга при распознавании образов (60-е гг.).

В 1945–1964 гг. создаются отдельные программы и исследуется поиск решения логических задач. В Ленинграде (ЛОМИ – Ленинградское отделение математического института им. В. А. Стеклова) создается программа, автоматически доказывающая теоремы (АЛПЕВ ЛОМИ). Она основана на оригинальном обратном выводе С. Ю. Маслова, аналогичном методу резолюций Робинсона.

В 1965–1980 гг. получает развитие новая наука – ситуационное управление (соответствует представлению знаний в западной терминологии). Основоположник этой научной школы – профессор Д. А. Поспелов. Разработаны специальные модели представления ситуаций – представления знаний.

В 1980–1990 гг. проводятся активные исследования в области представления знаний, разрабатываются языки представления знаний, экспертные системы (более 300). В Московском государственном университете создается язык РЕФАЛ.

В 1988 г. создается АИИ – Ассоциация искусственного интеллекта. Ее членами являются более 300 исследователей. Президент Ассоциации – Д. А. Поспелов. Крупнейшие центры расположены в Москве, Петербурге, Переславле-Залесском, Новосибирске.

В рамках Ассоциации проводится большое количество исследований, собираются конференции, издается журнал. Уровень теоретических исследований по искусственному интеллекту в России ничуть не ниже мирового. К сожалению, начиная с 1975 г. на развитии этого направления сказалось прогрессирующее отставание в технологии. На данный момент отставание в области промышленных интеллектуальных систем составляет порядка 5–7 лет.

**Направления развития искусственного интеллекта.** Искусственный интеллект – это одно из направлений информатики, цель которого разработка аппаратно-программных средств, позволяющих пользователю-непрограммисту ставить и решать свои задачи, традиционно считающиеся интеллектуальными, общаясь с ЭВМ на ограниченном подмножестве естественного языка.

*Представление знаний и разработка систем, основанных на знаниях.* Это основное направление искусственного интеллекта. Оно связано с разработкой моделей представления знаний, созданием баз знаний, образующих ядро экспертных систем (ЭС). В последнее время включает в себя модели и методы извлечения и структурирования знаний и сливается с инженерией знаний.

*Игры и творчество.* Традиционно искусственный интеллект включает в себя игровые интеллектуальные задачи – шахматы, шашки, го.

В основе лежит один из ранних подходов – лабиринтная модель плюс эвристики. Сейчас это скорее коммерческое направление, так как в научном плане эти идеи считаются тупиковыми.

*Разработка естественно-языковых интерфейсов и машинный перевод.* В 50-х гг. одной из популярных тем исследований искусственного интеллекта являлась область машинного перевода. Первая программа в этой области – переводчик с английского языка на русский. Первая идея – пословный перевод, оказалась неплодотворной. В настоящее время используется более сложная модель, включающая анализ и синтез естественно-языковых сообщений, которая состоит из нескольких блоков. Для анализа – это:

- морфологический анализ – анализ слов в тексте;
- синтаксический анализ – анализ предложений, грамматики и связей между словами;
- семантический анализ – анализ смысла каждого предложения на основе некоторой предметно-ориентированной базы знаний;
- прагматический анализ – анализ смысла предложений в окружающем контексте на основе собственной базы знаний.

Синтез включает аналогичные этапы, но несколько в другом порядке.

*Распознавание образов.* Традиционное направление искусственного интеллекта, берущее начало у самых его истоков. Каждому объекту ставится в соответствие матрица признаков, по которой происходит его распознавание. Это направление близко к машинному обучению, тесно связано с нейрокибернетикой.

*Новые архитектуры компьютеров.* Это направление занимается разработкой новых аппаратных решений и архитектур, направленных на обработку символьных и логических данных. Создаются Пролог и Лисп-машины, компьютеры V и VI поколений. Последние разработки посвящены компьютерам баз данных и параллельным компьютерам.

*Интеллектуальные роботы.* Роботы – это электромеханические устройства, предназначенные для автоматизации человеческого труда.

Идея создания роботов исключительно древняя. Само слово появилось в 20-х гг. Его автор – чешский писатель Карел Чапек. Со времени создания сменилось несколько поколений роботов.

Роботы с жесткой схемой управления. Практически все современные промышленные роботы принадлежат к первому поколению. Фактически это программируемые манипуляторы.

Адаптивные роботы с сенсорными устройствами. Есть образцы таких роботов, но в промышленности они пока не используются.

Самоорганизующиеся, или интеллектуальные, роботы. Это конечная цель развития робототехники. Основная проблема при создании интеллектуальных роботов – проблема машинного зрения.

В настоящее время в мире изготавливается более 60 тыс. роботов в год.

*Специальное программное обеспечение.* В рамках этого направления разрабатываются специальные языки для решения задач невычислительного плана. Эти языки ориентированы на символьную обработку информации – LISP, PROLOG, SMALLTALK, РЕФАЛ и др. Помимо этого создаются пакеты прикладных программ, ориентированные на промышленную разработку интеллектуальных систем, или программные инструментарии искусственного интеллекта, например KEE, ARTS. Достаточно популярно создание так называемых пустых экспертных систем, или «оболочек», – EXSYS, MI и др., в которых можно наполнять базы знаний, создавая различные системы.

*Обучение и самообучение.* Активно развивающаяся область искусственного интеллекта. Включает модели, методы и алгоритмы, ориентированные на автоматическое накопление знаний на основе анализа и обобщения данных. Включает обучение по примерам (или индуктивное), а также традиционные подходы распознавания образов.

### **5.3. Экспертные системы: структура и классификация**

В Республике Беларусь современное состояние разработок в области экспертных систем можно охарактеризовать как стадию всевозрастающего интереса среди широких слоев экономистов, финансистов, преподавателей, инженеров, медиков, психологов, программистов, лингвистов. К сожалению, этот интерес имеет пока достаточно слабое материальное подкрепление – явная нехватка учебников и специальной литературы, отсутствие символьных процессоров и рабочих станций искусственного интеллекта, ограниченное финансирование исследований в этой области, слабый отечественный рынок программных продуктов для разработки экспертных систем.

Поэтому распространяются подделки под экспертные системы в виде многочисленных диалоговых систем и интерактивных пакетов прикладных программ, которые дискредитируют в глазах пользователей это чрезвычайно перспективное направление. Процесс создания экспертной системы требует участия высококвалифицированных специалистов в области искусственного интеллекта, которых пока выпускает небольшое количество высших учебных заведений республики.

Современные экспертные системы широко используются для тиражирования опыта и знаний ведущих специалистов практически во всех сферах экономики. Традиционно знания существуют в двух видах – коллективный опыт и личный опыт.

Если большая часть знаний в предметной области представлена в виде коллективного опыта (например, высшая математика), эта предметная область не нуждается в экспертных системах (рис. 5.6).

Если в предметной области большая часть знаний является личным опытом специалистов высокого уровня (экспертов), если эти знания по каким-либо причинам слабо структурированы, такая предметная область скорее всего нуждается в экспертной системе (рис. 5.7).



Рис. 5.6. Предметная область, непригодная для создания экспертной системы



Рис. 5.7. Предметная область, пригодная для создания экспертной системы

**Обобщенная структура экспертной системы.** Экспертные системы (ЭС) – это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и тиражирующие этот эмпирический опыт для консультаций менее квалифицированных пользователей.

Обобщенная структура экспертной системы представлена на рис. 5.8. Следует учесть, что реальные экспертные системы могут иметь более сложную структуру, однако блоки, изображенные на рисунке, непременно присутствуют в любой действительно экспертной системе, поскольку являют собой негласный канон на структуру современной экспертной системы.



Рис. 5.8. Структура экспертной системы

Определим основные термины.

*Пользователь* – специалист предметной области, для которого предназначена система. Обычно его квалификация недостаточно высока, и поэтому он нуждается в помощи и поддержке своей деятельности со стороны ЭС.

*Инженер по знаниям* – специалист по искусственному интеллекту, выступающий в роли промежуточного буфера между экспертом и базой знаний. Синонимы: когнитолог, инженер-интерпретатор, аналитик.

*Интерфейс пользователя* – комплекс программ, реализующих диалог пользователя с ЭС как на стадии ввода информации, так и получения результатов.

*База знаний (БЗ)* – ядро ЭС, совокупность знаний предметной области, записанная на машинный носитель в форме, понятной эксперту и пользователю (обычно на некотором языке, приближенном к естественному). Параллельно такому «человеческому» представлению существует БЗ во внутреннем «машинном» представлении.

*Решатель* – программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в БЗ. Синонимы: дедуктивная машина, блок логического вывода.

*Подсистема объяснений* – программа, позволяющая пользователю получить ответы на вопросы: «Как была получена та или иная рекомендация?» и «Почему система приняла такое решение?» Ответ на вопрос «как» – это трассировка всего процесса получения решения с указанием использованных фрагментов БЗ, т. е. всех шагов цепи умозаключений. Ответ на вопрос «почему» – ссылка на умозаключение, непосредственно предшествовавшее полученному решению, т. е. отход на один шаг назад.

*Интеллектуальный редактор БЗ* – программа, представляющая инженеру по знаниям возможность создавать БЗ в диалоговом режиме. Включает в себя систему вложенных меню, шаблонов языка представления знаний, подсказок (*help* – режим) и других сервисных средств, облегчающих работу с базой.

В коллектив разработчиков ЭС входят как минимум четыре человека:

- эксперт;
- инженер по знаниям;
- программист;
- пользователь.

Возглавляет коллектив инженер по знаниям, это ключевая фигура при разработке систем, основанных на знаниях.

### **Классификация экспертных систем.**

Класс «экспертные системы» в настоящее время объединяет несколько тысяч различных программных комплексов, которые можно классифицировать по различным критериям. Полезными могут оказаться следующие классификации (рис. 5.9).

#### *Классификация по решаемой задаче.*

*Интерпретация данных.* Это одна из традиционных задач для экспертных систем. Под интерпретацией понимается определение смысла данных, результаты которого должны быть согласованными и корректными. Обычно предусматривается многовариантный анализ данных.





Рис. 5.9. Классификация экспертных систем

*Диагностика.* Под диагностикой понимается обнаружение неисправности в некоторой системе. Неисправность – это отклонение от нормы. Такая трактовка позволяет с единых теоретических позиций рассматривать и неисправность оборудования в технических системах, и заболевания живых организмов, и всевозможные природные аномалии. Важной спецификой является необходимость понимания функциональной структуры («анатомии») диагностирующей системы.

*Проектирование.* Проектирование состоит в подготовке спецификаций на создание «объектов» с заранее определенными свойствами. Под спецификацией понимается весь набор необходимых документов – чертеж, пояснительная записка и т. д. Основные проблемы здесь – получение четкого структурного описания знаний об объекте и проблема «следа». Для организации эффективного проектирования и, в еще большей степени, перепроектирования необходимо формировать не только сами проектные решения, но и мотивы их принятия. Таким образом, в задачах проектирования тесно связываются два основных процесса, выполняемых в рамках соответствующей ЭС: процесс вывода решения и процесс объяснения.

*Обучение.* Системы обучения диагностируют ошибки при изучении какой-либо дисциплины с помощью ЭВМ и подсказывают правильные решения. Они аккумулируют знания о гипотетическом «ученике» и его характерных ошибках, затем в работе способны диагностировать сла-

бые места в знаниях обучаемых и находить соответствующие средства для их ликвидации. Кроме того, они планируют акт общения с учеником в зависимости от успехов ученика с целью передачи знаний.

*Классификация по связи с реальным временем.*

*Статические* ЭС разрабатываются в предметных областях, в которых база знаний и интерпретируемые данные не меняются во времени. Они стабильны.

*Квазидинамические* ЭС интерпретируют ситуацию, которая меняется с некоторым фиксированным интервалом времени.

*Классификация по степени интеграции с другими программами:*

*Автономные* ЭС работают непосредственно в режиме консультаций с пользователем для специфических экспертных задач, для решения которых не требуется привлекать традиционные методы обработки данных (расчеты, моделирование и т. д.).

*Гибридные* ЭС представляют программный комплекс, агрегирующий стандартные пакеты прикладных программ (например, математическую статистику, линейное программирование или системы управления базами данных) и средства манипулирования знаниями. Это может быть интеллектуальная надстройка над ППП или интегрированная среда для решения сложной задачи с элементами экспертных знаний.

Несмотря на внешнюю привлекательность гибридного подхода, следует отметить, что разработка таких систем являет собой задачу, на порядок более сложную, чем разработка автономной ЭС. Стыковка не просто разных пакетов, а разных методологий (что происходит в гибридных системах) порождает целый комплекс теоретических и практических трудностей.

*Инструментальные средства построения экспертных систем.*

*Традиционные языки программирования.* В эту группу инструментальных средств входят традиционные языки программирования (C, C++, Basic, SmallTalk, Fortran и т. д.), ориентированные в основном на численные алгоритмы и слабо подходящие для работы с символьными и логическими данными. Поэтому создание систем искусственного интеллекта на основе этих языков требует большой работы программистов. Однако большим достоинством этих языков является высокая эффективность, связанная с их близостью к традиционной машинной архитектуре. Кроме того, использование традиционных языков программирования позволяет включать интеллектуальные подсистемы (например, интегрированные экспертные системы) в крупные программные комплексы общего назначения. Среди традиционных

языков наиболее удобными считаются объектно-ориентированные (SmallTalk, C++). Это связано с тем, что парадигма объектно-ориентированного программирования тесно связана с фреймовой моделью представления знаний. Кроме того, традиционные языки программирования используются для создания других классов инструментальных средств искусственного интеллекта.

*Языки искусственного интеллекта.* Это прежде всего Лисп (LISP) и Пролог (Prolog) – наиболее распространенные языки, предназначенные для решения задач искусственного интеллекта. Есть и менее распространенные языки искусственного интеллекта, например РЕФАЛ, разработанный в России. Универсальность этих языков меньшая, нежели традиционных языков, но ее потерю языки искусственного интеллекта компенсируют богатыми возможностями по работе с символическими и логическими данными, что крайне важно для задач искусственного интеллекта. На основе языков искусственного интеллекта создаются специализированные компьютеры (например, Лисп-машины), предназначенные для решения задач искусственного интеллекта. Недостаток этих языков – неприменимость для создания гибридных экспертных систем.

*Специальный программный инструментарий.* В эту группу программных средств искусственного интеллекта входят специальные инструментариумы общего назначения. Как правило, это библиотеки и надстройки над языком искусственного интеллекта Лисп: KEE (Knowledge Engineering Environment), FRL (Frame Representation Language), KRL (Knowledge Representation Language), ARTS и др., позволяющие пользователям работать с заготовками экспертных систем на более высоком уровне, нежели это возможно в обычных языках искусственного интеллекта.

*Оболочки.* Под «оболочками» (shells) понимают «пустые» версии существующих экспертных систем, т. е. готовые экспертные системы без базы знаний. Примером такой оболочки может служить EMYCIN (Empty MYCIN – пустой MYCIN), которая представляет собой незаполненную экспертную систему MYCIN. Достоинство оболочек в том, что они вообще не требуют работы программистов для создания готовой экспертной системы. Требуется только специалист(ы) в предметной области для заполнения базы знаний. Однако если некоторая предметная область плохо укладывается в модель, используемую в некоторой оболочке, заполнить базу знаний в этом случае весьма непросто.

### Технология разработки экспертных систем. Этапы разработки.

Разработка программных комплексов экспертных систем как за рубежом, так и в нашей стране находится на уровне скорее искусства, чем науки. Это связано с тем, что долгое время системы искусственного интеллекта внедрялись в основном во время фазы проектирования, а чаще всего разрабатывалось несколько прототипных версий программ, прежде чем был получен конечный продукт. Такой подход действует хорошо в исследовательских условиях, однако в коммерческих условиях он является слишком дорогим, чтобы оправдать коммерчески жизненный продукт.

Процесс разработки промышленной экспертной системы, опираясь на традиционные технологии, можно разделить на шесть более или менее независимых от предметной области этапов (рис. 5.10).

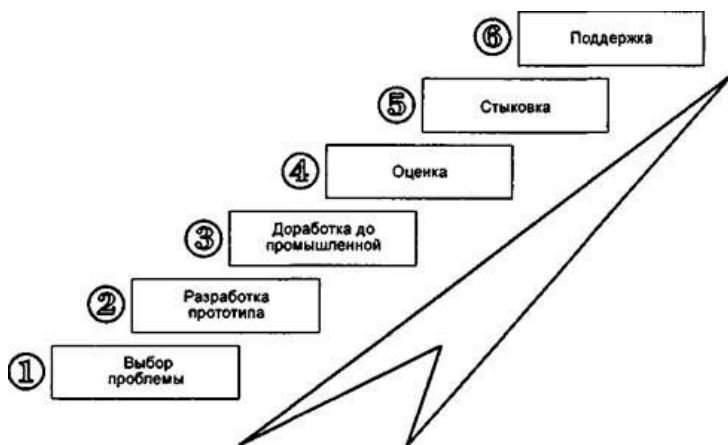


Рис. 5.10. Этапы разработки ЭС

Последовательность этапов дана для общего представления о создании идеального проекта. Конечно, последовательность эта не вполне фиксированная. В действительности каждый последующий этап разработки ЭС приносит новые идеи, которые могут повлиять на предыдущие решения и даже привести к их переработке.

За разработку экспертных систем целесообразно браться организации, где накоплен опыт по автоматизации рутинных процедур обработки информации, например:

- информационный поиск;

- сложные расчеты;
- графика;
- обработка текстов.

Решение таких задач, во-первых, подготавливает высококвалифицированных специалистов по информатике, необходимых для создания интеллектуальных систем, во-вторых, позволяет отделить от экспертных систем неэкспертные задачи.

*Этап 1: выбор подходящей проблемы.* Он включает:

- определение проблемной области и задачи;
- нахождение эксперта, желающего сотрудничать при решении проблемы, и назначение коллектива разработчиков;
- определение предварительного подхода к решению проблемы;
- анализ расходов и прибыли от разработки;
- подготовку подробного плана разработки.

При выборе области применения следует учитывать, что если знание, необходимое для решения задач, постоянное, четко формулируемое и связано с вычислительной обработкой, то обычные алгоритмические программы, по всей вероятности, будут самым целесообразным способом решения проблем в этой области.

Экспертная система ни в коем случае не устранит потребность в реляционных базах данных, статистическом программном обеспечении, электронных таблицах и системах текстовой обработки. Но если результативность задачи зависит от знания, которое является субъективным, изменяющимся, символьным или вытекающим частично из соображений здравого смысла, тогда область может обоснованно выступать претендентом на экспертную систему.

Приведем некоторые факты, свидетельствующие о необходимости разработки и внедрения экспертных систем:

- нехватка специалистов, тратящих много времени для оказания помощи другим;
- потребность в многочисленном коллективе специалистов, поскольку ни один из них не обладает достаточными знаниями;
- сниженная производительность, поскольку задача требует полного анализа сложного набора условий, а обычный специалист не в состоянии просмотреть (за отведенное время) все эти условия;
- большое расхождение между решениями самых хороших и самых плохих исполнителей;
- наличие конкурентов, имеющих преимущество в том, что они лучше справляются с поставленной задачей.

Данные задачи имеют следующие характеристики:

- являются узкоспециализированными;
- не зависят в значительной степени от общечеловеческих знаний или соображений здравого смысла;
- не являются для эксперта ни слишком легкими, ни слишком сложными (время, необходимое эксперту для решения проблемы, может составлять от трех часов до трех недель);
- условия исполнения задачи определяются самим пользователем системы;
- имеют результаты, которые можно оценить.

*Этап 2: разработка прототипной системы.* Прототипная система является усеченной версией экспертной системы, спроектированной для проверки правильности кодирования фактов, связей и стратегий рассуждения эксперта. Она также дает возможность инженеру по знаниям привлечь эксперта к активному участию в разработке экспертной системы и, следовательно, к принятию им обязательства приложить все усилия для создания системы в полном объеме.

Объем прототипа – несколько десятков правил, фреймов или примеров. На рис. 5.11 изображены шесть стадий разработки прототипа и минимальный коллектив разработчиков, занятых на каждой из стадий.



Рис. 5.11. Стадии разработки прототипа ЭС

*Этап 3: развитие прототипа до промышленной ЭС.* При неудовлетворительном функционировании прототипа эксперт и инженер по знаниям имеют возможность оценить, что именно будет включено в разработку окончательного варианта системы.

Если первоначально выбранные объекты или свойства оказываются неподходящими, их необходимо изменить. Можно сделать оценку общего числа эвристических правил, необходимых для создания окончательного варианта экспертной системы. Иногда при разработке промышленной системы выделяют дополнительные этапы для перехода: демонстрационный прототип – исследовательский прототип – действующий прототип – промышленная система.

Однако чаще реализуется плавный переход от демонстрационного прототипа к промышленной системе, при этом, если программный инструментариум выбран удачно, необязательна перепись другими программными средствами.

Основное на третьем этапе заключается в добавлении большого числа дополнительных эвристик. Эти эвристики обычно увеличивают глубину системы, обеспечивая большее число правил для трудноуловимых аспектов отдельных случаев. В то же время эксперт и инженер по знаниям могут расширить охват системы, включая правила, управляющие дополнительными подзадачами или дополнительными аспектами экспертной задачи (метазнания).

После установления основной структуры ЭС инженер по знаниям приступает к разработке и адаптации интерфейсов, с помощью которых система будет общаться с пользователем и экспертом. Необходимо обратить особое внимание на языковые возможности интерфейсов, их простоту и удобство для управления работой ЭС. Система должна обеспечивать пользователю возможность легким и естественным образом спрашивать непонятное, приостанавливать работу и т. д. В частности, могут оказаться полезными графические представления.

На этом этапе разработки большинство экспертов узнают достаточно о вводе правил и могут сами вводить в систему новые правила. Таким образом начинается процесс, во время которого инженер по знаниям передает право собственности и контроля за системой эксперту для уточнения, детальной разработки и обслуживания.

*Этап 4: оценка системы.* После завершения этапа разработки промышленной экспертной системы необходимо провести ее тестирование в отношении критериев эффективности. Экспертные системы оцениваются главным образом для того, чтобы проверить точность работы программы и ее полезность.

*Этап 5: стыковка системы.* На этом этапе осуществляется стыковка экспертной системы с другими программными средствами в среде, в которой она будет работать, и обучение людей, которых она будет обслуживать.

Когда экспертная система уже готова, инженер по знаниям должен убедиться в том, что эксперты, пользователи и персонал знают, как эксплуатировать и обслуживать ее. После передачи им своего опыта в области информационной технологии инженер по знаниям может полностью предоставить ее в распоряжение пользователей.

*Этап 6: поддержка системы.* При перекодировании системы на язык, подобный Си, повышается ее быстродействие и увеличивается переносимость, однако гибкость при этом уменьшается. Это приемлемо лишь в том случае, если система сохраняет все знания проблемной области и это знание не будет изменяться в ближайшем будущем. Однако, если экспертная система создана именно из-за того, что проблемная область изменяется, то необходимо поддерживать систему в инструментальной среде разработки.

### **Контрольные вопросы**

1. Что такое корпоративная информационная система?
2. Отличие корпоративной информационной системы от информационных систем.
3. Опишите структуру корпоративной информационной системы.
4. Перечислите требования, предъявляемые к КИС.
5. Какие международные стандарты известны в области КИС?
6. Что такое знания?
7. Чем отличаются знания от данных?
8. Перечислите основные классы моделей знаний.
9. Опишите обобщенную структуру экспертной системы.
10. Представьте классификацию экспертных систем.

### **ЛИТЕРАТУРА**

1. Корпоративные информационные системы: пособие / Л. К. Голенда [и др.]; под общ. ред. Л. К. Голенда, Н. Н. Говядиновой. – Минск: БГЭУ, 2011. – 291 с.
2. Олейник, П. П. Корпоративные информационные системы: учебник для вузов. Стандарт третьего поколения / П. П. Олейник. – СПб.: Питер, 2012. – 176 с.
3. Матяш, С. А. Корпоративные информационные системы: учеб. пособие / С. А. Матяш. – Москва, Берлин: Директ-Медиа, 2015. – 471 с.
4. Гантц, И. С. Корпоративные информационные системы: учеб. пособие / И. С. Гантц. – Москва: ФГБОУ ВО «МИРЭА – Российский технологический университет», 2021. – 68 с.
5. Борисов, В. В. Экспертные системы: учеб. пособие по направлению «Информатика и вычислительная техника» [Текст] / В. В. Борисов, А. В. Бобряков, А. Е. Мисник. – Смоленск: Универсум, 2021. – 110 с.
6. Герман, О. В. Экспертные системы: учеб.-метод. пособие / О. В. Герман. – Минск: БГУИР, 2008. – 91 с.



## СОДЕРЖАНИЕ

|                                                                                                            |     |
|------------------------------------------------------------------------------------------------------------|-----|
| ВВЕДЕНИЕ.....                                                                                              | 3   |
| Тема 1. ОСНОВНЫЕ ПОНЯТИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ.....                                                    | 4   |
| 1.1. Информатизация. Информационное общество.....                                                          | 4   |
| 1.2. Понятия информации и информационной системы. Информационные процессы. Информационные технологии ..... | 7   |
| Контрольные вопросы .....                                                                                  | 16  |
| Литература.....                                                                                            | 16  |
| Тема 2. ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ .....                                                           | 17  |
| 2.1. Классификация программного обеспечения .....                                                          | 17  |
| 2.2. Системы обработки текстовых документов.....                                                           | 26  |
| 2.3. Табличные процессоры .....                                                                            | 40  |
| 2.4. Системы подготовки презентаций.....                                                                   | 115 |
| Контрольные вопросы .....                                                                                  | 120 |
| Литература.....                                                                                            | 123 |
| Тема 3. СЕТЕВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ.....                                                             | 124 |
| 3.1. Локальные сети .....                                                                                  | 124 |
| 3.2. Глобальная сеть Интернет.....                                                                         | 131 |
| 3.3. Основы проектирования web-страниц.....                                                                | 143 |
| Контрольные вопросы .....                                                                                  | 151 |
| Литература.....                                                                                            | 151 |
| Тема 4. ТЕХНОЛОГИИ БАЗ ДАННЫХ .....                                                                        | 152 |
| 4.1. Базы данных: понятие, назначение и архитектура .....                                                  | 152 |
| 4.2. Модели данных .....                                                                                   | 154 |
| 4.3. Системы управления базами данных (СУБД). Обзор современных СУБД и направления их развития.....        | 164 |
| 4.4. Функциональные возможности СУБД на примере MS Access.....                                             | 171 |
| Контрольные вопросы .....                                                                                  | 204 |
| Литература.....                                                                                            | 205 |
| Тема 5. ИНФОРМАЦИОННЫЕ СИСТЕМЫ.....                                                                        | 206 |
| 5.1. Корпоративные информационные системы.....                                                             | 206 |
| 5.2. Системы искусственного интеллекта .....                                                               | 225 |
| 5.3. Экспертные системы: структура и классификация .....                                                   | 237 |
| Контрольные вопросы .....                                                                                  | 248 |
| Литература.....                                                                                            | 248 |

Учебное издание

**Азаренко** Ольга Николаевна  
**Ковалевская** Лилия Ивановна  
**Ракутин** Вячеслав Геннадьевич и др.

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

КУРС ЛЕКЦИЙ

Учебно-методическое пособие

Редактор *С. Н. Кириленко*  
Технический редактор *Н. Л. Якубовская*

Подписано в печать 16.10.2023. Формат 60×84 <sup>1</sup>/<sub>16</sub>. Бумага офсетная.  
Ризография. Гарнитура «Таймс». Усл. печ. л. 14,65. Уч.-изд. л. 12,16.  
Тираж 40 экз. Заказ .

УО «Белорусская государственная сельскохозяйственная академия».  
Свидетельство о ГРИРПИ № 1/52 от 09.10.2013.  
Ул. Мичурина, 13, 213407, г. Горки.

Опечатано в УО «Белорусская государственная сельскохозяйственная академия».  
Ул. Мичурина, 5, 213407, г. Горки.