

Г.Н.ФЕДОРОВА

# ИНФОРМАЦИОННЫЕ СИСТЕМЫ

**Учебник**

*Рекомендовано  
Федеральным государственным учреждением  
«Федеральный институт развития образования»  
в качестве учебника для использования  
в учебном процессе образовательных учреждений,  
реализующих программы среднего профессионального образования  
Регистрационный номер рецензии 323 от 16 июня 2009 г. ФГУ «ФИРО»*

3-е издание, стереотипное



Москва  
Издательский центр «Академия»  
2013

УДК 681.518(075.32)  
ББК 65.39я723  
Ф333

Рецензент —  
зав. лабораторией Центра компьютерного обучения Московского  
автомобилестроительного колледжа при Академии Народного Хозяйства РФ  
*А. А. Соломашкин*

**Федорова Г. Н.**

**Ф333** Информационные системы: учебник для студ. учреждений  
сред. проф. образования / Г. Н. Федорова. — 3-е изд., стер. —  
М. : Издательский центр «Академия», 2013. — 208 с.  
ISBN 978-5-7695-9642-1

Подробно рассмотрены основные направления развития информационных систем, их информационные ресурсы и технологии. Раскрыты положения и методологические принципы современных информационных систем управления. Проанализированы принципы построения интегрированных корпоративных информационных систем. Определены понятия жизненного цикла информационной системы и сопровождающих его процессов. Рассмотрены методы оценки эффективности автоматизированных информационных систем.

Учебник может быть использован при освоении общепрофессиональной дисциплины ОП.05 «Устройство и функционирование информационной системы», а также при освоении профессионального модуля ПМ. 02 «Участие в разработке информационных систем (МДК 02.01)» по специальности 230401 «Информационные системы (по отраслям)» укрупненной группы специальностей 230000 «Информатика и вычислительная техника»

Для студентов учреждений среднего профессионального образования.

УДК 681.518(075.32)  
ББК 65.39я723

*Оригинал-макет данного издания является собственностью Издательского центра «Академия», и его воспроизведение любым способом без согласия правообладателя запрещается*

ISBN 978-5-7695-9642-1

© Федорова Г. Н., 2010  
© Образовательно-издательский центр «Академия», 2010  
© Оформление. Издательский центр «Академия», 2010

## ПРЕДИСЛОВИЕ

В настоящее время остро встает вопрос о необходимости подготовки специалистов, имеющих соответствующие знания и навыки в области разработки и внедрения информационных систем на предприятиях различных сфер деятельности. Следовательно, возникает потребность в учебной литературе, соответствующей уровню современных требований, для того чтобы в достаточной мере осветить вопросы, связанные с созданием и эксплуатацией информационных систем. Учебные издания по информатизации производства и управления теряют свою актуальность в короткие сроки, так как стремительно меняются и информационные технологии, используемые на предприятиях, и экономические условия, в которых они работают. При этом подготовка в среднем специальном учебном заведении должна предусматривать овладение будущими специалистами самыми новейшими знаниями в области разработки и внедрения информационных систем и умением активно использовать эти знания в своей профессиональной деятельности.

Современный выпускник среднего профессионального учебного заведения должен знать процессы и стадии жизненного цикла информационных систем, их типовые компоненты и классификацию, требования к надежности и эффективности, технологию проектирования информационных систем, методы их разработки, эксплуатации и сопровождения.

В учебнике рассмотрены классификация и структура информационных систем, информационные ресурсы, технологии и связанные с ними понятия и определения. Описаны характеристики и базовые функции экспертных систем. Достаточно подробно раскрыты положения и методологические принципы современных информационных систем управления, а также изложена стратегия их развития. Проанализированы принципы построения интегрированных корпоративных информационных систем (ИКИС), дан сравнительный аналитический обзор внедренных ИКИС на отечественных и зарубежных предприятиях. Раскрыты понятия жизненного цикла информационной системы и сопровождающих его процессов в соответствии со стандартом ISO/IEC 12207, представлены различные методологии построения

моделей информационных систем, рассмотрены методы оценки экономической эффективности. Каждая глава учебника включает контрольные вопросы для самостоятельной работы, позволяющие студенту закрепить изученный материал.

Учебник может быть использован студентами при изучении дисциплин, связанных с информационными технологиями и управлением. Предлагаемый материал может стать полезным также для широкого круга специалистов, занимающихся внедрением информационных систем на предприятиях. Содержание данной книги предполагает, что читатель освоил курсы информатики и информационных технологий, владеет теорией баз данных и имеет представление об экономике и управлении предприятием.

# ОБЩАЯ ХАРАКТЕРИСТИКА АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

## 1.1. Состав и структура информационных систем

### 1.1.1. Основные понятия информационных систем

В общем смысле под системой понимают совокупность объектов, компонентов или элементов произвольной природы, образующих некоторую целостность (система образования, пищеварительная система, солнечная система и т. п.). Определяющей предпосылкой выделения некоторой совокупности как системы является возникновение у нее новых свойств, которых не имеют составляющие ее элементы. Системы значительно различаются между собой как по составу, так и по целям функционирования.

Наиболее общей моделью системы считают так называемый «черный ящик». В этом случае система имеет вид прямоугольника, внутреннее устройство которого скрыто от аналитика или неизвестно. Однако система не является полностью изолированной от внешней среды, поскольку последняя оказывает на систему некоторые информационные или материальные воздействия. Такие воздействия получили название входных воздействий. В свою очередь, система также оказывает на среду или другие системы определенные информационные или материальные воздействия, которые получили название выходных воздействий. Графическое изображение данной модели приведено на рис. 1.1.

Добавление к понятию «система» слова «информационная» отражает цель ее создания и функционирования. Назначение информационной системы — своевременное формирование и выдача достоверной информации для принятия решений. Информационные системы обеспечивают сбор, хранение, обработку, поиск, выдачу информации в задачах из любой области. Они помогают анализировать информацию, решать проблемы и создавать новые информационные продукты.

**И н ф о р м а ц и я** — это сведения об объектах, явлениях, процессах, событиях окружающего мира, уменьшающие неопределен-

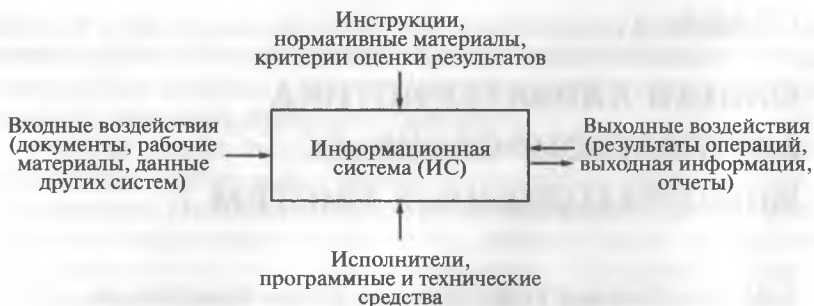


Рис. 1.1. Модель информационной системы в виде «черного ящика»

ность знаний о них. Эти знания отражают действительность в сознании человека. Информация должна быть полной, достоверной, своевременной, непротиворечивой, адекватной.

Разные авторы придают различные оттенки определению информационной системы, расширяя или сужая его смысл. В настоящее время информационные системы часто связывают с понятием автоматизации и называют автоматизированными информационными системами. Автоматизацией является процесс внедрения вычислительной техники в различные сферы информационной деятельности. Суть этого процесса состоит в том, что часть функций, выполняемых людьми при управлении производственными, административными, социальными и другими аспектами функционирования предприятия, организации или отрасли, передается вычислительной технике.

Необходимо понимать разницу между понятиями «информационная технология» и «информационная система».

Информационная технология — это приемы, способы и методы применения средств вычислительной техники при выполнении функций сбора, хранения, обработки и использования данных. Информационная технология является более емким понятием, чем информационная система. Информационная технология может существовать вне информационной системы, тогда как информационная система невозможна без применения конкретной информационной технологии.

К информационным системам относятся:

- информационно-справочные и информационно-поисковые системы;
- системы, обеспечивающие автоматизацию документооборота и учета (в том числе бухгалтерского);
- информационные системы управления;
- интеллектуальные (экспертные) системы;
- системы автоматизации научных исследований;

- системы автоматизированного проектирования;
- геоинформационные системы и др.

Процессы, обеспечивающие работу информационной системы любого назначения, условно можно представить в виде схемы, состоящей из следующих блоков (рис. 1.2):

- ввод информации (сбор информации о состоянии внешней среды и объекта управления, т. е. создание первичной, или входной, информации и представление ее в нужном формате);
- база данных (хранилище данных);
- обработка информации (поиск, фильтрация, сортировка, агрегирование, анализ, вывод информации для представления потребителям или передачи в другую систему);
- обратная связь (передача информации, переработанной потребителем для коррекции входной информации, т. е. выработка управляющих воздействий).

Любая система состоит из подсистем, подсистема любой системы может быть сама рассмотрена как система. Границы рассматриваемой системы определяются доступными ресурсами и окружением. Например, система бухгалтерского учета является частью (подсистемой) системы управления производственным предприятием и включает в себя подсистемы материального учета, налогового учета, расчета заработной платы и т. д.

Структура информационной системы должна быть такой, чтобы взаимодействие между ее подсистемами укладывалось в ограниченные стандартные рамки:

- каждая подсистема должна инкапсулировать свое содержимое (скрывать его от других подсистем);
- каждая подсистема должна иметь четкий интерфейс с другими подсистемами.

Инкапсуляция позволяет рассматривать структуру каждой подсистемы независимо от других подсистем. Наличие интерфейсов между подсистемами позволяет объединять их в систему более высокого уровня.

Система называется большой, если ее исследование или моделирование затруднено из-за большой размерности, т. е. множество состояний системы имеет большую размерность. Большая система сводится к системе меньшей размерности с использованием более



Рис. 1.2. Процессы в информационной системе

мощных вычислительных средств или ресурсов либо, если это возможно, разбиением задачи на ряд задач меньшей размерности.

Система называется сложной, если в ней не хватает информационных ресурсов для эффективного описания и управления системой, т.е. определения управляющих параметров или для принятия решений.

Для изучения информационных систем полезно иметь представление о задачах системного анализа. Системный анализ — это методология решения проблем, основанная на структуризации систем (социальных, экономических, технических и т.д.). Другими словами, системный анализ — это логически связанная совокупность теоретических и эмпирических положений из области математики, естественных наук и области разработки сложных систем, обеспечивающая повышение обоснованности решения конкретной проблемы.

В состав задач системного анализа в процессе создания информационной системы входят задачи декомпозиции, анализа и синтеза.

*Задача декомпозиции* означает представление системы в виде набора подсистем, состоящих из более мелких элементов.

*Задача анализа* состоит в нахождении различного рода свойств системы или среды, окружающей систему. Цель анализа — определение закона преобразования информации, задающего поведение системы.

*Задача синтеза* системы противоположна задаче анализа. Необходимо по описанию закона преобразования информации построить систему, фактически выполняющую это преобразование по определенному алгоритму.

## 1.1.2. История развития ИС

Идея информационных систем и некоторые принципы их организации возникли задолго до появления персональных компьютеров. Библиотеки, архивы, адресные бюро, телефонные справочники, словари — все это информационные системы. Однако именно компьютеризация придала информационным системам их нынешний облик, на несколько порядков повысила эффективность и расширила сферу их применения. ИС представляют собой системы, основанные на постоянно развивающихся концепциях использования информации.

Первые автоматизированные информационные системы появились в 50-х годах прошлого столетия. В эти послевоенные годы сложилась ситуация, когда меры по созданию новых организационных структур или совершенствованию экономических механизмов



регулирования не позволяли справиться с нарастанием трудностей в управлении экономикой и соответствующим ростом потерь. Принципиальное решение стало возможным только за счет кардинального роста производительности труда в сфере обработки информации. Тогда же начали разрабатывать автоматизированные системы — они были предназначены для выписки счетов, накладных, составления платежных ведомостей, расчета зарплаты и других операций бухгалтерского учета, а реализовывались на электромеханических бухгалтерских счетных машинах. Это привело к некоторому уменьшению затрат и сокращению времени на подготовку бумажных документов. В то время был распространен термин — «электронная обработка данных» (ЭОД).

В 1960-е годы средства вычислительной техники получают дальнейшее развитие: появляются операционные системы, дисковая технология, значительно улучшаются языки программирования. Развитие вычислительной техники обусловило появление новых возможностей в автоматизации различных видов деятельности, например, подготовки отчетной документации. Изменяется отношение к информационным системам. Информацию, полученную с их помощью, стали применять для периодической отчетности по многим параметрам. Были созданы системы управленческих отчетов, ориентированные на менеджеров, принимающих решения.

В 1970-е годы развиваются технология баз данных и средства для интерактивной обработки данных. Развитие новых информационных технологий создает условия для появления систем поддержки принятия решений (СППР). В отличие от систем управленческих отчетов, которые предоставляют информацию по заранее установленным формам отчетности, СППР предоставляют ее по мере необходимости.

В 1970 — 1980-х гг. в офисах начали применять разнообразные компьютерные и телекоммуникационные технологии, которые расширили область применения информационных систем. ИС начинают широко использоваться в качестве средства управленческого контроля, поддерживающего и ускоряющего процесс принятия решений. 1980-е годы характеризуются еще и тем, что информационные технологии начали претендовать на новую роль в организации: компании открыли для себя, что информационные системы являются стратегическим оружием. Информационные системы этого периода, предоставляя вовремя нужную информацию, помогают организации достичь успеха в своей деятельности, создавать новые товары и услуги, находить новые рынки сбыта, обеспечивать сотрудничество с достойными партнерами, организовывать выпуск продукции по низкой цене и многое другое.

Развитие информационных систем управления в России и на Западе проходило разными путями ввиду существенной разницы исторических условий. Плановая социалистическая система жестко регламентировала и распределяла человеческие, финансовые, материальные ресурсы, поэтому нерыночная экономика практически не стимулировала руководителей предприятий экономить ресурсы и оптимизировать финансовые и материальные потоки. Совершенно иная картина была присуща капиталистическому обществу. В парадигме капитализма для настоящего хозяина задача рационального управления ресурсами всегда была первоочередной. В силу этих объективных обстоятельств информатизация управления на Западе начиналась с решения задач управления запасами. В нашей же стране внедрение информационных систем начиналось с бухгалтерских и кадровых подразделений предприятий.

Первоначально отечественные информационные системы выполняли функции учета, затем зона их ответственности была распространена на функции управления и планирования. Так, наиболее востребованные на рынке продукты компании «IC» появлялись на свет в следующей последовательности: «Бухгалтерия», «Зарплата и Кадры», «Торговля и Склад», «Предприятие». Аналогичные этапы развития прошли и программные продукты других отечественных фирм: «БЭСТ», «Парус», «Галактика». Предприятия, убедившись в выгодности применения автоматизированных информационных систем, переходили к автоматизации всех функций управления. При этом зачастую использовали программные продукты разных фирм, базирующиеся на различных программно-аппаратных платформах. До сих пор на многих предприятиях, пытающихся автоматизировать управление своими бизнес-процессами, можно встретить большое разнообразие программно-аппаратных средств. В результате «лоскутной» автоматизации подразделения фирмы работают автономно и иногда даже менее эффективно, чем вообще без автоматизации.

В основе западных информационных систем с самого начала их развития лежали идеи сбережения ресурсов, т.е. оптимизации потоков материальных и финансовых средств. Это нашло отражение в названиях информационных систем: IC (Inventory Control — управление запасами), MRP (Material Requirements Planning — планирование потребности в материалах), MRP II (Manufacturing Resource Planning — планирование производственных ресурсов), ERP (Enterprise Resource Planning — планирование ресурсов корпорации). Это, конечно же, не означает, что разработчики западных систем не уделяли достаточного внимания модулю бухгалтерского учета. Однако задачи учета тесно увязывались с управленческими задачами.

Сегодня в нашей стране деловая среда стремительно меняется: расширяются внешние и внутренние связи компаний, увеличивается скорость бизнес-процессов. Требования к информационным технологиям повышаются, что способствует быстрому развитию систем управления. В итоге информационные системы становятся одним из важнейших инструментов управления, одновременно порождая новые бизнес-модели, требующие новых подходов к автоматизации.

### **1.1.3. Функциональная часть ИС**

Структура любой информационной системы может быть представлена совокупностью функциональных подсистем, реализующих решение информационных задач предметной области. Набор подсистем зависит от специфики предметной области и цели, для которой предназначена информационная система. Ранее определили, что общей функцией всех информационных систем является сбор хранение, обработка и выдача информации нового качества в той или иной форме. При разбиении (декомпозиции) общей цели системы на подцели получаем декомпозицию системы на функциональные подсистемы. Вообще говоря, такая декомпозиция даже для информационной системы одного типа не является однозначной. Набор функциональных подсистем, входящих в состав информационной системы, зависит от многих параметров — структуры системы, сферы ее функциональной направленности, уровней управленческой деятельности и т. д.

Для автоматизированных систем научных исследований (АСНИ), например, в функциональную часть могут входить подсистемы: планирования и подготовки проведения экспериментов; обработки результатов экспериментов и др.

Для автоматизированных обучающих систем (АОС) — это подсистемы:

- проведения учебных занятий;
- тестирования учащихся;
- регистрации и обработки результатов обучения и т. д.

Для систем автоматизированного проектирования (САПР) это могут быть подсистемы:

- функционально-логического и конструкторского проектирования;
  - подсистема параметрической оптимизации;
  - подсистема конструкторско-технологической документации
- и т. д.

Для информационной системы управления предприятием (ИСУ) к таким функциональным подсистемам можно отнести подсистемы:

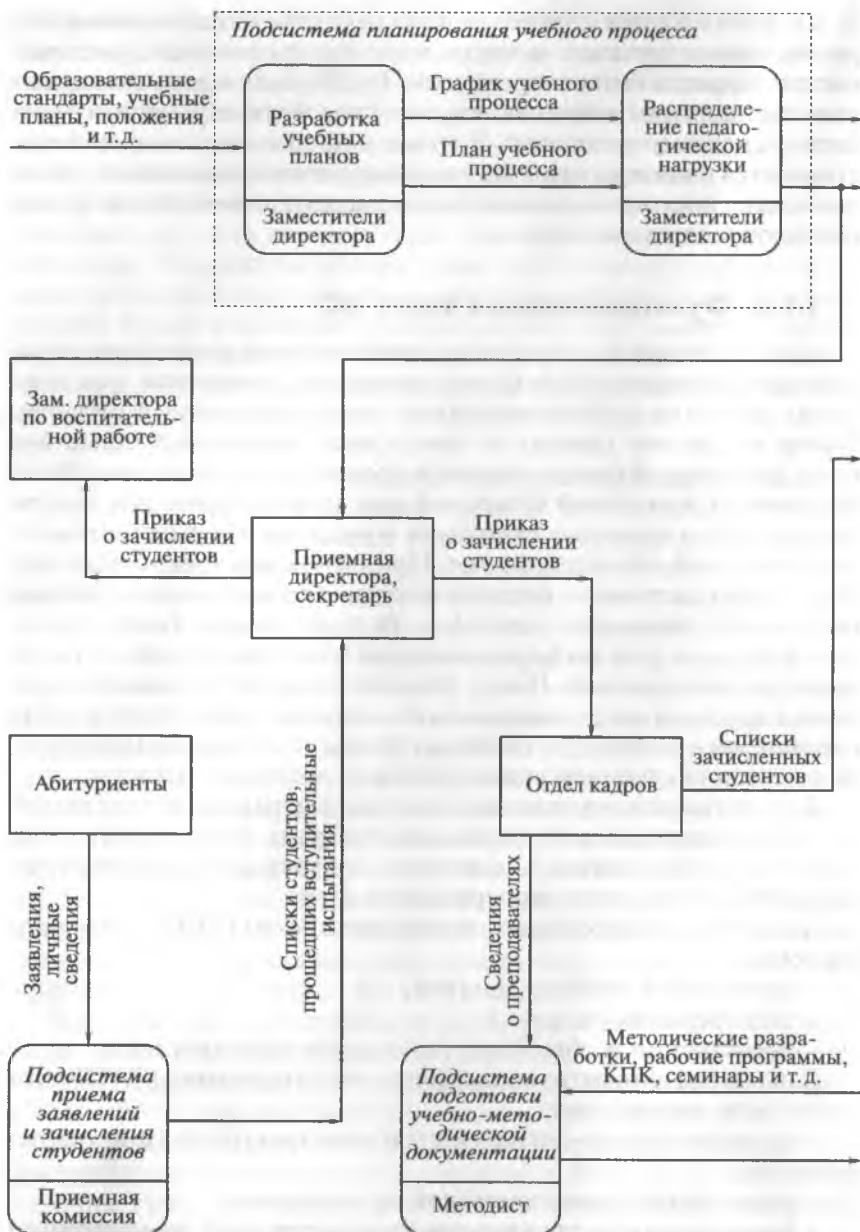
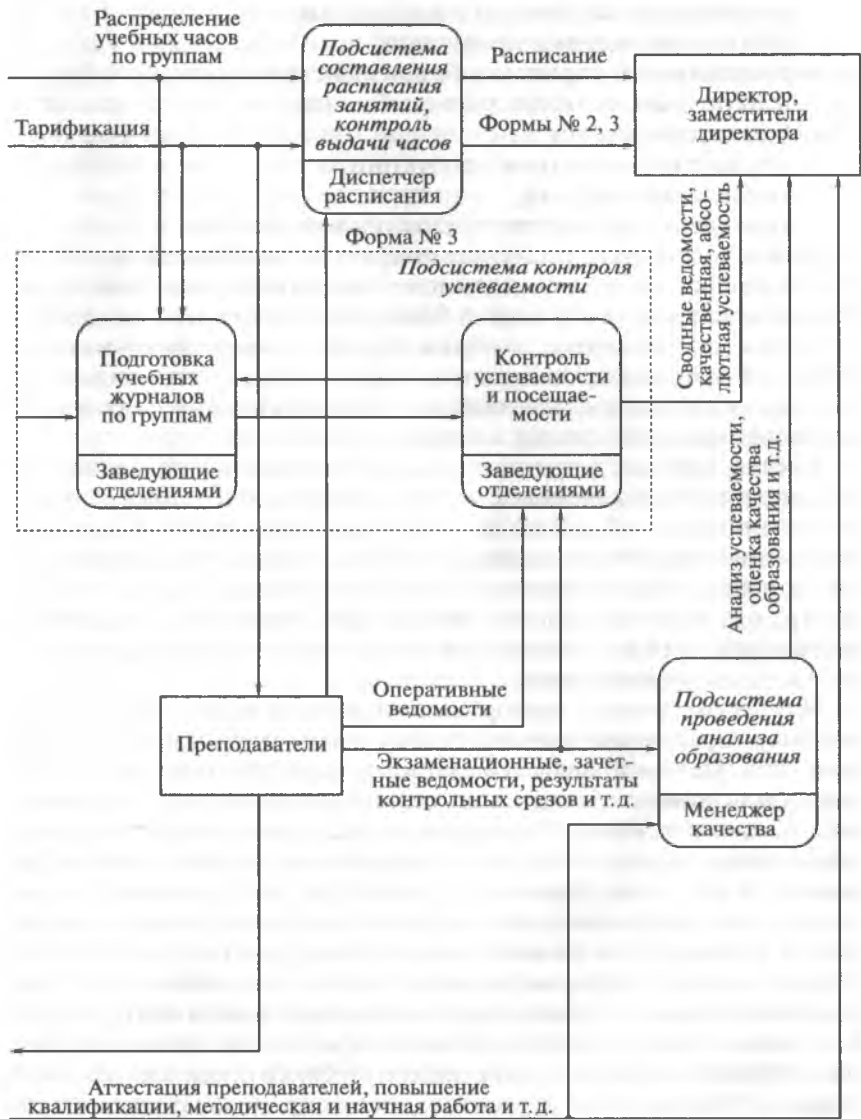


Рис. 1.3. Информационная система управления учебным заведением СПО



- технико-экономического планирования;
- технической подготовки производства;
- сбыта и реализации продукции;
- оперативного управления производством;
- материально-технического снабжения;
- бухгалтерского учета;
- управления качеством продукции;
- управления кадрами;
- управления вспомогательным производством.

Каждая из функциональных подсистем информационной системы решает некоторое подмножество функциональных задач. Решение каждой такой задачи базируется на некотором математическом обеспечении, отображающем ее экономико-математическую или информационную модель. Кроме того, каждая из этих задач имеет определенный набор входных данных и результат решения, представленный в определенной форме.

Состав, порядок и принципы взаимодействия функциональных подсистем устанавливаются с учетом достижения стоящей перед экономическим объектом цели функционирования. Основные принципы выделения самостоятельных подсистем, комплексов задач и отдельных расчетов — их относительная самостоятельность, т. е. наличие объекта управления, наличие конкретного набора функций и соответствующих им задач с четко выраженной целью функционирования.

В качестве примера информационной системы можно привести систему управления средним специальным учебным заведением (рис. 1.3). Целью ее является автоматизация документооборота и информационное обеспечение управления качеством образования. Система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи, и так далее. Директор, заместители директора и администратор системы имеют полномочный доступ ко всем подсистемам и возможность использования всех функций программы. Представляемая информационная система управления учебным заведением включает следующие подсистемы: прием абитуриентов и их зачисление; ведение кадрового учета студентов; контроль выдачи учебных часов; планирование учебного процесса; контроль и анализ успеваемости студентов; оценка преподавательского состава.

Для каждой подсистемы определены свои входные и выходные данные. Каждая из подсистем имеет собственные цели, задачи, функциональные характеристики, четкий интерфейс с другими подсистемами. Так, основными функциями подсистемы приема абитуриентов являются следующие:

- сбор анкетной информации о поступающих;

- ведение базы данных абитуриентов;
- контроль и проведение вступительных испытаний;
- зачисление студентов по результатам экзаменов;
- формирование необходимой отчетности по итогам приема (списков, статистических отчетов и т. д.).

К основным функциям подсистемы ведения кадрового учета студентов можно отнести:

- ведение базы данных студентов;
- формирование личной карточки студента;
- формирование кадровых приказов, справок для обучающихся и завершивших обучение студентов;
- формирование обязательной отчетности перед вышестоящими организациями и государственными органами.

Для подсистемы анализа и контроля успеваемости можно определить следующий состав выполняемых функций:

- формирование сводных ведомостей по успеваемости и посещаемости студентов;
- формирование приложений к дипломам и академических справок;
- формирование различных диаграмм и отчетов, по которым можно провести анализ успеваемости студентов, а также сделать выводы по качеству образования и т. д.

Взаимодействие между подсистемами информационной системы укладывается в ограниченные стандартные рамки.

Для некоторых предметных областей возможна типизация функций управления, структуры данных и алгоритмов обработки. Для снижения затрат на проектирование, внедрение и сопровождение информационных систем, а следовательно, и для повышения их эффективности используется типизация проектных решений по отдельным функциональным задачам и на этой основе по функциональным подсистемам и системе в целом. В свою очередь, ориентация на типовые проектные решения служит основой для автоматизации процесса проектирования информационных систем, что ведет к снижению затрат на разработку и сокращению сроков проектирования систем. Например, типовая конфигурация системы «1С:Бухгалтерия» применяется на сотнях тысяч отечественных предприятий и не требует дополнительных доработок.

#### **1.1.4. Обеспечивающая часть**

Обеспечивающая часть ИС — это совокупность средств, с использованием которых решаются задачи функциональных подсистем.

К компонентам обеспечивающей части информационных систем относятся:

- техническое обеспечение;
- математическое обеспечение;
- программное обеспечение;
- информационное обеспечение;
- организационное обеспечение;
- правовое обеспечение.

Кроме того, для крупномасштабных информационных систем рассматривают в качестве отдельных компонентов также:

- эргономическое обеспечение;
- технологическое обеспечение;
- лингвистическое обеспечение;
- кадровое обеспечение и др.

Рассмотрим кратко основное содержание компонентов обеспечивающей части.

**Техническое обеспечение.** Техническое обеспечение представляет собой комплекс технических средств для сбора, регистрации, передачи, обработки, отображения и тиражирования информации. Комплекс технических средств составляют устройства вычислительной техники; исполнительные устройства; средства оргтехники; устройства сбора, накопления, обработки, передачи и вывода информации; устройства передачи данных; линии связи и др.

Структурными элементами технического обеспечения наряду с техническими средствами являются также методические материалы и руководства, техническая документация и обслуживающий их персонал. Документацией оформляются предварительный выбор технических средств, организация их эксплуатации, технологический процесс обработки данных, технологическое оснащение. Документацию можно условно разделить на три группы:

- общесистемную, включающую государственные и отраслевые стандарты по техническому обеспечению;
- специализированную, содержащую комплекс методик по всем этапам разработки технического обеспечения;
- нормативно-справочную, используемую при выполнении расчетов по техническому обеспечению.

**Математическое обеспечение.** Математическое обеспечение — это совокупность математических методов, моделей и алгоритмов обработки информации, используемых при решении функциональных задач и в процессе автоматизации проектных работ. Математическое обеспечение включает в себя средства моделирования процессов управления, методы и средства решения типовых задач управления, методы оптимизации исследуе-



мых управленческих процессов и принятия решений (методы многокритериальной оптимизации, математического программирования, математической статистики, теории массового обслуживания и т. п.). Техническая документация по этому виду обеспечения информационных систем содержит описание задач по алгоритмизации, экономико-математические методы и модели решения задач. Персонал составляют специалисты в области организации управления объектом, постановщики функциональных задач, специалисты по моделированию процессов управления и вычислительным методам, проектировщики информационных систем.

**Программное обеспечение.** Программное обеспечение — это совокупность алгоритмов и программ для реализации целей и задач информационной системы, а также для нормального функционирования комплекса технических средств. Классификация программного обеспечения представлена на рис. 1.4.

*Системное программное обеспечение* — это совокупность программ и программных комплексов для обеспечения работы компьютера и вычислительных устройств.

Назначение системного программного обеспечения:

- создание операционной среды функционирования других программ;
- обеспечение надежной и эффективной работы самого компьютера и вычислительной сети;
- проведение диагностики и профилактики аппаратуры компьютера и вычислительных сетей;
- выполнение вспомогательных технологических процессов (копирование, архивация, восстановление файлов программ и баз данных и т. п.).

Базовое программное обеспечение — это минимальный набор программных средств, обеспечивающих работу компьютера (операционные системы).

Сервисное программное обеспечение — это программы и программные комплексы, которые расширяют возможности базового программного обеспечения и организуют более удобную среду

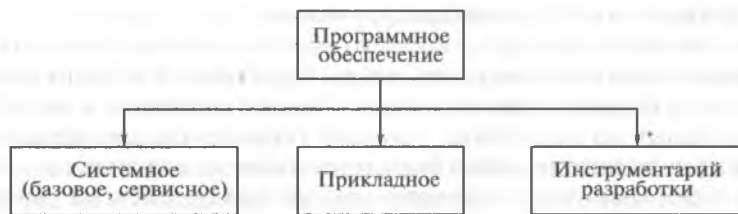


Рис. 1.4. Классификация программного обеспечения

работы пользователя (утилиты для обслуживания компьютера, архиваторы и т. п.).

*Прикладное программное обеспечение* служит программным инструментарием решения функциональных задач и является самым многочисленным классом программного обеспечения. В данный класс входят программные продукты, выполняющие обработку информации различных предметных областей. Таким образом, прикладное программное обеспечение — это комплекс взаимосвязанных программ для решения задач определенного класса предметной области. Данный класс программных средств наиболее представительен, что обусловлено прежде всего широким применением средств компьютерной техники во всех сферах деятельности человека, созданием автоматизированных информационных систем различных предметных областей.

*Инструментарий разработки программного обеспечения* — это программные продукты поддержки технологии программирования. Инструментарий включает в себя специализированное программное обеспечение, которое является инструментальным средством разработки. Программное обеспечение данного класса поддерживает все технологические этапы процесса проектирования, программирования, отладки и тестирования создаваемых программ. Пользователями данного программного обеспечения являются системные и прикладные программисты.

В состав программного обеспечения помимо общесистемных и специальных программ входят также инструктивно-методические материалы по применению средств программного обеспечения.

**Информационное обеспечение.** Информационное обеспечение — это совокупность необходимой информации, циркулирующей в системе, средства и методы, стандартизирующие процессы ее хранения и обработки. Информационное обеспечение образует информационную базу системы. Оно включает в себя специально организованную совокупность показателей, классификаторов и кодовых обозначений элементов информации, унифицированные системы документации, массивы информации в базах и банках данных на машинных носителях, а также средства, обеспечивающие надежность хранения, своевременность и качество технологии обработки информации.

Для создания информационного обеспечения необходимо ясно понимать цели, задачи, функции всей системы. Следует выявить движения информации от момента возникновения и до ее использования на различных уровнях управления. Движение информации в системе может быть представлено для анализа в виде схем информационных потоков, как, например, система управления учебным заведением СПО на рис 1.3. (В главе 4 излагаются принципы построения информационных моделей.)

Информационное обеспечение должно быть достаточным для поддержания всех автоматизируемых функций объекта. Кроме того, должна поддерживаться совместимость информационного обеспечения взаимодействующих подсистем (данные одной подсистемы должны быть доступны для данных другой взаимодействующей подсистемы). В информационной системе необходимо предусмотреть средства контроля входной и выходной информации, способы обновления данных в информационных массивах, контроля целостности информационной базы, защиту от несанкционированного доступа.

Информационную базу подразделяют на немашинную и машинную части, при этом немашинная часть также может быть представлена на машинных носителях.

К немашинной части относят системы классификации и кодирования информации, нормативно-справочную информацию, методические инструктивные материалы, унифицированные системы документации (типовые формы, документы, проекты, решения). К немашинной части можно отнести ту информацию, которую используют вне информационной системы.

К машинной части относят ту информацию, которая используется только внутри информационной системы — совокупность входной и выходной информации, базы данных, экранные формы для ввода первичных данных в компьютер или вывода результирующей информации и др.

**Организационное обеспечение.** Организационное обеспечение — это совокупность методов и средств, регламентирующих взаимодействие работников с техническими средствами и между собой в процессе разработки и эксплуатации информационной системы. Организационное обеспечение реализуется в различных методических и руководящих материалах по стадиям разработки, внедрения и эксплуатации ИС. Это различного рода документы (приказы, положения, инструкции, штатные расписания, квалификационные требования и др.), определяющие положение автоматизированной системы в организации, ее структуру и схему функционирования и регламентирующие деятельность персонала в условиях функционирования информационной системы.

Организационное обеспечение реализует следующие функции:

- анализ существующей системы управления организацией, где будет использоваться ИС, и выявление задач, подлежащих автоматизации;
- подготовку технического задания на проектирование ИС и технико-экономическое обоснование ее эффективности;
- разработку управленческих решений, направленных на повышение эффективности системы управления.

Организационное обеспечение создается по результатам предпроектного обследования и представляет собой комплекс документов, составленный в процессе проектирования ИС, утвержденный и положенный в основу эксплуатации информационной системы.

**Правовое обеспечение.** Правовое обеспечение представляет собой совокупность правовых норм, регламентирующих правоотношения при создании, внедрении и эксплуатации ИС.

Главная цель правового обеспечения — укрепление законности. В состав правового обеспечения входят законы, указы, постановления государственных органов власти, приказы, инструкции и другие нормативные документы. В правовом обеспечении можно выделить:

- общую часть, регулирующую функционирование любой информационной системы;

- локальную часть, регулирующую функционирование конкретной информационной системы.

На этапе разработки информационных систем правовое обеспечение включает нормативные акты, связанные с договорными отношениями разработчика и заказчика, с правовым регулированием различных отклонений в ходе этого процесса, а также акты, необходимые для обеспечения процесса разработки ИС различными видами ресурсов.

На этапе функционирования информационных систем правовое обеспечение включает:

- определение статуса ИС;
- правовое положение в организации;
- права, обязанности и ответственность персонала;
- порядок создания и использования информации в ИС;
- процедуры регистрации, сбора, хранения, передачи и обработки информации;
- порядок приобретения и использования вычислительной и телекоммуникационной техники и других технических средств;
- создание и использование математического и программного обеспечения.

Часть правового обеспечения — защита информационных систем. Она подразумевает ограничение несанкционированного доступа к программам, техническому обеспечению, документации, защита их от преднамеренного разрушения и хищения, а также исключение несанкционированного копирования (тиражирования) программного обеспечения и баз данных. Правовые методы защиты включают: патентную защиту, закон о производственных секретах, лицензионные соглашения и контракты.

Патентная защита устанавливает приоритет в разработке и использовании нового подхода или метода, примененного при разработке программ, удовлетворяет их оригинальность.

Статус производственного секрета для программы ограничивает круг лиц, допущенных к эксплуатации, а также определяет меру их ответственности за разглашение секретов.

Лицензионное соглашение распространяется на все аспекты правовой охраны программных продуктов, включая авторское право, патентную защиту, производственные секреты. Наиболее часто используются лицензионные соглашения (лицензия) на передачу авторских прав.

Лицензия — договор на передачу одним лицом другому лицу права на использование имени, продукции, технологии или услуги. Лицензиар увеличивает свои доходы сбором лицензионных платежей, расширяет область распространения программного обеспечения или базы данных. Лицензиат извлекает доходы за счет их применения. В лицензии оговариваются все условия эксплуатации информационной системы, в том числе создание копий. Существует несколько типов лицензий на программное обеспечение.

*Исключительная лицензия* — продажа всех имущественных прав на программное или информационное обеспечение, покупателю предоставляется исключительное право на их использование, а автор или владелец патента отказывается от самостоятельного их применения или предоставления другим лицам.

*Простая лицензия* — лицензиар предоставляет лицензиату использовать программное или информационное обеспечение, оставляя за собой право применять их и предоставлять на аналогичных условиях неограниченному числу лиц (лицензиат может лишь продавать копии приобретенного программного или информационного обеспечения).

*Этикеточная лицензия* — лицензия на одну копию программного или информационного обеспечения. Данный тип применяется в розничной продаже.

**Лингвистическое обеспечение.** Лингвистическое обеспечение представляет собой совокупность языковых средств, используемых для повышения эффективности разработки системы и облегчения общения человека с ЭВМ. В его состав входят:

- информационные языки для описания структурных единиц информационной базы (документов, показателей, реквизитов и т. п.);
- языки управления и манипулирования данными информационной базы;
- языковые средства информационно-поисковых систем;
- языковые средства автоматизации проектирования информационных систем;
- диалоговые языки специального назначения;

– система терминов и определений, используемых в процессе разработки и функционирования автоматизированных информационных систем и др.

**Эргономическое обеспечение.** Эргономическое обеспечение — это совокупность методов и средств, создающих благоприятные условия для быстрого освоения АИС, высокоэффективной и безошибочной деятельности человека в системе.

Эргономическое обеспечение как совокупность методов и средств, используемых на разных этапах разработки и функционирования ИС, предназначено для создания оптимальных условий высококачественной, высокоэффективной деятельности человека при использовании ИС. В состав эргономического обеспечения информационных систем входят:

– комплекс документации, содержащей эргономические требования к рабочим местам, информационным моделям, условиям деятельности персонала, а также набор наиболее целесообразных способов реализации этих требований и осуществления эргономической экспертизы уровня их реализации;

– комплекс методов учебно-методической документации и технических средств, обеспечивающих обоснование и формулировку требований к уровню подготовки персонала, а также формирование системы отбора и подготовки персонала;

– комплекс методов, обеспечивающих высокую эффективность деятельности персонала.

Обеспечивающие подсистемы ИС в отличие от функциональных подсистем, как правило, аналогичны по составу для информационных систем различных экономических объектов.

## 1.2. Классификация информационных систем

Информация в современном мире превратилась в один из наиболее важных ресурсов, а информационные системы стали необходимым инструментом практически во всех сферах деятельности. Разнообразие задач, решаемых с помощью ИС, привело к появлению множества разнотипных систем, различающихся принципами построения и заложенными в них правилами обработки информации.

Информационные системы можно классифицировать по целому ряду различных признаков.

По степени автоматизации информационные системы подразделяют на ручные, автоматические и автоматизированные.

*Ручные системы* характеризуются отсутствием современных технических средств переработки информации и выполнением всех операций человеком.

В *автоматических системах* все операции по переработке информации выполняются без участия человека.

*Автоматизированные системы* предполагают участие в процессе обработки информации и человека, и технических средств, причем главная роль в выполнении рутинных операций обработки данных отводится компьютеру. Именно этот класс систем соответствует современному представлению понятия «информационная система».

По типу используемых данных информационные системы можно классифицировать на фактографические и документальные.

*Фактографические системы* предназначены для хранения и обработки структурированных данных. Над такими данными можно выполнять различные операции — поиск, фильтрацию, сортировку, агрегирование и т. д.

В *документальных системах* информация представлена в виде документов, состоящих из рефератов, текстов, описаний и пр. Поиск по неструктурированным данным осуществляется с использованием семантических признаков. Отобранные документы предоставляются пользователю, а обработка данных в таких системах практически не производится.

При создании или при классификации информационных систем неизбежно возникают проблемы, связанные с формальным (математическим и алгоритмическим) описанием решаемых ими задач. От степени формализации во многом зависят эффективность работы всей системы, а также уровень автоматизации, определяемый степенью участия человека при принятии решения на основе получаемой информации. Чем точнее математическое описание задачи, тем выше возможности компьютерной обработки данных и тем меньше степень участия человека в процессе ее решения. Это и определяет степень автоматизации задачи.

Различают три типа задач, для которых создаются информационные системы: структурированные (формализованные), неструктурированные (неформализованные) и частично структурированные.

Структурированная (формализованная) задача — задача, где известны все ее элементы и взаимосвязи между ними. В структурированной задаче удастся выразить ее содержание в форме математической модели, имеющей точный алгоритм решения. Подобные задачи обычно приходится решать многократно, и они носят рутинный характер. Целью использования информационной системы для решения структурированных задач является полная автоматизация их решения, т. е. сведение роли человека к нулю. Например, задача расчета заработной платы — структурированная задача, где полностью известен алгоритм решений. Рутинный

характер этой задачи определяется тем, что расчеты всех начислений и удержаний многократно повторяются ежемесячно для большого числа сотрудников.

Неструктурированная (неформализованная) задача — задача, в которой невозможно выделить элементы и установить между ними связи. Решение неструктурированных задач из-за сложности создания математического описания и разработки алгоритма связано с большими трудностями. Решение в таких случаях принимается человеком на основе опыта и, возможно, косвенной информации из разных источников.

Заметим, что в практике работы любой организации существует сравнительно немного полностью структурированных или совершенно неструктурированных задач. О большинстве задач можно сказать, что известна лишь часть их элементов и связей между ними. Такие задачи называются частично структурированными. В этих условиях в информационной системе получаемая информация анализируется человеком, который будет принимать решения.

Информационные системы, используемые для решения частично структурированных задач, подразделяются на два вида.

1. Создающие управленческие отчеты и ориентированные главным образом на обработку данных (поиск, сортировку, агрегирование, фильтрацию). Используя сведения, содержащиеся в этих отчетах, управляющий принимает решение. Информационные системы данного вида обеспечивают информационную поддержку пользователя, т. е. предоставляют доступ к информации в базе данных и имеют возможность получения результирующей информации либо в агрегированном виде, либо как сравнительные отчеты в различных разрезах за произвольные промежутки времени и т. п.

2. Разрабатывающие возможные альтернативы решения. Принятие решения при этом сводится к выбору одной из предложенных альтернатив. Информационные системы могут быть модельными или экспертными.

*Модельные системы* предоставляют пользователю математические, статистические, финансовые и другие модели, использование которых облегчает выработку и оценку альтернатив решения. Пользователь может получить недостающую ему для принятия решения информацию путем ведения диалога с системой в процессе исследования модели.

Основные функции модельной информационной системы:

— возможность работы в среде типовых математических моделей, включая решение основных задач моделирования типа «как сделать, чтобы?» и «что будет, если?»;

— достаточно быстрая и адекватная интерпретация результатов моделирования;



- оперативная подготовка и корректировка входных параметров и ограничений модели;
- возможность графического отображения динамики модели;
- возможность объяснения пользователю необходимых шагов формирования и работы модели.

*Экспертные системы* обеспечивают выработку и оценку возможных альтернатив пользователем за счет создания экспертных систем, связанных с обработкой знаний.

Экспертная поддержка принимаемых пользователем решений реализуется на двух уровнях.

На первом уровне экспертная поддержка исходит из концепции «типовых управленческих решений», в соответствии с которой часто возникающие в процессе управления проблемные ситуации можно свести к некоторым однородным классам управленческих решений, т. е. к некоторому типовому набору альтернатив. Для реализации экспертной поддержки на этом уровне создается информационный фонд хранения и анализа типовых альтернатив.

Если возникающая проблемная ситуация не ассоциируется с имеющимися классами типовых альтернатив, в работу должен вступать второй уровень экспертной поддержки управленческих решений. Этот уровень генерирует альтернативы на базе имеющихся в информационном фонде данных и правил преобразования.

Существует классификация информационных систем по характеру обработки данных (рис. 1.5).

*Информационно-поисковые системы* производят ввод, систематизацию, хранение, выдачу информации по запросу пользователя без сложных преобразований данных. Пользователь обращается к системе с информационным запросом — текстом, отра-



Рис. 1.5. Классификация информационных систем по характеру обработки данных

жающим информационную потребность (например, список книг определенного автора). Поиск информации ведется в поисковом массиве, который формируется (и по мере необходимости обновляется) администраторами системы или пользователем. Элементы поискового массива вводятся в информационно-поисковую систему на естественном (или близком к нему) языке, а затем подвергаются индексированию, т. е. переводу на формальный информационно-поисковый язык. Информационно-поисковыми являются системы библиотечного обслуживания, системы резервирования и продажи билетов, бронирования мест в гостиницах и пр.

*Информационно-решающие системы* кроме вышеперечисленных операций осуществляют операции переработки информации по определенному алгоритму. Они подразделяются на управляющие и советующие.

В *управляющих системах* результирующая информация непосредственно трансформируется в принимаемые человеком решения. Для этих систем характерны задачи расчетного характера и обработка больших объемов данных. Например, информационные системы планирования производства или заказов, системы бухгалтерского учета и т. п.

*Советующие системы* вырабатывают информацию, которая принимается человеком к сведению и учитывается при формировании управленческих решений и не инициирует конкретные действия. Эти системы имитируют интеллектуальные процессы обработки не данных, а знаний. К советующим системам относят экспертные системы.

На рис. 1.6 представлен один из способов классификации информационных систем по сфере применения. На самом деле области применения информационных систем весьма разнообразны, и потому классифицировать их по сфере применения можно самыми различными способами. Мы рассмотрим наиболее распространенную классификацию.

*Информационные системы организационного управления* предназначены для автоматизации функций управленческого персонала как промышленных предприятий, так и непромышленных объектов (гостиниц, банков, магазинов, учебных заведений и пр.). Специфика данного класса информационных систем заключается в использовании экономических (хозяйственных, производственных) параметров при отражении состояния элементов системы. Информационная система является частью организации (фирмы, предприятия), а один из ключевых элементов организации — ее структура. Координация работы всех подразделений организации осуществляется через органы управления разного уровня.

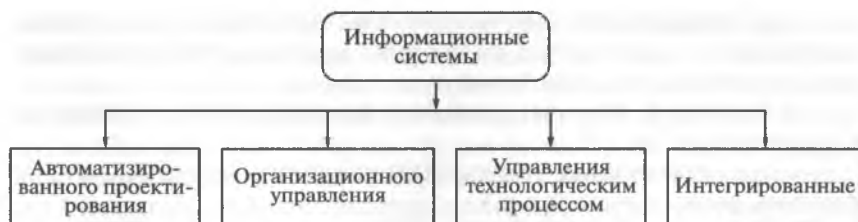


Рис. 1.6. Классификация информационных систем по сфере применения

Под управлением предприятием (фирмой, компанией и т. п.) понимают обеспечение поставленной цели при условии реализации следующих функций: организационной, плановой, учетной, анализа, контрольной, стимулирования.

1. Организационная функция заключается в разработке организационной структуры и комплекса нормативных документов: штатного расписания фирмы, отдела, лаборатории, группы и т. п. с указанием подчиненности, ответственности, сферы компетенции, прав, обязанностей и т. п. Чаще всего это излагается в положении по отделу, лаборатории или должностных инструкциях.

2. Планирование (плановая функция) состоит в разработке и реализации планов по выполнению поставленных задач. Например, бизнес-план для всей фирмы, план производства, план маркетинговых исследований, финансовый план, план проведения научно-исследовательской работы и т. д. на различные сроки (год, квартал, месяц, день).

3. Учетная функция заключается в разработке или использовании уже готовых форм и методов учета показателей деятельности фирмы, например бухгалтерский учет. В общем случае учет можно определить как получение, регистрацию, накопление, обработку и предоставление информации о реальных хозяйственных процессах.

4. Анализ (аналитическая функция) связывается с изучением итогов выполнения планов и заказов, определением влияющих факторов, выявлением резервов, изучением тенденций развития и т. д. Выполняется анализ разными специалистами в зависимости от сложности и уровня анализируемого объекта или процесса. Анализ результатов хозяйственной деятельности фирмы за год и более проводят специалисты, а на уровне цеха, отдела — менеджер этого уровня (начальник или его заместитель) совместно со специалистом-экономистом.

5. Контрольная функция чаще всего осуществляется менеджером: контроль за выполнением планов, расходованием материальных ресурсов, использованием финансовых средств и т. п.

6. Стимулирование (мотивационная функция) предполагает разработку и применение различных методов стимулирования труда подчиненных работников:

– финансовые стимулы (зарплата, премия, акции, повышение в должности и т. п.);

– психологические стимулы (благодарности, грамоты, звания, степени, доски почета и т. п.).

*Информационные системы управления технологическими процессами* служат для автоматизации функций производственного персонала по контролю и управлению производственными операциями. В таких системах обычно предусматривается наличие развитых средств измерения параметров технологических процессов (температуры, давления, химического состава продукта и т. п.), процедур контроля допустимости значений параметров и регулирования технологических процессов. Например, в качестве средств автоматизации процессов изготовления деталей используют оборудование с числовым программным управлением (ЧПУ), гибкие производственные линии, автоматизированные линии, промышленные роботы, роботизированные технологические комплексы. Проектирование технологических процессов обработки заготовок деталей на автоматизированной линии начинают с анализа информации о продукции. Технологический процесс разрабатывают поэтапно. Необходимым элементом перехода от локальной автоматизации к гибким производственным линиям являются роботизированные технологические комплексы. Структура информационной системы управления технологическими процессами и степень участия в ней человека зависит от уровня автоматизации, избираемого для данных конкретных условий, и характера связей с внешними и смежными производственными подразделениями. Такого рода системы можно подразделить на три основные подсистемы: обработки, обслуживания, контроля и управления. Подсистемы обработки и обслуживания формируются на основе технологического оборудования. Подсистема контроля и управления состоит из комплекса программных средств контроля, измерений, регулирования, вычислений, логического управления, регистрации и аварийной защиты. Используются датчики контроля ориентации и точности расположения изготавливаемых изделий, выполняемого технологического процесса, правильности срабатывания оборудования и т. д.

*Системы автоматизированного проектирования* (САПР — Computing Aided Design, CAD) предназначены для автоматизации функций инженеров-проектировщиков, конструкторов, архитекторов, дизайнеров, т. е. для выполнения расчетов, чертежей, планов, схем, моделирования объектов. Различают САПР конструктора и САПР технолога. Основные функции систем автоматизи-

рованного проектирования — инженерные расчеты, создание графической и проектной документации, моделирование проектируемых объектов, увеличение производительности труда проектировщиков, сокращение сроков проектирования, повышение качества разработки проектов. Развитие систем автоматизированного проектирования опирается на прочную научно-техническую базу. Это современные средства вычислительной техники, новые способы представления и обработки информации, создание новых численных методов решения инженерных задач и оптимизации. Системы автоматизированного проектирования дают возможность на основе новейших достижений фундаментальных наук отрабатывать и совершенствовать методологию проектирования, стимулировать развитие математической теории проектирования сложных систем и объектов. К этому же классу систем можно отнести автоматизированные системы поддержки производства (Computing Aided Manufacturing, CAM), автоматизированные системы инженерного проектирования (Computing Aided Engineering, CAE).

*Интегрированные (корпоративные)* информационные системы используются для автоматизации всех функций предприятия, как организационных, так и производственных, и охватывают весь цикл работ от планирования деятельности до сбыта продукции. Они включают в себя ряд модулей (подсистем), работающих в едином информационном пространстве и выполняющих функции поддержки соответствующих направлений деятельности. Подробнее интегрированные информационные системы рассмотрены в главе 5.

### **1.3. Архитектуры ИС**

Современные информационные системы характеризуются огромными объемами хранимой и обрабатываемой информации, сложностью ее организации, а также необходимостью удовлетворять разнообразные требования пользователей.

Существуют различные способы организации данных и методы доступа к ним. Традиционные архитектурные решения информационных систем основаны на использовании выделенных файловых серверов или серверов баз данных (БД). Существуют также варианты архитектур корпоративных информационных систем, базирующихся на технологии Интернет. В настоящее время достаточно популярной является также разновидность архитектуры информационной системы, основывающаяся на концепции «хранилища данных» (DataWarehouse) — интегрированной информационной среды, включающей разнородные информационные ресурсы. И наконец, для построения глобальных распределенных инфор-

мационных приложений используется архитектура интеграции информационно-вычислительных компонентов на основе объектно-ориентированного подхода.

### **1.3.1. Архитектура файл-сервер**

В архитектуре файл-сервер базы данных хранятся на сервере, клиент обращается к серверу с файловыми командами, а механизм управления всеми информационными ресурсами находится на компьютере клиента.

Файл-серверные базы данных могут быть доступны многим клиентам через сеть. Сама база данных хранится на сетевом файл-сервере в единственном экземпляре. Для каждого клиента во время работы создается локальная копия данных, с которой он манипулирует. При этом возникают проблемы, связанные с возможным одновременным доступом нескольких пользователей к одной и той же информации. Эти проблемы решаются разработчиками приложений баз данных (каждый раз при обращении к данным проверяется их доступность).

Архитектура файл-сервер обладает значительными недостатками. Одним из них является непроизводительная загрузка сети. При каждом запросе клиента данные в его локальной копии полностью обновляются из базы данных на сервере. Даже если запрос относится всего к одной записи, обновляются все записи базы данных. Если записей в базе данных много, то и при небольшом числе клиентов сеть будет загружена очень основательно, что серьезно скажется на скорости выполнения запросов. В результате циркуляции в сети больших объемов избыточной информации резко возрастает нагрузка на сеть, что приводит к соответствующему снижению ее быстродействия и производительности информационной системы в целом. Значительный сетевой трафик особенно сказывается при организации удаленного доступа к базам данных на файл-сервере через низкоскоростные каналы связи.

Другой недостаток связан с тем, что забота о целостности данных при такой организации работы возлагается на программы клиентов. Одним из традиционных средств, на основе которых создаются файл-серверные системы, являются локальные СУБД. Такие системы, как правило, не отвечают требованиям обеспечения целостности данных, в частности, они не поддерживают обработки транзакций (завершенных операций с документами). Поэтому при их использовании задача обеспечения целостности данных возлагается на клиентские приложения, что приводит к их усложнению. Если они недостаточно тщательно продуманы, в

базу данных легко занести ошибки, которые могут отразиться на всех пользователях.

Кроме того, в файл-серверной архитектуре изменения, сделанные в базе данных одним пользователем, не видны другим пользователям. Пока один пользователь редактирует какую-либо запись, она заблокирована для других клиентов. Возникает необходимость синхронизации работы отдельных пользователей, связанная с блокировкой записей.

И еще один недостаток — управление базой данных осуществляется с разных компьютеров, поэтому в значительной степени затруднена организация контроля доступа, соблюдения конфиденциальности, что также усложняет поддержку целостности базы данных.

Тем не менее файл-серверная архитектура привлекает своей простотой и доступностью. Поэтому файл-серверные информационные системы до сих пор представляют интерес для малых рабочих групп и более того нередко используются в качестве информационных систем в масштабах предприятия.

### **1.3.2. Архитектура клиент-сервер**

Архитектура клиент-сервер предназначена для разрешения проблем файл-серверных приложений путем разделения компонентов приложения и размещения их там, где они будут функционировать наиболее эффективно. Особенностью архитектуры клиент-сервер является наличие выделенных серверов баз данных, принимающих запросы на языке структурированных запросов (Structured Query Language, SQL) и выполняющих поиск, сортировку и агрегирование информации. Отличительная черта серверов баз данных — наличие хранилища метаданных, в котором записаны структура базы данных, ограничения целостности данных, форматы и серверные процедуры обработки данных по вызову или по событиям в программе. Объектами разработки в таких приложениях, помимо диалога и логики обработки, являются прежде всего реляционная модель данных и связанный с ней набор SQL-операторов для типовых запросов к базе данных.

База данных на платформе клиент-сервер используется для систем с большим количеством пользователей. В этом случае доступ к базе данных для группы клиентов выполняется специальным компьютером — сервером. Клиент дает задание серверу выполнить те или иные операции поиска или обновления базы данных. И мощный сервер, ориентированный на операции с запросами, выполняет их и сообщает клиенту результаты своей работы. Подобная организация работы повышает эффективность

выполнения приложений за счет использования мощности сервера, разгружает сеть, обеспечивает хороший контроль целостности данных.

В базах данных клиент-сервер возникает дополнительная проблема — спроектировать приложение так, чтобы оно максимально использовало возможности сервера и минимально нагружало сеть, передавая через нее только минимум информации. Для сокращения нагрузки на сеть, упрощения администрирования приложений и ускорения работы клиентских приложений с удаленной БД вся логика принятия решений оформляется в виде хранимых процедур и выполняется на сервере баз данных. Хранимые процедуры представляют собой подпрограммы, принимающие и возвращающие параметры и способные выполнять запросы к БД, условные ветвления и циклическую обработку. Хранимые процедуры пишутся на специальном алгоритмическом языке. В них программируются часто повторяемые последовательности запросов к БД. Текст процедур хранится на сервере в откомпилированном виде. Преимущества в использовании хранимых процедур очевидны:

- отпадает необходимость синтаксической проверки каждого запроса и его компиляции перед выполнением, что повышает скорость выполнения запросов;

- отпадает необходимость реализации в клиентской программе запросов, определенных в теле хранимых процедур;

- увеличивается скорость обработки транзакций, так как вместо подчас длинного SQL-запроса по сети передается относительно короткое обращение к хранимой процедуре.

Хранимые процедуры улучшают целостность приложений и БД, гарантируют актуальность коллективных операций и вычислений. Улучшается сопровождение таких процедур, а также безопасность (нет прямого доступа к данным).

Однако следует помнить, что перегрузка хранимых процедур прикладной логикой может перегрузить сервер, что приведет к потере производительности. Эта проблема особенно актуальна при разработке крупных информационных систем, в котором к серверу может одновременно обращаться несколько клиентов. Поэтому в большинстве случаев следует принимать компромиссные решения: часть логики приложения размещать на стороне сервера, часть — на стороне клиента. Такие клиент-серверные системы называют системами с разделенной логикой.

Описанная архитектура является двухуровневой и называется «толстым клиентом» (рис. 1.7).

Создание архитектуры клиент-сервер возможно и на основе многотерминальной системы. В этом случае в многозадачной среде сервера приложений выполняются программы пользовате-





Рис. 1.7. Двухуровневая архитектура клиент-сервер

лей, а клиентские узлы вырождены и представлены терминалами.

В настоящее время архитектура клиент-сервер получила признание и широкое распространение как способ организации приложений для рабочих групп и информационных систем корпоративного уровня. Подобная организация повышает эффективность работы приложений за счет использования возможностей сервера базы данных, снижения нагрузки на сеть и обеспечения контроля целостности данных. Повышение безопасности информации связано с тем, что обработка запросов всех клиентов выполняется единой программой, расположенной на сервере. Сервер устанавливает общие для всех пользователей правила использования БД, управляет режимами доступа клиентов к данным, запрещая, в частности, одновременное изменение одной записи разными пользователями.

Также уменьшается сложность клиентских приложений за счет отсутствия у них кода, связанного с контролем БД и разграничением доступа к ней.

Двухуровневые схемы архитектуры клиент-сервер могут привести к некоторым проблемам в сложных информационных приложениях с множеством пользователей и запутанной логикой. Решением этих проблем может стать применение многоуровневой архитектуры.

### 1.3.3. Многоуровневая архитектура

Многоуровневая архитектура является развитием архитектуры клиент-сервер и в своей классической форме состоит из трех уровней (рис. 1.8).



Рис. 1.8. Трехуровневая архитектура клиент-сервер

На нижнем уровне на компьютерах пользователей расположены приложения клиентов, выделенные для выполнения функций и логики представлений, обеспечивающие программный интерфейс для вызова приложения на среднем уровне.

На среднем уровне расположен сервер приложений, на котором выполняется прикладная логика и с которого логика обработки данных выполняет операции с базой данных, т.е. этот уровень обеспечивает обмен данными между пользователями и распределенными базами данных. Сервер приложений размещается в узле сети доступно всем клиентам.

На третьем, верхнем, уровне расположен удаленный специализированный сервер баз данных, принимающий информацию от сервера приложений. Сервер баз данных выделен для услуг обработки данных и файловых операций.

Достоинства трехуровневой архитектуры:

- разгрузка сервера баз данных от выполнения части операций, перенесенных на сервер приложений;
- уменьшение размера клиентских приложений за счет разгрузки их от лишнего кода;
- единое поведение всех клиентов;
- упрощение настройки клиентов — при изменении общего кода сервера приложений автоматически изменяется поведение клиентских приложений.

Трехуровневая архитектура устраняет недостатки двухуровневой модели клиент-сервер. Она позволяет еще больше сбалансировать нагрузку на сеть. С ростом систем клиент-сервер необходимость трех уровней становится все более очевидной.

### 1.3.4. Интернет(интранет)-технологии

Интернет (Internet, Interconnected Networks) — всемирная система объединенных компьютерных сетей, построенная на использовании протокола IP (Internet Protocol) и маршрутизации пакетов данных. Интернет образует единую (всемирную) информационную среду.

Интранет, в отличие от сети Интернет, — внутренняя частная сеть организации. Как правило, Интранет — это Интернет в миниатюре, который построен на использовании протокола IP для обмена и совместного использования некоторой части информации внутри этой организации. В информационную базу могут быть включены сведения о сотрудниках, списки телефонов партнеров и заказчиков, различная корпоративная информация. Чаще всего под этим термином имеют в виду только видимую часть Интранет — внутренний Web-сайт организации, основанный на базовых протоколах HTTP и HTTPS и организованный по принципу клиент-сервер, интранет-сайт доступен с любого компьютера через браузер (Mozilla Firefox, Microsoft Internet Explorer, Opera и др.). Таким образом, Интранет — это что-то вроде частного Интернета, ограниченного виртуальным пространством отдельно взятой организации.

Преимущества использования Интранет очевидны:

- высокая производительность при совместной работе над какими-то общими проектами;
- легкий доступ персонала к данным;
- гибкий уровень взаимодействия: можно менять бизнес-схемы взаимодействия как по вертикали, так и по горизонтали;
- мгновенная публикация актуальной информации (служебные инструкции, стандарты, службы рассылки новостей и т. д.);
- не требуется инсталляция программы-клиента на компьютерах пользователей (в качестве нее используется браузер), соответственно, при изменениях функциональности корпоративной информационной системы обновление клиентского приложения также не требуется;
- быстрый обмен информацией за счет использования удобного Web-интерфейса.

Интранет, к сожалению, обладает и рядом недостатков. В частности, сеть может быть взломана и использована хакерами, также в свободном интерактивном пространстве могут распространяться нелегитимные материалы. Следует учитывать, что легкий доступ к корпоративным данным может спровоцировать их утечку к конкурентам через недобросовестных работников. Кроме того, работоспособность и гибкость Интранета требуют



Рис. 1.9. Соединение интернет(интранет)-технологий с архитектурой клиент-сервер

значительных накладных расходов на разработку и администрирование.

Для создания удобных и простых в использовании и сопровождении информационных систем, эффективно работающих с базами данных, объединяют интернет(интранет)-технологии с многоуровневой архитектурой. При этом структура информационного приложения приобретает следующий вид (рис. 1.9).

Благодаря интеграции интернет(интранет)-технологий с архитектурой клиент-сервер, процесс внедрения и управления корпоративной информационной системой существенно упрощается при сохранении достаточно высокой эффективности и простоты совместного использования информации. Любая интеграция предопределяет вполне конкретные ограничения для доступа в информационное пространство. Если в глобальных сетях таких ограничений практически нет, то в корпоративных сетях они явно присутствуют и согласуются исключительно с интересами корпоративных пользователей или участников партнерских объединений.

## **1.4. Процессы в информационной системе. Эксплуатация информационных систем**

### **1.4.1. Этапы и виды технологических процессов обработки информации**

Технологический процесс обработки информации представляет собой комплекс взаимосвязанных операций по преобразованию информации в соответствии с поставленной целью с момента ее возникновения (входа в систему) до момента потребления пользователем. Технологические процессы весьма разнообразны и могут быть подразделены на этапы и операции.

Этапы технологического процесса — это его относительно самостоятельные части, характеризующиеся логической законченностью, пространственной или временной обособленностью. Этапы делятся на технологические операции, различаются их составом и последовательностью выполнения. Технологический

процесс включает, как правило, первичный и основной этапы. На первичном этапе обеспечивается сбор первичной информации, ее подготовка и регистрация и передача на обработку. На основном этапе происходит непосредственная обработка информации и получение необходимых результатов.

На всех этапах выполняется необходимый объем операций для достижения достоверности и полноты преобразования информации. При этом объектами особого внимания являются время преобразования и качество результатной информации. По содержанию и последовательности преобразования информации различают следующие технологические операции: сбор и регистрация информации, передача, прием, запись на носители, арифметическая и логическая обработка, получение результатной информации и выходных отчетов, передача их пользователю, принятие решений и выработка управляющих воздействий.

По степени механизации и автоматизации операции бывают ручные (выписка первичного документа), механизированные — с использованием технических средств, но преимущественно выполняемые человеком, автоматизированные — в большей степени выполняются техническими средствами, но предполагается и участие человека, автоматические — без участия человека. По роли в технологическом процессе различают рабочие и контрольные операции. Рабочие обеспечивают получение конечного результата, а контрольные — надежность выполнения рабочих операций.

Способы сбора, анализа и обработки информации существенно различаются для структурированных и неструктурированных данных. Наиболее развитыми в настоящее время (с точки зрения задач обработки и анализа информации) являются программные средства обработки структурированных данных, так как структуризацию можно считать первичной и наиболее трудно формализуемой обработкой.

**Информационный поток** — это информация (группа данных), рассматриваемая в процессе ее движения в пространстве и времени в определенном направлении. У этих данных есть общий источник и общий приемник. Поток, состоящий из смысловых структурных элементов, называют сообщением. Действие информации направлено на уменьшение или снятие неопределенности в какой-либо ситуации. Например, в информационных системах управления одной из задач является передача руководству информации, необходимой для определения состояния производства и принятия решения.

При оперировании информацией в процессах ее создания (рождения), сбора, выдачи и потребления огромное значение имеет понятие документирования информации или просто до-

кумента. В большинстве случаев информация предстает и фигурирует в виде документа, исключая ту часть информационных потоков, в которых используются только данные (например, в автоматизированных информационных системах управления технологическими процессами информация рождается в виде показаний датчиков, обрабатывается, выдается и потребляется в виде управляющих сигналов на технологическое оборудование). Под документированием информации в широком смысле слова понимают выделение единичной смысловой части информации (данных) по некоторой предметной области, обособление этой части с приданием ей самостоятельной роли (имя, статус, реквизиты и т. п.). Процесс документирования превращает информацию в информационные ресурсы — отдельные документы или массивы документов в информационных системах.

Документооборот — последовательность прохождения документов с момента их составления или получения до момента их обработки и использования. Документооборот включает в себя комплекс работ с документами: прием, регистрация, рассылка, контроль исполнения, формирование дел, хранение и повторное использование документации. Для эффективного управления организацией и принятия оптимальных решений требуется иметь в распоряжении обширную и достоверную информацию. Достичь этого можно в настоящее время только с помощью средств и методов автоматизации информационных потоков. Для этого необходимо опираться на основные принципы документооборота:

- рациональное и своевременное составление документов;
- однократная регистрация документа, позволяющая однозначно идентифицировать документ, исключение возможности дублирования документов;
- последовательный охват документами всех видов хозяйственной деятельности организации;
- взаимосвязь и рациональная обработка документов;
- сокращение путей прохождения документов.

При сборе информации в информационной системе исходными документами являются те, которые служат источниками информации. Производные документы формируются на основании первичных документов, непосредственно отражающих входную информацию. Конечные документы (выходные документы) являются результатом обработки информации и непосредственно влияют на объект управления.

Итак, технологическим процессом обработки информации можно назвать упорядоченную последовательность взаимосвязанных действий, выполняемых в строго определенной очередности с момента возникновения информации до получения заданных результатов. Технологический процесс обработки информации

зависит от характера решаемых задач, используемых технических средств, систем контроля, числа пользователей, принятого документооборота и т. д.

#### **1.4.2. Организация сбора, размещения, хранения, накопления, преобразования и передачи данных в информационную систему**

Сбор и регистрация информации происходят по-разному на различных экономических объектах. Особое значение при этом придается достоверности, полноте и своевременности первичной информации. На предприятии сбор и регистрация информации происходят при выполнении различных хозяйственных операций (прием готовой продукции, отпуск товара, получение и отгрузка материалов и т. п.), в банках — при выполнении финансово-кредитных операций с юридическими и физическими лицами. Учетные данные могут возникать на рабочих местах в результате подсчета количества обработанных деталей, прошедших сборку готовых изделий, выявления бракованной продукции и т. д.

Хранение и накопление информации вызвано необходимостью многократного ее использования, применением условно-постоянной, справочной и других видов информации, необходимостью комплектации первичных данных до их обработки. Хранение и накопление информации осуществляется в информационных базах на машинных носителях в виде информационных массивов, структурированных определенным образом. На сегодняшний день объемы хранимой информации в десятки терабайтов уже стали реальностью. Информационные ресурсы такого объема уже накоплены в хранилищах данных некоторых крупных корпораций (например, данными терабайтового объема оперируют системы хранения и обработки результатов спутниковой цифровой фотосъемки поверхности Земли).

С хранением и накоплением непосредственно связан поиск данных, т. е. выборка нужных данных из хранимой информации, включая поиск информации, подлежащей корректировке или замене. Процедура поиска информации осуществляется автоматически на основе запроса пользователя.

Обработка информации производится на ПЭВМ, как правило, децентрализованно, в местах возникновения первичной информации, где организуются автоматизированные рабочие места специалистов, а также в вычислительных центрах. В ходе обработки информации формируются выходные отчеты, которые можно просмотреть на экране или вывести на печатающее устройство.

При проектировании технологических процессов ориентируются на режимы их реализации. Режим реализации технологии зависит от объемно-временных особенностей решаемых задач и от режимных возможностей технических средств. Рассмотрим некоторые режимы обработки данных.

Организация вычислительного процесса при пакетном режиме строится без доступа пользователя к ЭВМ. Сбор и регистрация информации, ввод и обработка не совпадают по времени. Вначале пользователь собирает информацию, формирует ее в пакеты в соответствии с какими-либо признаками или типами задач (задачи неоперативного характера с долговременным сроком действия результатов решения). После завершения приема происходит ввод информации в ЭВМ и обработка. Этот режим используется, как правило, при централизованном способе обработки информации.

Интерактивный режим предусматривает непосредственное взаимодействие пользователя с информационно-вычислительной системой, может носить характер запроса (как правило, регламентированного) или диалога с ЭВМ.

Диалоговый режим открывает пользователю возможность непосредственно взаимодействовать с вычислительной системой в допустимом для него темпе работы, реализуя повторяющийся цикл выдачи задания, получения и анализа ответа. При этом ЭВМ сама может инициировать диалог, сообщая пользователю последовательность шагов (представление меню) для получения искомого результата.

Под запросным режимом понимается одноразовое обращение к системе, после которого она выдает ответ и отключается, тогда как в диалоговом режиме система выдает ответ и ждет дальнейших действий пользователя.

Обработка данных может быть централизованной, децентрализованной и распределенной.

*Централизованная обработка информации* была первой исторически сложившейся информационной технологией. Создавались крупные вычислительные центры коллективного пользования, оснащенные большими ЭВМ, позволяющими обрабатывать большие массивы информации, которая затем передавалась пользователям. Принцип централизованной обработки данных не отвечал высоким требованиям надежности процесса обработки, затруднял развитие систем и не мог обеспечить необходимые временные параметры при диалоговой обработке данных в многопользовательском режиме. Кратковременный выход из строя центральной ЭВМ приводил к роковым последствиям для системы в целом, так как приходилось дублировать функции центральной ЭВМ, значительно увеличивая затраты на создание и эксплуатацию систем обработки данных.



*Децентрализованная обработка информации* связана с появлением в 1980-х годах персональных компьютеров и средств телекоммуникаций. Она существенно потеснила централизованную информационную технологию, поскольку дала пользователю широкие возможности при работе с информацией. Появление персональных компьютеров потребовало нового подхода к организации систем обработки данных, созданию новых информационных технологий. Возникло логически обоснованное требование перехода от использования отдельных ЭВМ в системах централизованной обработки данных к распределенной обработке данных. Последняя позволяет повысить эффективность удовлетворения информационных потребностей пользователей, обеспечить гибкость и оперативность принимаемых ими решений и др.

Под *распределенной обработкой данных* понимают обработку приложений несколькими территориально разделенными ЭВМ. При этом в приложениях, связанных с обработкой базы данных, собственно управление базой данных может выполняться централизованно. Распределенная обработка данных (Distributed Data Processing, DDP) — это методика выполнения прикладных программ группой систем. При этом пользователь получает возможность работать с сетевыми службами и прикладными процессами в нескольких взаимосвязанных абонентских системах.

Распределенная обработка данных позволяет повысить эффективность удовлетворения информационных потребностей пользователей, обеспечивает гибкость и оперативность принимаемых ими решений.

Наиболее часто данные размещаются в БД. Базами обычно управляют локальные СУБД, т.е. размещенные на том же компьютере. Когда несколько таких БД удалены друг от друга на большие расстояния, возникает необходимость решения задач управления ими, т.е. распределенными БД. Для решения таких задач между ЭВМ с локальными СУБД и БД организуют сеть передачи данных по каналам связи, а в ней обеспечивают техническую и программную поддержку обмена данными. В этом случае используют ПО, управляющее распределенными базами данных, которые могут образовывать банки данных.

Распределенные базы данных (Distributed DataBase, DDB) — это определенным образом связанные между собой БД, рассредоточенные на какой-либо территории (локально или регионально) и обеспечивающие свободный обмен информацией и поиск данных в них. Распределенная база данных предполагает хранение и выполнение функций управления данными в нескольких узлах и передачу данных между этими узлами в процессе выполнения запросов. Разбиение данных в распределенной базе данных может достигаться путем хранения различных таблиц на разных компью-

терах или даже хранения разных частей и фрагментов одной таблицы на разных компьютерах. Для пользователя или прикладной программы не имеет значения, каким образом распределены данные между компьютерами. Работа с распределенной базой данных осуществляется так же, как и с централизованной, т. е. размещение БД должно быть прозрачно.

При распределенной обработке работа с базой (представление данных, их обработка и др.) ведется на компьютере клиента, а поддержание базы в актуальном состоянии — на сервере. При этом такие БД обычно располагаются на нескольких серверах — различных узлах компьютерной сети, а некоторые данные могут дублироваться.

Размещение частей общей БД бывает избыточным или избыточным. При избыточном размещении определяют степень дублирования частей (фрагментов) единой БД. Чтобы поддерживать целостность БД, необходимо постоянно корректировать все ее копии. Преимущества дублирования уменьшаются, когда увеличивается стоимость хранения ее частей, что связано с необходимостью обеспечивать устойчивость системы.

Создание распределенных баз данных (РБД) вызвано попыткой одновременного решения двух задач: интеграции и децентрализации.

Интеграция подразумевает централизованное управление и ведение баз данных. Децентрализация обеспечивает хранение данных там, где они появились и обрабатываются. При этом снижается стоимость системы и увеличивается степень ее надежности, а также повышается скорость обработки данных.

Выделяют однородные и неоднородные распределенные базы данных. В однородных распределенных базах данных применяют одну СУБД. В неоднородных распределенных базах данных используются несколько различных СУБД. Основная проблема при этом заключается в сложности их интеграции.

Очень важно обеспечить достоверность информации в процессе хранения и обработки. Резервное копирование может защитить данные от потери и является практически единственным и самым надежным способом предохранить данные от потери в результате поломки диска, сбоев электропитания, действий злоумышленников и т. д. Для минимизации потери данных и восстановления утраченных данных нужно иметь стратегию резервного копирования, которая должна вернуть систему обратно в то состояние, которое было до наступления проблемы. Резервное копирование сохраняет базу данных на жестком диске или другом носителе информации. Для обеспечения максимальной надежности хранения данных следует регулярно проводить резервное копирование на внешние носители.

### 1.4.3. Системы классификации информации

Для того чтобы обеспечить эффективный поиск, обработку на ЭВМ и передачу по каналам связи технико-экономической информации, ее необходимо представить в цифровом виде. С этой целью ее нужно сначала упорядочить (классифицировать), а затем формализовать (закодировать) с использованием классификатора.

**К л а с с и ф и к а т о р** — это систематизированный свод наименований и кодов классификационных группировок. Классификаторы по сфере действия разделяются на международные, общегосударственные (общесистемные), отраслевые, локальные.

*Международные классификаторы* входят в состав Системы международных экономических стандартов (СМЭС) и обязательны для передачи информации между организациями разных стран мирового сообщества.

*Общегосударственные (общесистемные) классификаторы* обязательны для организации процессов передачи и обработки информации между экономическими системами государственно-го уровня внутри страны.

*Отраслевые классификаторы* используют для выполнения процедур обработки информации и передачи ее между организациями внутри отрасли.

*Локальные классификаторы* используют в пределах отдельных предприятий.

**К л а с с и ф и к а ц и я о б ъ е к т о в** — это процедура группировки на качественном уровне, направленная на выделение однородных свойств. Применительно к информации как к объекту классификации выделенные классы называют информационными объектами.

Свойства информационного объекта определяются информационными параметрами — **р е к в и з и т а м и**, логически неделимым информационным элементом, описывающим определенное свойство объекта, явления и т.п. Реквизиты выражаются либо числовыми данными (масса, стоимость, год), либо признаками (цвет, марка машины, фамилия).

Классификация предусматривает следующие задачи — выявление общих свойств информационного объекта; разработку правил (алгоритмов) и процедур обработки информации, представленной совокупностью реквизитов. При классификации объектов необходимо соблюдать определенные требования, а именно: полноту охвата объектов, однозначность реквизитов, возможность включения новых реквизитов.

При классификации применяются понятия классификационный признак и его значение, которые позволяют установить сход-

ство или различие объектов. Признак классификации — свойство или характеристика объекта классификации, которое позволяет установить его сходство или различие с другими объектами классификации.

Классификационная группировка — это множество или подмножество, объединяющее часть объектов классификации по одному или нескольким признакам.

В настоящее время чаще всего применяются два типа систем классификации: иерархическая и многоаспектная. Эти системы различаются разной стратегией применения классификационных признаков.

*Иерархическая система классификации* информации построена следующим образом.

1. Исходное множество элементов составляет нулевой уровень и делится в зависимости от выбранного классификационного признака на классы (группировки), которые образуют первый уровень.

2. Каждый класс первого уровня в соответствии с характерным для него классификационным признаком делится на подклассы, которые образуют второй уровень.

3. Каждый класс второго уровня аналогично делится на группы, которые образуют третий уровень и т. д.

Достоинства иерархической системы классификации — простота построения; использование независимых классификационных признаков в различных ветвях иерархической структуры.

Недостатки иерархической системы классификации — жесткая структура, которая приводит к сложности внесения изменений, так как приходится перераспределять все классификационные группировки; невозможность группировать объекты по заранее не предусмотренным сочетаниям признаков.

Многоаспектная система (фасетная и дескрипторная) — это система классификации, которая использует параллельно несколько независимых признаков (аспектов) в качестве основания классификации.

*Фасетная система классификации* информации представляет собой параллельное разделение множества объектов на независимые классификационные группировки по определенному аспекту классификации — фасету. Например, классифицировать фильмы можно в соответствии со следующими группами — тип (документальный, игровой, анимационный); жанр (боевик, комедия, романтика, фантастика); продолжительность; год; страна; режиссер; другие параметры (немой, звуковой, цветной, черно-белый и т. п.). Таким образом, каждый фильм обладает множеством признаков. При поиске нужного фильма используется пересечение требуемых атрибутов.

Эта система классификации позволяет (в отличие от иерархической) выбирать признаки классификации независимо как друг от друга, так и от семантического содержания классифицируемого объекта. При построении фасетной системы классификации необходимо, чтобы значения, используемые в различных фасетах, не повторялись.

Достоинства фасетной системы классификации — возможность создания классификации большой емкости без изменения структуры существующих группировок; возможность простой модификации всей системы классификации без изменения структуры существующих группировок.

Недостаток фасетной системы классификации — сложность ее построения, (необходимо учитывать все разнообразие классификационных признаков).

Для организации поиска информации, для ведения тезаурусов (словарей) эффективно используется *дескрипторная система классификации*, язык которой приближается к естественному языку описания информационных объектов. Суть дескрипторного метода заключается в следующем.

1. Отбирают совокупность ключевых слов (*дескрипторов*) или словосочетаний, описывающих определенную предметную область или совокупность однородных объектов, причем среди ключевых слов могут находиться синонимы.

2. Выбранные ключевые слова и словосочетания подвергают нормализации, т.е. из совокупности синонимов выбирают один или несколько наиболее употребляемых.

3. Создают словарь дескрипторов, т.е. словарь ключевых слов и словосочетаний, отобранных в результате процедуры нормализации.

Между дескрипторами устанавливаются связи, которые позволяют расширить область поиска информации. Связи могут быть трех видов:

— синонимические, указывающие на некоторую совокупность ключевых слов как синонимов («студент — учащийся — обучаемый»);

— родовидовые, отображающие включение некоторого класса объектов в более представительный класс («университет — факультет — кафедра»);

— ассоциативные, соединяющие дескрипторы, обладающие общими свойствами («студент — экзамен — профессор — аудитория»).

Наиболее сложными вопросами, которые приходится решать при разработке классификатора, являются выбор методов классификации и кодирования и выбор системы признаков классификации. Основой классификатора должны быть наиболее су-

ественные признаки классификации, соответствующие характеру решаемых с помощью классификатора задач. При этом данные признаки могут быть или соподчиненными, или несоподчиненными. При соподчиненных признаках классификации и стабильном комплексе задач, для решения которых предназначен классификатор, целесообразно использовать иерархический метод классификации, который представляет собой последовательное разделение множества объектов на подчиненные классификационные группировки. При несоподчиненных признаках классификации и при большой динамичности решаемых задач целесообразно использовать фасетный метод классификации.

#### **1.4.4. Экспортирование структур баз данных**

При работе с информационной системой могут возникать ситуации, когда требуется производить перенос данных из одной базы данных в другую. Этот процесс называется **экспортированием**.

Экспортирование данных требуется при переводе БД на другой физический носитель, при создании копии БД, переносе данных между информационными системами. Задачи переноса данных между ИС возникают при пересечении предметных областей систем, что приводит их к согласованной работе с одними и теми же данными. Например, это могут быть системы различного уровня иерархии (информационные системы министерства и подчиненной организации) либо автоматизированная система бухгалтерского учета на предприятии и информационная система в налоговой инспекции, куда поступает отчетность и т. п. Сложность экспортирования данных зависит от характеристик источника и получателя данных и их соответствия друг другу. Экспортирование может заключаться в простом переносе данных или в выполнении ряда преобразований переносимых данных.

Преобразование данных может быть следующих видов:

— переименование — объекты данных (таблицы, поля и т. п.) источника получают имена в соответствии с организацией данных получателя;

— реструктуризация — изменение структуры БД источника в соответствии со структурой получателя;

— агрегирование — БД получает не просто данные, а некоторый сводный или итоговый отчет;

— кодирование и декодирование — данные изменяются так, чтобы они соответствовали системе кодирования БД-получателя;

— конвертирование — приведение к формату атрибута в БД-получателе;

— согласование — в разных БД могут использоваться разные способы отображения одной и той же информации (например, в километрах и в метрах), необходимо обеспечить их согласование;

— проверка на допустимость значений.

Таким образом, преобразование данных при экспортировании — это сложная задача, требующая создания специального алгоритма запросов (сценария).

## **1.5. Показатели эффективности информационных систем**

Эффективность информационной системы — это совокупность свойств системы, обуславливающих возможность ее использования для удовлетворения определенных потребностей пользователей в соответствии с ее назначением.

Основными показателями эффективности информационных систем являются надежность, достоверность, безопасность.

*Надежность* — свойство системы сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных условиях применения. Надежность информационных систем является средством обеспечения актуальной и достоверной информации на выходе системы.

*Достоверность* — свойство системы, обуславливающее безошибочность производимых ею преобразований информации. Достоверность функционирования информационной системы полностью определяется и измеряется достоверностью ее резульатной информации.

*Безопасность* информационной системы — свойство, заключающееся в способности системы обеспечить конфиденциальность и целостность информации, то есть защиту информации от несанкционированного доступа.

Показатели эффективности должны отражать количественную оценку степени достижения системой поставленной цели.

Эффективность системы является сложным, интегральным свойством и зависит также от ряда простых свойств, таких как:

— прагматическая эффективность — действенность системы, т.е. степень реализации системой своего предназначения;

— техническая эффективность — техническое совершенство системы;

— технологическая эффективность — простота и технологичность разработки и создания системы;

— эксплуатационная эффективность — удобство использования и обслуживания системы и др.

Прагматическую эффективность можно представить показателями достоверности преобразования информации, безопасности информационной системы, точности вычислений и преобразования информации; полноты формирования системой результатной информации, оперативности.

Показатели технической эффективности должны оценивать техническое совершенство информационной системы, научно-технический уровень организации и функционирования этой системы.

Показатели технологическо-эксплуатационной эффективности весьма разнообразны. В качестве таких показателей могут выступать показатели надежности, функциональные возможности, количество обслуживаемых клиентов, производительность, пропускная способность, тактовая частота, временные задержки, емкость памяти, эксплуатационные характеристики, технологии обслуживания и т. п.

Обобщающими показателями эффективности информационной системы являются показатели экономической эффективности, характеризующие целесообразность произведенных на создание и функционирование системы затрат. Расчет затрат обычно не составляет большого труда, а вот расчет результатов остается сложной, до конца не решенной проблемой. Часто прибыль определяется путем экспертной оценки и по аналогии с другими подобными системами, а социальный эффект количественно вообще не определяется.

Подробно о расчете экономической эффективности информационных систем мы будем говорить в завершении изучения курса. А теперь подробнее остановимся на вопросах безопасности информации, так как именно безопасность оказывает огромное влияние и на надежность, и на эффективность, и на достоверность информации в ИС.

## **1.6. Безопасность информационных систем**

Безопасность информационной системы — свойство, заключающееся в способности системы обеспечить конфиденциальность и целостность информации, т.е. защиту информации от несанкционированного доступа, обращенного на ее раскрытие, изменение или разрушение. Информационную безопасность часто указывают среди основных информационных проблем XXI века.



Действительно, хищение информации, ее сознательное искажение и уничтожение часто приводят к трагическим для пострадавшей стороны последствиям, ведущим к разорению и банкротству фирм и даже к человеческим жертвам. Перечень бед от нарушения безопасности информации огромен — тысячи коммерческих компьютерных преступлений, приведших к потерям миллионов долларов, моральные потери, связанные с хищением конфиденциальной информации. Если раньше для успешного совершения революции или переворота важно было захватить почту и телеграф, то теперь необходимо парализовать системы телекоммуникаций.

Информационным ресурсам в сетях общего и корпоративного пользования могут грозить:

- приведение сети в неработоспособное состояние в результате злонамеренных или неосторожных действий, например, путем перегрузки сети бесполезной информацией;

- несанкционированный доступ к конфиденциальным данным как извне, так и изнутри сети, их корыстное использование и разглашение;

- целенаправленное искажение, фальсификация или подмена данных при несанкционированном доступе;

- подмена и искажение информации, предоставленной для свободного доступа (например Web-страниц);

- приводящее к невозможности использования информационных ресурсов вирусное их заражение по каналам сети Интернет, электронной почты или посредством инфицированных внешних носителей (сменных дисков, дискет, CD и DVD-дисков) и т.д.

Все угрозы информационным системам можно объединить в три группы:

- угроза раскрытия — возможность того, что информация станет известной тому, кому не следовало бы ее знать;

- угроза целостности — умышленное несанкционированное изменение (модификация или удаление) данных, хранящихся в вычислительной системе или передаваемых из одной системы в другую;

- угроза отказа в обслуживании — опасность появления блокировки доступа к некоторому ресурсу вычислительной системы.

Методы обеспечения информационной безопасности в зависимости от способа их реализации можно разделить на следующие классы.

*Организационные методы* подразумевают рациональное конфигурирование, организацию и администрирование системы. В первую очередь это касается сетевых информационных систем, операционных систем, полномочий сетевого администратора, набора обязательных инструкций, определяющих порядок доступа и работы в сети.

*Технологические методы* включают в себя технологии выполнения сетевого администрирования, мониторинга и аудита безопасности информационных ресурсов, ведения электронных журналов регистрации пользователей, фильтрации и антивирусной обработки поступающей информации.

*Аппаратные методы* реализуют физическую защиту системы от несанкционированного доступа, аппаратные функции идентификации периферийных терминалов системы и пользователей, режимы подключения сетевых компонентов и т. д.

*Программные методы* — самые распространенные методы защиты информации (например, программы идентификации пользователей, парольной защиты и проверки полномочий, брандмауэры, криптопротоколы и т. д.). Без использования программной составляющей практически невыполнимы никакие, в том числе и первые три группы методов (то есть в чистом виде организационные, технологические и аппаратные методы защиты, как правило, реализованы быть не могут — все они содержат программный компонент). При этом следует иметь в виду, что стоимость реализации многих программных системных решений по защите информации существенно превосходит по затратам аппаратные, технологические и тем более организационные решения (конечно, если использовать лицензионные, а не «пиратские» программы).

Наибольшее внимание со стороны разработчиков и потребителей в настоящее время вызывают следующие направления защиты информации и соответствующие им программно-технические средства защиты:

— от несанкционированного доступа информационных ресурсов автономно работающих и сетевых компьютеров. Наиболее остро проблема стоит для серверов и пользователей интернет- и интранет-сетей. Эта функция реализуется многочисленными программными, программно-аппаратными и аппаратными средствами;

— секретной, конфиденциальной и личной информации от чтения посторонними лицами и целенаправленного ее искажения. Эта функция обеспечивается как средствами защиты от несанкционированного доступа, так и с помощью криптографических средств, традиционно выделяемых в отдельный класс;

— информационных систем от многочисленных компьютерных вирусов, способных не только разрушить информацию, но иногда и повредить технические компоненты системы (такие, как BIOS).

Активно развиваются также средства защиты от утечки информации по цепям питания, каналам электромагнитного излучения компьютера или монитора (применяется экранирование помещений, использование генераторов шумовых излучений, специальный подбор мониторов и комплектующих компьютера, обладаю-

щих наименьшим излучением), средства защиты от электронных «жучков», устанавливаемых непосредственно в комплектующие компьютера, и т. д.

Защита от несанкционированного доступа к ресурсам компьютера — это комплексная проблема, подразумевающая решение следующих вопросов:

- присвоение пользователю, а равно и терминалам, программам, файлам и каналам связи уникальных имен и кодов (идентификаторов);

- выполнение процедур установления подлинности при обращениях (доступе) к информационной системе и запрашиваемой информации, т. е. проверка того, что лицо или устройство, сообщившее идентификатор, в действительности ему соответствует. Уже используются и аппаратно-программные системы биометрической идентификации пользователей (мышь и клавиатуры с функцией дактилоскопической идентификации, системы опознавания пользователя по голосу, по видеоизображению, в том числе по сетчатке и радужной оболочке глаз и т. п.);

- проверку полномочий, т. е. проверку права пользователя на доступ к системе или запрашиваемым данным (на выполнение над ними определенных операций — чтение, обновление) в целях разграничения прав доступа к сетевым и компьютерным ресурсам;

- автоматическую регистрацию в специальном журнале всех как удовлетворенных, так и отвергнутых запросов к информационным ресурсам с указанием идентификатора пользователя, терминала, времени и сущности запроса, то есть ведение журналов аудита, позволяющих определить, через какой хост-компьютер действовал хакер, а иногда и определить его IP-адрес и точное местоположение. В литературе имеются рекомендации по количественной оценке параметров систем защиты информации.

Полномочия (возможности) пользователя в системе определяются набором его прав. Права пользователей бывают стандартные и расширенные. К стандартным относятся такие права, как возможность изменять системное время, выполнять резервное копирование файлов, загружать драйверы устройств, изменять системную конфигурацию, выполнять выключение сервера и т. п. Расширенные права во многом являются специфичными для операционной системы или приложений. Механизмы защиты Windows позволяют гибко ограничивать права пользователей или предоставлять им доступ к любым ресурсам системы.

Вопросам информационной безопасности сейчас уделяется огромное внимание, существует множество публикаций по этой тематике, посвященных различным аспектам и прикладным вопросам защиты информации, на международном и государствен-

ном уровнях приняты законы по обеспечению безопасности информации.

## 1.7. Мировые информационные ресурсы

К мировым информационным ресурсам относятся информационные системы, действующие на международном, глобальном уровне. На мировом рынке информации и информационных услуг можно выделить следующие основные секторы, характерные для всех развитых стран, в том числе и для России.

*Сектор деловой информации* (биржевой, финансовой, коммерческой, экономической, статистической) включает в себя:

- биржевую и финансовую информацию — информацию о котировках ценных бумаг, валютных курсах, учетных ставках, рынке товаров и капиталов, инвестициях, ценах, предоставляемую биржами, специальными службами биржевой и финансовой информации, брокерскими компаниями, банками;

- экономическую и социальную статистическую информацию — числовую экономическую, демографическую, социальную информацию в виде прогнозных оценок, предоставляемую государственными статистическими службами;

- коммерческую информацию — информацию по компаниям, фирмам, корпорациям, направлениям работы и их продукции, ценам, финансовому состоянию, связям, сделкам, руководителям и т. п.;

- деловые новости в области экономики и бизнеса, предоставляемые специальными информационными службами.

*Сектор информации для специалистов* (научно-технической и специальной) охватывает:

- профессиональную информацию — специальные данные и информацию для юристов, врачей, фармацевтов, преподавателей, инженеров и т. п.;

- научно-техническую информацию — документальную и реферативную, справочную информацию и данные в области фундаментальных и прикладных, естественных, технических и общественных наук, отраслей производства и сфер человеческой деятельности;

- услуги организации доступа к первоисточникам (в том числе в виде копий документов) — через библиотеки и специализированные службы.

*Сектор массовой и потребительской информации* (новости, услуги на основе современных средств глобальной телекоммуникации) охватывает:

– новости и литературу — информацию служб новостей и агентств прессы, электронные журналы, справочники, энциклопедии;

– потребительскую и развлекательную информацию, ориентированную на домашнее, а не служебное использование — местные новости, погоду, расписания движения транспорта, игры, предложения по обмену, покупкам и продажам, справочники отелей и ресторанов, информацию по обмену валюты, аренде машин, турам, дачам для аренды.

*Сектор электронных сделок* представлен в основном системами банковских карт, системами резервирования билетов и мест в гостиницах, заказа товаров и услуг, а также биржевых, банковских и расчетных операций. Интернет привлек в эти области массового потребителя, а также все большее число ранее замкнутых систем, которые начали использовать эту сеть для совершения электронных сделок, несмотря на все еще сохраняющиеся явно недостаточные возможности защиты информации. Поэтому Интернет быстро превращает данный рынок в массовый, в быстро развивающуюся новую отрасль электронной торговли. На развитии рынке электронных сделок имеются системы финансовых (валютных и фондовых), банковских и расчетных операций, специальные платежные системы для обслуживания сделок в Интернете, службы резервирования билетов и мест в самолетах, поездах, автобусах, гостиницах, кемпингах, заказа товаров и услуг и т. п.

На рынке электронной глобальной коммуникации можно выделить различные системы на основе современных средств связи и человеческого общения — коммерческие и публичные сети передачи данных, системы электронной почты, коммерческие диалоговые системы, объединяющие владельцев ПК, телеконференции, электронные сетевые доски объявлений и бюллетени и т. п. Интернет вошел на мировой информационный рынок именно через эту область и долго существовал и развивался параллельно с другими службами связи и компьютерными сетями.

Появление интернет-технологий, позволяющих выстраивать деловые отношения в среде Интернет, дает возможность говорить о возникновении нового типа экономики, которая может быть названа «интернет-экономикой».

### **Контрольные вопросы**

1. Назовите основные этапы развития ИС.
2. В чем различие между понятиями «информационная технология» и «информационная система»?
3. Какие процессы обеспечивают работу информационной системы?

4. Что входит в функциональную часть информационных систем?
5. Назовите виды обеспечения информационных систем.
6. По каким признакам можно классифицировать информационные системы?
7. В чем различие между управляющими и советующими информационными системами?
8. Каковы достоинства архитектуры клиент-сервер по сравнению с файл-серверной архитектурой?
9. В чем особенность трехуровневой архитектуры информационных систем?
10. Перечислите достоинства и недостатки интранет-технологии.
11. Что включают в себя мировые информационные ресурсы?
12. Назовите показатели эффективности информационных систем.
13. Что может угрожать корпоративной информационной системе?
14. Как обеспечить безопасность информации в информационной системе?
15. Какие существуют средства и методы обеспечения информационной безопасности корпоративных информационных систем?

# ЭКСПЕРТНЫЕ СИСТЕМЫ

### 2.1. Понятие искусственного интеллекта. Информационная технология экспертных систем

Термин интеллект (intelligence) происходит от латинского intellectus, что означает ум, рассудок, разум, мыслительные способности человека. Соответственно искусственный интеллект ИИ (artificial intelligence, AI) можно толковать как свойство технических систем брать на себя отдельные функции интеллекта человека, например выбирать и принимать оптимальные решения на основе ранее полученного опыта и рационального анализа внешних воздействий. Большинство из нас уверены, что смогут отличить «разумное» поведение от «неразумного». Однако вряд ли кто-нибудь сможет дать достаточно полное определение искусственному интеллекту, отражающее сложность человеческого разума.

Проблема определения искусственного интеллекта сводится к проблеме определения интеллекта вообще. Является ли он чем-то единым или же этот термин объединяет набор разрозненных способностей? Можно ли судить о наличии интеллекта только по наблюдаемому поведению или же требуется свидетельство наличия некоторого скрытого механизма? Как представляются знания в нервных тканях живых существ и как можно применить это в проектировании интеллектуальных устройств? И более того, необходимо ли создавать интеллектуальную компьютерную программу по образу и подобию человеческого разума или же достаточно строго инженерного подхода? Возможно ли вообще достичь разумности посредством компьютерной техники или же сущность интеллекта требует богатства чувств и опыта, присущего лишь человеку?

На эти вопросы ответа пока не найдено, но все они помогают сформировать задачи и методологию, составляющие основу современного ИИ. Отчасти привлекательность искусственного интеллекта в том и состоит, что он является оригинальным и мощным орудием для исследования именно этих проблем.

Исторически сложились три основных подхода в моделировании ИИ.

В рамках первого подхода объектом исследований являются структура и механизмы работы мозга человека, а конечная цель заключается в раскрытии тайн мышления. Необходимыми этапами исследований в этом направлении являются построение моделей на основе психофизиологических данных, проведение экспериментов с ними, выдвижение новых гипотез относительно механизмов интеллектуальной деятельности, совершенствование моделей и т. д.

Второй подход предполагает моделирование интеллектуальной деятельности с помощью ЭВМ. Цель работ в этом направлении — создание алгоритмического и программного обеспечения, позволяющего решать интеллектуальные задачи не хуже человека.

Наконец, третий подход ориентирован на создание смешанных человеко-машинных, или, как еще говорят, интерактивных интеллектуальных систем, на симбиоз возможностей естественного и искусственного интеллекта. Важнейшими проблемами в этих исследованиях являются оптимальное распределение функций между естественным и искусственным интеллектом и организация диалога между человеком и машиной.

В начале 80-х годов прошлого столетия в исследованиях по искусственному интеллекту сформировалось самостоятельное направление, получившее название экспертные системы (ЭС). Цель исследований по ЭС состоит в разработке программ, которые при решении задач, трудных для эксперта-человека, получают результаты, не уступающие по качеству и эффективности решениям, получаемым экспертом. Стратегии экспертных систем основаны на знаниях эксперта.

Экспертные системы — это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и распространяющие этот опыт для консультаций менее квалифицированных пользователей. Программные средства, базирующиеся на технологии экспертных систем или инженерии знаний, получили значительное распространение в мире. Технология экспертных систем позволяет пользователю принимать решения, превосходящие его возможности. По мнению ведущих специалистов, в недалекой перспективе ЭС будут играть ведущую роль во всех фазах проектирования, разработки, производства, распределения, продажи, поддержки и оказания услуг. Технология экспертных систем, получившая коммерческое распространение, обеспечит революционный прорыв в интеграции приложений из готовых интеллектуально взаимодействующих модулей.

Экспертные системы предназначены для так называемых неформализованных задач, при этом они не отвергают и не заменя-



ют традиционного подхода к разработке программ, ориентированного на решение формализованных задач.

Неформализованные задачи обычно обладают следующими особенностями:

- ошибочностью, неоднозначностью, неполнотой и противоречивостью исходных данных;

- ошибочностью, неоднозначностью, неполнотой и противоречивостью знаний о проблемной области и решаемой задаче;

- большой размерностью пространства решения, т. е. перебор при поиске решения весьма велик;

- динамически изменяющимися данными и знаниями.

Следует подчеркнуть, что неформализованные задачи представляют большой и очень важный класс задач. Многие специалисты считают, что эти задачи являются наиболее массовым классом задач, решаемых ЭВМ.

Решение специальных задач требует специальных знаний. Главная идея использования технологии экспертных систем заключается в том, чтобы получить от эксперта его знания и, загрузив их в память компьютера, использовать всякий раз, когда в этом возникнет необходимость. Являясь одним из основных приложений искусственного интеллекта, экспертные системы представляют собой компьютерные программы, трансформирующие опыт экспертов в какой-либо области знаний в форму эвристических правил. **Эвристика** представляет собой некоторое знание, приобретенное человеком по мере накопления практического опыта решения аналогичных проблем. Эвристики не гарантируют получения оптимального результата с такой же уверенностью, как обычные алгоритмы, используемые для решения задач в рамках технологии поддержки принятия решений. Однако часто они дают в достаточной степени приемлемые решения для их практического использования. Все это делает возможным использовать технологию экспертных систем в качестве советующих систем.

Экспертные системы имеют много особенностей. Они применяются для решения только трудных практических задач. По качеству и эффективности решения экспертные системы не уступают (не должны уступать) решениям специалиста. Кроме того, решения экспертных систем обладают «прозрачностью», т. е. могут быть объяснены пользователю на качественном уровне. Экспертные системы способны пополнять свои знания в ходе взаимодействия с экспертом. Экспертные системы и системы искусственного интеллекта отличаются от систем обработки данных еще и тем, что в них в основном используют символьный (а не числовой) способ представления данных, символьный вывод и эвристический поиск решения (а не исполнение известного алгоритма).

Экспертное знание — это сочетание теоретического понимания проблемы и набора эвристических правил для ее решения, которые эффективны в данной предметной области. Большинство экспертных систем были написаны для специализированных предметных областей, эти области довольно хорошо изучены и располагают четко определенными стратегиями принятия решений. Несмотря на воодушевляющие перспективы экспертных систем, было бы ошибкой переоценивать возможности этой технологии. Основные проблемы таковы:

- трудности в передаче «глубоких» знаний предметной области;

- недостаток гибкости. Если людей поставить перед задачей, которую они не в состоянии решить немедленно, то они обычно сначала исследуют основные принципы и вырабатывают какую-то стратегию для перехода к решению проблемы. Экспертным системам этой способности не хватает;

- трудности в предоставлении аргументированных объяснений. Обычно ограничиваются описанием шагов, которые были приняты в поиске решения;

- трудности в тестировании. Хотя обоснование корректности любой большой компьютерной системы достаточно трудоемко, экспертные системы проверять особенно тяжело;

- ограниченные возможности обучения на опыте.

Очевидное решение этих проблем — «заставить» программы учиться самим на опыте, аналогах или примерах. Несмотря на эти ограничения, экспертные системы доказали свою ценность во многих важных областях.

Типичная экспертная система состоит из следующих основных компонентов: интерфейс пользователя, база знаний, интерпретатор, модуль создания системы, подсистема объяснений (рис. 2.1).

*Интерфейс пользователя* — комплекс программ, реализующих диалог пользователя с экспертной системой как на стадии ввода информации, так и на стадии получения результатов.

*База знаний* — ядро экспертной системы, 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 БД в машинном представлении.

*Интерпретатор* — программа, моделирующая ход рассуждений эксперта на основании знаний, имеющихся в БЗ (его иногда называют решателем, дедуктивной машиной или блоком логического вывода).

*Модуль создания системы* — это программный модуль, предназначенный для преобразования данных и правил, полученных

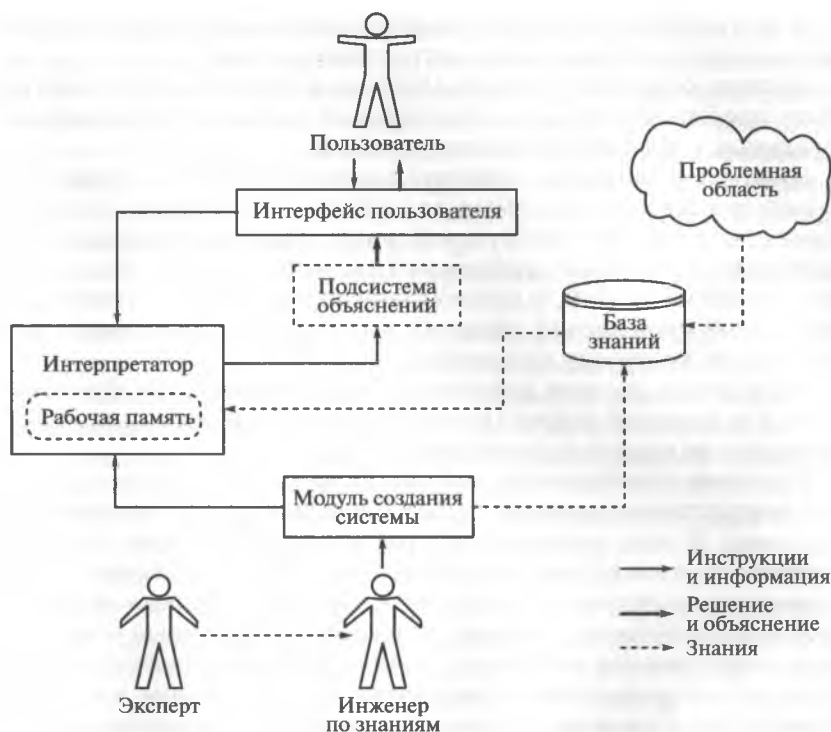


Рис. 2.1. Основные компоненты информационной технологии экспертных систем

от инженера по знаниям, в форму, пригодную для использования их в программе.

*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 в цепочке решений. Ответ на вопрос «почему» — ссылка на решение, непосредственно предшествовавшее полученному решению, т.е. отход на один шаг назад.

В разработке экспертных систем участвуют представители следующих специальностей:

- эксперт в проблемной области, задачи которой будет решать экспертная система;
- инженер по знаниям — специалист по разработке экспертных систем (используемые им технологии, методы называют технологией (методами) инженерии знаний);

– программист по разработке инструментальных средств, предназначенных для разработки экспертных систем.

Эксперт определяет знания (данные и правила), характеризующие проблемную область, обеспечивает полноту и правильность введенных в экспертную систему знаний.

Инженер по знаниям помогает эксперту выявить и структурировать знания, необходимые для работы экспертной системы, осуществляет выбор инструментального средства, наиболее подходящего для данной проблемной области, определяет способ представления знаний, выделяет стандартные функции (типичные для данной проблемной области), которые будут использоваться в правилах, вводимых экспертом.

Экспертная система работает в двух режимах: приобретения знаний и решения задачи (называемом также режимом консультации или режимом использования ЭС).

В режиме приобретения знаний общение с экспертной системой осуществляет эксперт (через посредничество инженера по знаниям). В этом режиме эксперт, используя модуль создания системы, наполняет систему знаниями, которые позволяют ЭС в режиме решения самостоятельно (без эксперта) решать задачи из проблемной области. Эксперт описывает проблемную область в виде совокупности данных и правил. Данные определяют объекты, их характеристики и значения, существующие в области экспертизы. Правила определяют способы манипулирования с данными, характерные для рассматриваемой области.

Отметим, что режиму приобретения знаний в традиционном подходе к разработке программ соответствуют этапы алгоритмизации, программирования и отладки, выполняемые программистом. В случае экспертной системы, в отличие от традиционного подхода, программы разрабатывает не программист, а эксперт, не владеющий программированием.

В режиме консультации общение с ЭС осуществляет конечный пользователь, которого интересует результат и (или) способ его получения. Необходимо отметить, что в зависимости от назначения экспертной системы пользователь может не быть специалистом в данной проблемной области. В этом случае он обращается к ЭС за результатом. Если пользователь — специалист, то в этом случае он может сам получить результат, но он обращается к ЭС с целью либо ускорить процесс получения результата, либо возложить на ЭС рутинную работу. В режиме консультации данные о задаче пользователя после обработки их диалоговым компонентом поступают в рабочую память. Интерпретатор на основе входных данных из рабочей памяти, общих данных о проблемной области и правил из базы знаний формирует решение задачи.

Необходимо отметить, что в настоящее время технология экспертных систем используется для решения различных типов задач (интерпретация, предсказание, диагностика, планирование, конструирование, контроль, отладка, инструктаж, управление) в самых разнообразных проблемных областях, таких как финансы, нефтяная и газовая промышленность, энергетика, транспорт, фармацевтическое производство, космос, металлургия, образование, телекоммуникации и связь и др.

Ниже приведены примеры крупномасштабных экспертных систем.

**MICIN** — экспертная система для медицинской диагностики. Разработана группой по инфекционным заболеваниям Стенфордского университета. Ставит соответствующий диагноз исходя из представленных ей симптомов и рекомендует курс медикаментозного лечения любой из диагностированных инфекций. База данных состоит из нескольких сотен правил.

**PUFF** — анализ нарушения дыхания. Данная система представляет собой MICIN, из которой удалили данные по инфекциям и вставили данные о легочных заболеваниях.

**DENDRAL** — распознавание химических структур. Данная система старейшая из имеющих звание экспертных. Первые версии данной системы появились еще в 1965 г. все в том же Стенфордском университете. Пользователь дает системе DENDRAL некоторую информацию о веществе, а также данные спектроскопии (инфракрасной, ядерного магнитного резонанса и масс-спектрометрии), и та в свою очередь выдает диагноз в виде соответствующей химической структуры.

**PROSPECTOR** — экспертная система, созданная для содействия поиску коммерчески оправданных месторождений полезных ископаемых.

## 2.2. Смысл экспертного анализа

Способность выполнить экспертный анализ — это не только наличие определенных знаний и уровня квалификации. Для этого нужно обладать и очень специфическими навыками и умением разобраться в конкретной ситуации в данной предметной области.

Ранее было сказано, что экспертная система должна обладать знаниями. Просто способность выполнять некоторый алгоритм, например производить анализ списка элементов на наличие какого-либо свойства, явно не отвечает этому требованию. Это все равно, что дать первому случайному прохожему список вопросов и ответов и ожидать от него успешного выполнения поиска и

устранения неисправностей в системах определенного типа. Раньше или позже, но он обязательно столкнется с ситуацией, не предусмотренной в том списке, которым его снабдили.

Знания, которыми обладает программа, должны быть сконцентрированы в определенной предметной области. Случайный набор имен, дат и мест событий, сентенций из классиков и т. п. — это отнюдь не те знания, которые могут послужить основой для программы, претендующей на способность выполнить экспертный анализ. Знания предполагают определенную организацию и интеграцию, т. е. отдельные сведения должны соотноситься друг с другом и образовывать нечто вроде цепочки, в которой одно звено «тянет» за собой следующее (причинно-следственная связь). Недостаточно получить доступ к оперативной документации — необходимо получить в свое распоряжение специалиста (или программу), способного справиться с возникшими проблемами.

Экспертная система может полностью взять на себя функции, выполнение которых обычно требует привлечения опыта специалиста, или играть роль ассистента для человека, принимающего решение. Другими словами, система (техническая или социальная), требующая принятия решения, может получить его непосредственно от программы или через промежуточное звено — человека, который общается с программой. Тот, кто принимает решение, может быть экспертом со своими собственными правами, и в этом случае программа может оправдать свое существование, повышая эффективность его работы. Альтернативный вариант — человек, работающий в сотрудничестве с такой программой, может добиться с ее помощью результатов более высокого качества. Правильное распределение функций между человеком и машиной является одним из ключевых условий высокой эффективности внедрения экспертных систем.

Перечень типовых задач, решаемых экспертными системами, включает:

- извлечение информации из первичных данных (таких, как сигналы, поступающие от гидролокатора);
- диагностика неисправностей (как в технических системах, так и в человеческом организме);
- структурный анализ сложных объектов (например, химических соединений);
- выбор конфигурации сложных многокомпонентных систем (например, распределенных компьютерных систем);
- планирование последовательности выполнения операций, приводящих к заданной цели (например, выполняемых промышленными роботами).

Хотя известны и «обычные» программы, специализирующиеся на определенных задачах из представленного перечня (или ана-

логичных им в смежных областях), далее мы покажем, в чем состоит существенная разница между «обычным» подходом и предлагаемым в сфере искусственного интеллекта и почему экспертные системы можно выделить в отдельный, достаточно хорошо различимый класс программ. Четкого формального определения экспертной системы, которое всех бы удовлетворило, не существует — приведенное выше тоже довольно расплывчато. Но тем не менее существует довольно много важных признаков, присущих в той или иной степени всем экспертным системам.

### **2.3. Характеристики экспертных систем**

Прежде всего нужно отметить, что экспертная система моделирует не столько физическую (или иную) природу определенной проблемной области, сколько механизм мышления человека применительно к решению задач в этой проблемной области. Это существенно отличает экспертные системы от систем математического моделирования или компьютерной анимации. Нельзя, конечно, сказать, что программа полностью воспроизводит психологическую модель специалиста в этой предметной области (эксперта), но важно, что основное внимание все-таки уделяется воспроизведению компьютерными средствами методики решения проблем, применяемой экспертом, т. е. выполнению некоторой части задач так же (или иногда даже лучше), как это делает эксперт.

Система помимо выполнения вычислительных операций формирует определенные соображения и выводы, основываясь на тех знаниях, которыми она располагает. Знания в системе представлены, как правило, на некотором специальном языке и хранятся отдельно от собственно программного кода, который и формирует выводы и соображения.

При решении задач экспертными системами основными являются эвристические и приближенные методы, которые в отличие от алгоритмических не всегда гарантируют успех. Такие методы являются приблизительными в том смысле, что они не требуют исчерпывающей исходной информации, и существует определенная степень уверенности (или неуверенности) в том, что предлагаемое решение является верным.

Одна из основных характеристик экспертной системы — ее производительность, т. е. скорость получения результата и его достоверность, или надежность. Экспертная система должна за достаточно короткое время найти решение не хуже того, которое может предложить специалист данной предметной области.

Экспертная система должна обладать способностью обосновать принятие именно такого решения и доказать его. ЭС проектиру-

ется в расчете на взаимодействие с разными пользователями, для которых ее работа должна быть по возможности «прозрачной». Это особенно необходимо в областях, характеризующихся неопределенностью, неточностью информации (например, в медицинской диагностике). В этих случаях способность к объяснению нужна для того, чтобы повысить степень доверия пользователя к советам системы, а также для того, чтобы дать возможность пользователю обнаружить возможный дефект в рассуждениях системы. Пользователь должен получить всю необходимую информацию, чтобы быть уверенным, что решение принято «не с потолка». В связи с этим в экспертных системах следует предусматривать дружественное взаимодействие с пользователем. Кроме того, отсутствие достаточной прозрачности поведения системы не позволит эксперту повлиять на ее производительность или дать совет, как можно ее повысить. Прослеживание и оценка поведения системы — задача довольно сложная и для ее решения необходимы совместные усилия эксперта и специалиста по информатике.

Экспертные системы должны решать задачи, требующие экспертных знаний в некоторой конкретной области. В той или иной форме экспертные системы должны обладать этими знаниями. Зачастую термин «система», основанная на знаниях, и используется в качестве синонима термина «экспертная система», хотя, строго говоря, экспертная система — это более широкое понятие. Система, основанная на знаниях, — это любая система, процесс работы которой основан на применении правил отношений к символическому представлению знаний, а не на использовании алгоритмических или статистических методов. Таким образом, программа, способная рассуждать о погоде, будет системой, основанной на знаниях, даже в том случае, если она не способна выполнить метеорологическую экспертизу. А вот чтобы иметь право называться метеорологической экспертной системой, программа должна давать прогноз погоды (достоверность прогноза — это другой вопрос).

От других видов программ из области искусственного интеллекта экспертные системы отличаются тем, что имеют ярко выраженную практическую направленность в научной и коммерческой областях.

Суммируя все сказанное, отметим, что экспертная система содержит знания в определенной предметной области, накопленные в результате практической деятельности человека, и использует их для решения специфичных для этой области проблем. Этим экспертные системы отличаются от прочих, «традиционных», систем, в которых предпочтение отдается более общим и менее связанным с предметной областью теоретическим методам, чаще всего математическим.



## 2.4. Функции экспертных систем

В самом общем случае, для того чтобы построить ЭС, необходимо разработать механизмы выполнения следующих функций:

- 1) приобретение знаний в конкретной предметной области;
- 2) представление знаний;
- 3) решение задач с использованием знаний о конкретной предметной области (управление процессом решения);
- 4) объяснение намерений и решений системы во время и после окончания процесса решения задачи (разъяснение принятого решения).

Каждая из этих функций может оказаться очень сложной и зависит от прикладной области, а также от различных практических требований. В процессе разработки и реализации могут возникнуть разнообразные проблемы.

### 2.4.1. Приобретение знаний

Приобретение знаний — это передача потенциального опыта решения проблемы от некоторого источника знаний и преобразование его в вид, позволяющий использовать эти знания в программе.

Передача знаний выполняется в процессе достаточно длительных собеседований между специалистом по проектированию экспертной системы (инженером по знаниям) и экспертом в определенной предметной области, способным четко сформулировать имеющийся у него опыт. Многие исследователи рассматривают функцию приобретения знаний в качестве одного из главных «узких мест» технологии экспертных систем. Проблемы могут возникнуть самые разнообразные.

Специалисты в узкой области, как правило, пользуются собственным профессиональным жаргоном, который трудно перевести на обычный язык. Но смысл жаргонных слов (профессионализмов) совсем не очевиден, а потому требуется много дополнительных вопросов для уточнения их логического значения. Неудовлетворительные результаты собеседований эксперта и инженера по знаниям ввиду использования первым профессионализмов пробудили у некоторых исследователей интерес к автоматизации процесса передачи знаний специалистом машине. Одно из направлений исследований в этой области — автоматизированное извлечение знаний — появилось как побочный продукт в развитии систем человеко-машинного диалога. Другое направление — машинное обучение. Идея состоит в том, чтобы машина училась решать проблемы примерно так же, как учится человек.

Проблемой является еще и то, что факты и принципы, лежащие в основе многих специфических областей знания эксперта, не могут быть четко сформулированы в терминах математической теории или детерминированной модели, свойства которой хорошо понятны. Так, эксперту может быть известно, что определенное событие может наступить, но он ничего не сможет сказать о механизмах, которые приводят к наступлению этого события. Для того чтобы решить проблему в определенной области, эксперту недостаточно просто обладать суммой знаний о фактах и принципах в этой области. Например, опытный специалист знает, какого рода информацией нужно располагать для формулировки того или иного суждения, насколько надежны различные источники информации и как можно расчленить сложную проблему на несколько простых, которые можно решать более или менее независимо. Выявить в процессе собеседования такого рода знания, основанные на личном опыте и плохо поддающиеся формализации, значительно сложнее, чем получить простой перечень каких-то фактов или общих принципов.

Экспертный анализ даже в очень узкой области, выполняемый человеком, очень часто нужно поместить в довольно обширный контекст, который включает и многие вещи, кажущиеся эксперту само собой разумеющимися, но для постороннего отнюдь таковыми не являющиеся.

#### **2.4.2. Представление знаний**

Представление знаний — это отдельная область исследований, тесно связанная с философией формализма и когнитивной психологией. Предмет исследования в этой области — методы ассоциативного хранения информации, подобные тем, которые существуют в мозгу человека. При этом основное внимание, естественно, уделяется логической, а не биологической стороне процесса, опуская подробности физических преобразований.

База знаний — наиболее важная компонента экспертной системы, на которой основаны ее «интеллектуальные способности». В отличие от всех остальных компонент экспертной системы база знаний — изменяемая часть системы, которая может пополняться и модифицироваться инженерами по знаниям. Существует несколько способов представления знаний в экспертной системе, однако общим для всех них является то, что знания представлены в символьной форме (элементарными компонентами представления знаний являются тексты, списки и другие символьные структуры). Таким образом, в экспертной системе реализуется принцип символьной природы рассуждений, который заключается в том,

что процесс рассуждения представляется как последовательность символьных преобразований.

Наиболее распространенный способ представления знаний — в виде конкретных фактов и правил, по которым из имеющихся фактов могут быть выведены новые. Факты могут быть представлены, например, в виде троек: атрибут, объект, значение.

Такой факт означает, что заданный объект имеет заданный атрибут (свойства) с заданным значением. Например, тройка (возраст, пациент, 77) представляет факт, что возраст пациента — 77 лет. В более простых случаях факт выражается не конкретным значением атрибута, а каким-либо простым утверждением, которое может быть *истинным* или *ложным*, например: «Пациент курит». В таких случаях факт можно обозначить каким-либо кратким именем (например, *курение*) или использовать для представления факта сам текст соответствующей фразы.

Правила в базе знаний имеют вид:

*ЕСЛИ  $a$  ТО  $s$ ,*

где  $a$  — условие;  $s$  — действие.

Действие  $s$  выполняется, если  $a$  истинно. Наиболее часто действие  $s$ , так же, как и условие, представляет собой утверждение, которое может быть выведено системой (т.е. становится ей известной), если истинно условие правила  $a$ . Правила в базе знаний служат для представления эвристических знаний (эвристик), т.е. неформальных правил рассуждения, вырабатываемых экспертом на основе опыта его деятельности. Простой пример правила:

*ЕСЛИ* «Пациент курит»

*ТО* «У пациента больные легкие».

В качестве условия  $a$  может выступать либо факт (как в данном примере), либо несколько фактов  $a_1, \dots, a_n$ , соединенные логическими операциями.

Пример предыдущего правила с более сложным условием (правило 1):

*ЕСЛИ*

(«Пациент курит») *И* («Кровяное давление выше среднего»)

*ТО*

«Риск высокий» (правило 1)

Действия, входящие в состав правил, могут содержать новые факты. При применении таких правил эти факты становятся известны системе, т.е. включаются в множество фактов, которое называется *рабочим множеством*. Например, если факты «Пациент курит» и «Кровяное давление выше среднего» уже имеются в рабочем множестве, то после применения приведенного выше правила в него также включается факт «Риск высокий».

Если система не может вывести некоторый факт, истинность или ложность которого требуется установить, то система спрашивает о нем пользователя. Например:

*Верно ли, что «Пациент курит»?*

При получении положительного ответа от пользователя факт «Пациент курит» включается в рабочее множество.

Существуют статические и динамические базы знаний. Статические базы данных не изменяются со временем. Динамические базы знаний изменяются со временем. Новые факты, добавляемые в базу знаний, являются результатом вывода, который состоит в применении правил к имеющимся фактам.

Экспертные системы могут иметь монотонный и немонотонный выводы.

В системах с монотонным выводом факты, хранимые в базе знаний, статичны, т. е. не изменяются в процессе решения задачи. В системах с немонотонным выводом допускается изменение или удаление фактов из базы знаний. Изменение фактов в свою очередь приводит к необходимости удаления из базы знаний заключений, полученных с помощью упомянутых правил. Тем самым вывод выполняется повторно для того, чтобы пересмотреть решения, полученные на основе подвергшихся изменению фактов.

### **2.4.3. Решение задач с использованием знаний**

При проектировании экспертной системы серьезное внимание должно быть уделено и тому, как осуществляется доступ к знаниям и как они используются при поиске решения. Знание о том, какие знания нужны в той или иной конкретной ситуации, и умение ими распорядиться — важная часть процесса функционирования экспертной системы. Такие знания получили наименование *метазнаний*, т. е. знаний о знаниях. Решение нетривиальных проблем требует и определенного уровня планирования и управления при выборе, какой вопрос нужно задать, какой тест выполнить и т. д.

Использование разных стратегий перебора имеющихся знаний, как правило, довольно существенно влияет на характеристики эффективности программы. Эти стратегии определяют, каким способом программа отыскивает решение проблемы в некотором пространстве альтернатив.

Существует такая программная компонента экспертных систем, как *подсистема вывода*, реализующая процесс рассуждений системы на основе базы знаний и рабочего множества. Она выполняет две функции:

1) просмотр существующих фактов из рабочего множества и правил из базы знаний и добавление (по мере возможности) в рабочее множество новых фактов;

2) определение порядка просмотра и применения правил. Эта подсистема управляет процессом консультации, сохраняет для пользователя информацию о полученных заключениях и запрашивает у него информацию, когда для срабатывания очередного правила в рабочем множестве оказывается недостаточно данных.

Цель экспертной системы — вывести некоторый заданный факт, который называется *целевым утверждением*, т. е. в результате применения правил добиться того, чтобы этот факт был включен в рабочее множество, либо опровергнуть этот факт, т. е. убедиться, что его вывести невозможно, следовательно, при данном уровне знаний системы он является ложным. Целевое утверждение либо «заложено» заранее в базу знаний системы, либо извлекается системой из диалога с пользователем.

Работа системы представляет собой последовательность шагов, на каждом из которых система выбирает из базы некоторое правило, которое можно применить к текущему содержимому рабочего множества. Цикл заканчивается, когда выведено или опровергнуто целевое утверждение. Цикл работы экспертной системы иначе называется *логическим выводом*. Логический вывод может происходить многими способами, из которых наиболее распространенные — прямой и обратный порядок вывода.

Прямой порядок вывода — от фактов, которые находятся в рабочем множестве, к заключению. Если такое заключение удастся найти, то оно заносится в рабочее множество. Прямой вывод часто называют выводом, управляемым данными.

Для иллюстрации добавим к нашему примеру базы знаний о здоровье еще одно правило (правило 2):

*ЕСЛИ*

«Риск высокий»

*ТО*

«Продолжительность жизни менее 60 лет».

Предположим также, что факты «Пациент курит» и «Кровяное давление выше среднего» имеются в рабочем множестве, а целью системы является определение наиболее вероятной продолжительности жизни пациента (пользователя).

При прямом выводе работа системы будет протекать следующим образом.

1. Рассматривается правило 1. Его условие истинно, так как оба элемента конъюнкции имеются в рабочем множестве. Применяем правило 1. Добавляем к рабочему множеству факт «Риск высокий».

2. Рассматривается правило 2. Его условие истинно, так как утверждение из условия имеется в рабочем множестве. Примеряем правило 2. Добавляем к рабочему множеству факт «Продолжительность жизни менее 60 лет». Целевое утверждение выведено.

При обратном порядке вывода заключения просматриваются до тех пор, пока не будут обнаружены в рабочей памяти или получены от пользователя факты, подтверждающие одно из них. В системах с обратным выводом вначале выдвигается некоторая гипотеза, а затем механизм вывода в процессе работы как бы возвращается назад, переходя от нее к фактам, и пытается найти среди них те, которые подтверждают эту гипотезу. Если достоверность доказана, то выбирается следующая гипотеза, детализирующая первую и являющаяся по отношению к ней подцелью. Далее отыскиваются факты, подтверждающие истинность подчиненной гипотезы. Вывод такого типа называется *управляемым целями*. Обратный поиск применяется в тех случаях, когда цели известны и их сравнительно немного.

Заметим, что для упрощения ситуации мы предположили, что в обоих случаях факты «Пациент курит» и «Кровяное давление выше среднего» уже известны системе. На самом деле система выясняет истинность или ложность факта, входящего в условие некоторого правила, спрашивая об этом пользователя в тот момент, когда она пытается применить правило.

Приведенный пример сознательно выбран простым и не отражающим многих проблем, связанных с организацией вывода в экспертной системе. В частности, из примера может создаться впечатление, что прямая цепочка рассуждений эффективнее, чем обратная, что на самом деле не так. Эффективность той или иной стратегии вывода зависит от характера задачи и содержимого базы знаний. В системах диагностики чаще применяется прямой вывод, в то время как в планирующих системах более эффективным оказывается обратный вывод. В некоторых системах вывод основывается на сочетании обратного и ограниченно-прямого. Такой комбинированный метод получил название *циклического*.

### Контрольные вопросы

1. Что такое искусственный интеллект?
2. Перечислите основные направления в моделировании ИИ.
3. Разъясните понятия «данные» и «знания».
4. Что называется экспертной системой?
5. Назовите основные проблемы технологии экспертных систем.
6. Какие особенности имеют экспертные системы?
7. Назовите основные компоненты типичной экспертной системы.
8. Приведите примеры крупномасштабных экспертных систем.

9. В чем смысл экспертного анализа?
10. Перечислите типовые задачи, решаемые экспертными системами.
11. Чем характеризуются экспертные системы?
12. Назовите функции экспертных систем.
13. Как осуществляется передача знаний в экспертной системе?
14. Что представляет собой база знаний? Как осуществляется в ней представление знаний?
15. Что такое подсистема вывода? Каковы ее функции?
16. Что называется целевым утверждением?
17. В чем отличие прямого порядка вывода от обратного?

# ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

### 3.1. Стадии жизненного цикла информационных систем

Понятие жизненного цикла является одним из базовых понятий методологии проектирования информационных систем. Жизненный цикл информационной системы представляет собой непрерывный процесс, начинающийся с момента принятия решения о создании информационной системы и заканчивающийся в момент полного изъятия ее из эксплуатации. Существует международный стандарт, регламентирующий жизненный цикл информационных систем, — ISO/IEC 12207. ISO расшифровывается как International Organization of Standardization (Международная организация по стандартизации), IEC — как International Electrotechnical Commission (Международная комиссия по электротехнике). Стандарт ISO/IEC 12207 определяет структуру жизненного цикла, включая процессы, действия и задачи, которые должны быть выполнены во время создания информационной системы.

Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования. Разработка информационной системы, как правило, выполняется для определенного предприятия. Особенности деятельности предприятия или предметной области его функционирования, безусловно, влияют на состав информационной системы, но в то же время структуры разных предприятий в целом похожи между собой. Каждая организация независимо от рода ее деятельности состоит из ряда подразделений, непосредственно осуществляющих тот или иной вид деятельности компании. И эта ситуация справедлива практически для всех организаций, каким бы видом деятельности они ни занимались. Любую организацию можно рассматривать как совокупность взаимодействующих элементов (подразделений), каждый из которых может иметь свою структуру. Взаимосвязи между подразделениями тоже достаточно сложны. В общем случае можно выделить три вида связей между подразделениями предприятия:



— функциональные связи — каждое подразделение выполняет определенные виды работ в рамках единого бизнес-процесса;

— информационные связи — подразделения обмениваются информацией (документами, факсами, письменными и устными распоряжениями и т. п.);

— внешние связи — некоторые подразделения взаимодействуют с внешними системами, причем их взаимодействие также может быть как информационным, так и функциональным.

Информационная система предприятия разрабатывается как некоторый проект. Многие особенности управления проектами и фазы разработки проекта (фазы жизненного цикла) являются общими, не зависящими не только от предметной области, но и от характера проекта. Каждый проект независимо от сложности и объема работ, необходимых для его выполнения, проходит в своем развитии определенные состояния. Совокупность ступеней развития от возникновения идеи до полного завершения проекта принято разделять на стадии или этапы.

В определении количества стадий и их содержания имеются некоторые отличия, поскольку эти характеристики во многом зависят от условий осуществления конкретного проекта и опыта основных участников. Тем не менее логика и основное содержание процесса разработки информационной системы почти во всех случаях являются общими.

Можно выделить следующие стадии развития информационной системы — формирование требований (концепции), проектирование, реализация, тестирование, ввод системы в эксплуатацию, эксплуатация (сопровождение проекта).

Завершается жизненный цикл информационной системы выводом ее из эксплуатации.

Для каждого этапа определяют состав и последовательность выполняемых работ, получаемые результаты, методы и средства, необходимые для выполнения работ, роли и ответственность участников и т. д. Такое формальное описание жизненного цикла информационной системы позволяет спланировать и организовать процесс коллективной разработки и обеспечить управление этим процессом. Рассмотрим каждую из стадий более подробно.

Стадия формирования требований к информационной системе является одной из важнейших, поскольку определяет успех всего проекта. На начальной стадии устанавливается область применения системы и определяются граничные условия. Для этого необходимо идентифицировать все внешние объекты, с которыми должна взаимодействовать разрабатываемая система, и определить характер этого взаимодействия на высоком уровне, т. е. идентифицировать все функциональные возможности системы и произвести описание наиболее существенных из них.

Данная стадия включает в себя следующие этапы:

1) планирование работ. Основными задачами этого этапа являются:

- определение целей разработки;
- предварительная экономическая оценка проекта;
- построение плана-графика выполнения работ;

2) проведение обследования деятельности автоматизируемого объекта, в рамках которого осуществляются:

- предварительное определение требований к будущей системе;
- определение структуры организации;
- определение целевых функций организации;
- анализ распределения функций по подразделениям и сотрудникам;
- выявление функциональных взаимодействий между подразделениями;
- анализ информационных потоков внутри подразделений и между ними;
- анализ информации, поступающей из внешних источников;
- анализ существующих средств автоматизации деятельности организации;

3) построение моделей деятельности организации на основании результатов обследования:

- модели «как есть» (as-is), отражающей существующее на момент обследования положение дел в организации и позволяющей выявить узкие места в функционировании и сформулировать предложения по улучшению ситуации (оптимизации бизнес-процессов);
- модели «как должно быть» (to-be), представляющей наиболее оптимальную технологию работы предприятия.

Каждая из моделей представляет собой совокупность функциональной и информационной моделей деятельности организации. Необходимо определить способы перехода от модели «как есть» к модели «как должно быть». Переход может быть осуществлен либо путем совершенствования существующих бизнес-процессов и технологий их обработки на основе оценки их эффективности, либо радикальным перепроектированием бизнес-процессов и технологий их обработки (реинжиниринг бизнес-процессов).

Стадия проектирования, как правило, включает определение архитектуры системы, ее функций, внешних условий функционирования, интерфейсы и распределение функций между пользователями и системой, требования к программным и информационным компонентам, состав исполнителей и сроки разработки. Проектирование осуществляется на основе моделей «как должно быть».

Границы каждой стадии определены некоторыми моментами времени, в которые необходимо принимать определенные критические решения и, следовательно, достигать определенных ключевых целей. Содержание остальных этапов совпадает с соответствующими процессами жизненного цикла и будет рассмотрено далее.

### **3.2. Модели жизненного цикла информационной системы**

Моделью жизненного цикла информационной системы будем называть некоторую структуру, определяющую последовательность осуществления процессов, действий и задач, выполняемых на протяжении жизненного цикла информационной системы, а также взаимосвязи между этими процессами, действиями и задачами.

В стандарте ISO/IEC 12207 не конкретизируются детально методы выполнения действий и решения задач, входящих в процессы жизненного цикла информационной системы, а лишь описываются структуры этих процессов. Это вполне понятно, так как регламенты стандарта являются общими для любых моделей жизненного цикла, методологий и технологий разработки. Модель же жизненного цикла зависит от специфики информационной системы и условий, в которых она создается и функционирует.

К настоящему времени наибольшее распространение получили две основные модели жизненного цикла — каскадная модель, или модель водопада (waterfall), и спиральная модель.

Каскадная модель предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.

В спиральной модели на каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка. Особое внимание уделяется начальным этапам разработки — анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования).

#### **3.2.1. Каскадная модель жизненного цикла информационной системы**

Каскадная модель демонстрирует классический подход к разработке различных систем в любых прикладных областях. Для разработки информационных систем данная модель широко ис-

пользовалась в 70-х и первой половине 80-х годов XX века. Каскадные методы проектирования хорошо описаны в зарубежной и отечественной литературе разных направлений: методических монографиях, стандартах, учебниках. Организация работ по каскадной схеме официально рекомендовалась и широко применялась в различных отраслях. Таким образом, наличие не только теоретических оснований, но и промышленных методик и стандартов, а также использование этих методов в течение десятилетий позволяет называть каскадные методы классическими.

Каскадная модель предусматривает последовательную организацию работ. При этом основной особенностью является разбиение всей разработки на этапы, причем переход с одного этапа на следующий происходит только после того, как полностью завершены все работы на предыдущем этапе. Каждый этап завершается выпуском полного комплекта документации, достаточной для того, чтобы разработка могла быть продолжена другой командой разработчиков.

За десятилетия существования каскадной модели разбиение работ на стадии и названия этих стадий менялись. Кроме того, наиболее разумные методики и стандарты избегали жесткого и однозначного приписывания определенных работ к конкретным этапам. Тем не менее все же можно выделить ряд устойчивых этапов разработки, практически не зависящих от предметной области (рис. 3.1).

На первом этапе проводится исследование проблемы, которая должна быть решена, четко формулируются все требования заказчика. Результатом, получаемым на данном этапе, является техническое задание (разработка требований), согласованное со всеми заинтересованными сторонами.



Рис. 3.1. Каскадная модель жизненного цикла ИС

На втором этапе разрабатывают проектные решения, удовлетворяющие всем требованиям, сформулированным в техническом задании. Результатом данного этапа является комплект проектной документации, содержащей все необходимые данные для реализации проекта.

Третий этап — реализация проекта. Здесь осуществляется разработка программного обеспечения (кодирование) в соответствии с проектными решениями, полученными на предыдущем этапе. Методы, используемые для реализации, не имеют принципиального значения. Результатом выполнения данного этапа является готовый программный продукт.

На четвертом этапе проводится проверка (тестирования) полученного программного обеспечения на предмет соответствия требованиям, заявленным в техническом задании. Опытная эксплуатация позволяет выявить различного рода скрытые недостатки, проявляющиеся в реальных условиях работы информационной системы.

Последний этап — сдача готового проекта, ввод его в действие. Главная задача этого этапа — документально подтвердить, что все требования заказчика выполнены в полной мере.

Этапы работ в рамках каскадной модели часто также называют частями «проектного цикла» системы. Такое название возникло потому, что этапы состоят из многих итерационных процедур уточнения требований к системе и вариантов проектных решений. Жизненный цикл самой системы существенно сложнее и длиннее. Он может включать в себя произвольное число циклов уточнения, изменения и дополнения уже принятых и реализованных проектных решений. В этих циклах происходят развитие информационной системы и модернизация отдельных ее компонентов.

**Достоинства каскадной модели.** Каскадная модель имеет ряд положительных сторон, благодаря которым она хорошо зарекомендовала себя при выполнении различного рода инженерных разработок и получила широкое распространение. Рассмотрим ее основные достоинства.

1. На каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности. На заключительных этапах также разрабатывается пользовательская документация, охватывающая все предусмотренные стандартами виды обеспечения информационной системы (организационное, методическое, информационное, программное, аппаратное).

2. Выполняемые в логичной последовательности этапы работ позволяют планировать сроки завершения и соответствующие затраты.

Каскадная модель изначально разрабатывалась для решения различного рода инженерных задач и не потеряла своего значения для прикладной области до настоящего времени. Кроме того, каскадный подход хорошо зарекомендовал себя при разработке определенных информационных систем. Имеются в виду системы, для которых в самом начале разработки можно достаточно точно и полно сформулировать все требования, с тем чтобы предоставить разработчикам свободу выбора реализации, наилучшей с технической точки зрения. К таким информационным системам, в частности, относятся сложные расчетные системы, системы реального времени.

Тем не менее, несмотря на все свои достоинства, каскадная модель имеет ряд недостатков, ограничивающих ее применение при разработке информационных систем. Причем эти недостатки делают ее либо полностью неприменимой, либо приводят к увеличению сроков разработки и стоимости проекта. В настоящее время многие неудачи программных проектов объясняются именно последовательным процессом разработки.

**Недостатки каскадной модели.** Перечень недостатков каскадной модели при ее использовании для разработки информационных систем достаточно обширен. Вначале просто перечислим их, а затем рассмотрим основные из них более подробно:

- существенная задержка в получении результатов;
- ошибки и недоработки на любом из этапов проявляются, как правило, на последующих этапах работ, что приводит к необходимости возврата назад;
- сложность параллельного ведения работ по проекту;
- чрезмерная информационная перенасыщенность каждого из этапов;
- сложность управления проектом;
- высокий уровень риска и ненадежность инвестиций.

Задержка в получении результатов обычно считается главным недостатком каскадной схемы. Данный недостаток проявляется в основном в том, что вследствие последовательного подхода к разработке согласование результатов с заинтересованными сторонами производится только после завершения очередного этапа работ. Может оказаться, что разрабатываемая информационная система не соответствует требованиям пользователей, причем такие несоответствия могут возникать на любом этапе разработки — искажения могут непреднамеренно вноситься и проектировщиками-аналитиками, и программистами, так как они могут недостаточно хорошо разбираться в тех предметных областях, для которых разрабатывают информационную систему.

Кроме того, используемые при разработке информационной системы модели автоматизируемого объекта, отвечающие крите-

риям внутренней согласованности и полноты, могут в силу различных причин устареть за время разработки (например, из-за внесения изменений в законодательство, колебания курса валют и т. п.). Это относится и к функциональной модели, и к информационной модели, и к проектам интерфейса пользователя, и к пользовательской документации.

Возврат на более ранние стадии — недостаток каскадной модели, который является одним из проявлений предыдущего недостатка. Поэтапная и последовательная работа над проектом может быть следствием того, что ошибки, допущенные на более ранних этапах, как правило, обнаруживаются только на последующих стадиях работы над проектом. Поэтому после того как ошибки проявятся, проект возвращается на предыдущий этап, перерабатывается и снова передается на следующую стадию. Это может служить причиной срыва графика работ и усложнения взаимоотношений между группами разработчиков, выполняющих отдельные этапы работы.

Самым же неприятным является то, что недоработки предыдущего этапа могут обнаруживаться не сразу на последующем этапе, а позднее (например, на стадии опытной эксплуатации могут проявиться ошибки в описании предметной области). Это означает, что часть проекта должна быть возвращена на начальный этап работы. Вообще работа может быть возвращена с любого этапа на любой предыдущий этап, поэтому в реальности каскадная схема разработки выглядит так, как показано на рис. 3.2.

Одной из причин данной ситуации является то, что в качестве экспертов, участвующих в описании предметной области, часто выступают будущие пользователи системы, которые иногда не могут четко сформулировать то, что они хотели бы получить. Кроме того, заказчики и исполнители часто неправильно пони-

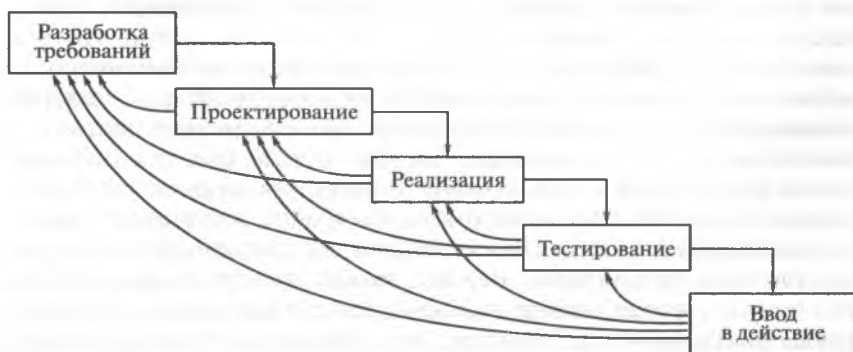


Рис. 3.2. Поэтапная модель с промежуточным контролем

мают друг друга вследствие того, что исполнители обычно не являются специалистами в предметной области решаемой задачи, а заказчики далеки от программирования.

Сложность параллельного ведения работ является также одним из недостатков каскадной модели. Отмеченные проблемы возникают вследствие того, что работа над проектом строится в виде цепочки последовательных шагов. Причем даже в том случае, когда разработку некоторых частей проекта (подсистем) можно вести параллельно, при использовании каскадной схемы распараллеливание работ весьма затруднительно. Сложности параллельного ведения работ связаны с необходимостью постоянного согласования различных частей проекта. Чем сильнее взаимозависимость отдельных частей проекта, тем чаще и тщательнее должна выполняться синхронизация, тем сильнее зависят друг от друга группы разработчиков. Поэтому преимущества параллельного ведения работ просто теряются.

Отсутствие параллелизма негативно сказывается и на организации работы всего коллектива разработчиков. Работа одних групп сдерживается другими. Пока производится анализ предметной области, проектировщики, разработчики и те, кто занимается тестированием и администрированием, почти не загружены. Кроме того, при последовательной разработке крайне сложно внести изменения в проект после завершения этапа и передачи проекта на следующую стадию. Например, если после передачи проекта на следующий этап группа разработчиков нашла более эффективное решение, оно не может быть использовано. Это связано с тем, что более раннее решение уже, возможно, реализовано и связано с другими частями проекта. Поэтому исключается (или, по крайней мере, существенно затрудняется) доработка проекта после его передачи на следующий этап.

Вследствие сильной зависимости между различными группами разработчиков возникает проблема информационной перенасыщенности. Данная проблема заключается в том, что при внесении изменений в одну из частей проекта необходимо оповещать всех разработчиков, которые использовали или могли бы использовать эту часть в своей работе. Когда система состоит из большого количества взаимосвязанных подсистем, то синхронизация внутренней документации становится важной самостоятельной задачей. Разработчикам необходимо ознакомиться с изменениями и оценить, не сказались ли эти изменения на уже полученных результатах. Все это может потребовать проведения повторного тестирования и даже внесения изменений в уже готовые части проекта. Причем эти изменения, в свою очередь, должны быть отражены во внутренней документации и переданы всем группам разработчиков. Как следствие, объем документации



по мере разработки проекта растет очень быстро, так что требуется все больше времени для составления документации и ознакомления с ней.

Следует также отметить, что помимо изучения нового материала не отпадает необходимость в анализе старой информации. Это связано с тем, что вполне вероятна ситуация, когда в процессе разработки изменяется состав группы разработчиков (этот процесс носит название ротации кадров). Новым разработчикам необходимы сведения о том, что было сделано до них. Причем чем сложнее проект, тем больше времени требуется, чтобы ввести нового разработчика в курс дела.

Сложность управления проектом при использовании каскадной схемы в основном обусловлена строгой последовательностью стадий разработки и наличием сложных взаимосвязей между различными частями проекта. Последовательность разработки проекта приводит к тому, что одни группы разработчиков должны ожидать результатов работы других команд. Поэтому для согласования сроков работы и состава передаваемой документации требуется административное вмешательство. При обнаружении ошибок в выполненной работе необходимо вернуться к предыдущим этапам выполнения проекта. Это приводит к дополнительным сложностям в управлении проектом. Разработчики, допустившие просчет или ошибку, вынуждены прервать текущую работу (над новым проектом) и заняться исправлением ошибок. Следствием этого обычно является срыв сроков выполнения как исправляемого, так и нового проектов. Требовать же от команды разработчиков ожидания окончания следующей стадии разработки нерационально, так как приводит к существенным потерям рабочего времени.

Упростить взаимодействие между группами разработчиков и уменьшить информационную перенасыщенность документации можно, сокращая количество связей между отдельными частями проекта.

Очевидно, что при использовании каскадного подхода повышается уровень риска проекта. Чем сложнее проект, тем больше продолжительность каждого из этапов разработки и тем сложнее взаимосвязи между отдельными частями проекта, количество которых также увеличивается. Причем результаты разработки можно реально увидеть и оценить лишь на этапе тестирования, т.е. после завершения анализа, проектирования и разработки — этапов, выполнение которых требует значительного времени и средств. Как уже было отмечено, запоздалая оценка порождает серьезные проблемы при выявлении ошибок анализа и проектирования — требуется возврат проекта на предыдущие стадии и повторение процесса разработки.

Возврат на предыдущие стадии может быть связан не только с ошибками, но и с изменениями, произошедшими в предметной области или в требованиях заказчика за время разработки. Причем возврат проекта на доработку вследствие этих причин не гарантирует, что предметная область снова не изменится к тому моменту, когда будет готова следующая версия проекта. Фактически это означает, что существует вероятность того, что процесс разработки «заикнется», и система никогда не дойдет до сдачи в эксплуатацию. Расходы на проект будут постоянно расти, а сроки сдачи готового продукта постоянно откладываться.

Поэтому можно утверждать, что сложные проекты, разрабатываемые по каскадной схеме, имеют повышенный уровень риска.

Помимо рассмотренных существует еще один серьезный недостаток, присущий каскадной модели разработки, на который также следует обратить внимание. Этот недостаток связан с конфликтом (не всегда явным) между разработчиками, участвующими в выполнении проекта. Конфликт обусловлен тем, что возврат части проекта на предыдущую стадию обычно сопровождается поиском причин и виновных. А так как однозначно персонифицировать ответственного за ошибки можно далеко не всегда, попытки поиска виноватых могут сильно усложнить отношения в коллективе. Как следствие, в рабочей группе часто ценится не тот руководитель, который имеет высокую квалификацию и большой опыт, а тот, кто умеет «отстоять» своих подчиненных, обеспечить им более удобные условия работы и т. п. В результате появляется опасность снижения и квалификации, и творческого потенциала всей команды. Соответственно, техническое руководство проектом начинает в большей степени подменяться организационным руководством, все более детальной проработкой должностных инструкций и все более формальным их исполнением.

Каскадный подход хорошо зарекомендовал себя при построении относительно простых информационных систем, когда в самом начале разработки можно достаточно точно и полно сформулировать все требования к системе.

### **3.3.2. Спиральная модель жизненного цикла**

Спиральная модель ЖЦ была предложена для преодоления перечисленных проблем каскадной модели. На этапах анализа и проектирования реализуемость технических решений и степень удовлетворения потребностей заказчика проверяется путем создания прототипов. Каждый виток спирали соответствует созданию работоспособного фрагмента или версии системы. Это позволяет уточнить требования, цели и характеристики проекта, определить

качество разработки, спланировать работы следующего витка спирали. Таким образом, углубляются и последовательно конкретизируются детали проекта, и в результате выбирается обоснованный вариант, который удовлетворяет действительным требованиям заказчика и доводится до реализации.

Спиральная модель (рис. 3.3) в отличие от каскадной предполагает итерационный процесс разработки информационной системы. При этом возрастает значение начальных этапов жизненного цикла, таких как анализ и проектирование. На этих этапах проверяется и обосновывается реализуемость технических решений путем создания прототипов. Каждая итерация представляет собой законченный цикл разработки, приводящий к выпуску внутренней или внешней версии изделия (или подмножества конечного продукта), которое совершенствуется от итерации к итерации, чтобы стать законченной системой. На каждом витке спирали уточняются цели и характеристики проекта, определяется его качество, планируются работы на следующем витке. На каждой итерации углубляются и последовательно конкретизируются детали проекта, в результате чего выбирается обоснованный вариант, который доводится до окончательной реализации.

Использование спиральной модели позволяет перейти на следующий этап выполнения проекта, не дожидаясь полного завершения текущего — недоделанную работу можно будет выполнить на следующей итерации. Главная задача каждой итерации — как можно быстрее создать работоспособный продукт, который можно показать пользователям системы. Таким образом, существенно упрощается процесс внесения уточнений и дополнений в проект.

**Достоинства спиральной модели.** Спиральный подход к разработке программного обеспечения позволяет преодолеть боль-

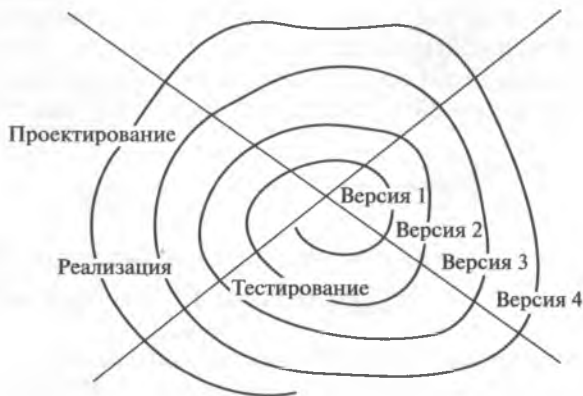


Рис. 3.3. Спиральная модель жизненного цикла ИС

шинство недостатков каскадной модели и, кроме того, обеспечивает ряд дополнительных возможностей, делая процесс разработки более гибким. Рассмотрим преимущества итерационного подхода более подробно.

1. Итерационная разработка существенно упрощает внесение изменений в проект при изменении требований заказчика.

2. При использовании спиральной модели отдельные элементы информационной системы интегрируются в единое целое постепенно. При итерационном подходе интеграция производится фактически непрерывно. Поскольку интеграция начинается с меньшего количества элементов, то возникает гораздо меньше проблем при ее проведении (по некоторым оценкам, при использовании каскадной модели разработки интеграция занимает до 40 % всех затрат в конце проекта).

3. Уменьшение уровня рисков. Данное преимущество является следствием предыдущего, так как риски обнаруживаются именно во время интеграции. Поэтому уровень рисков максимален в начале разработки проекта. По мере продвижения разработки ожидаемый уровень рисков снижается. Данное утверждение справедливо при любой модели разработки, однако при использовании спиральной модели снижение уровня рисков происходит с наибольшей скоростью. Это связано с тем, что при итерационном подходе интеграция выполняется уже на первой итерации, и на начальных итерациях выявляются многие аспекты проекта, такие как пригодность используемых инструментальных средств и программного обеспечения, квалификация разработчиков и т. п. На рис. 3.4 приведены в сравнении графики зависимости уровня рисков от времени разработки для каскадного и итерационного подходов.

4. Итерационная разработка обеспечивает большую гибкость в управлении проектом, давая возможность внесения тактических изменений в разрабатываемое изделие. Например, можно сократить сроки разработки за счет снижения функциональности системы или использовать в качестве составных частей системы



Рис. 3.4. Зависимость рисков от времени разработки

продукцию сторонних фирм вместо собственных разработок. Это может быть актуальным в условиях конкурентной борьбы, когда необходимо противостоять продвижению изделия, предлагаемого конкурентами.

5. Итерационный подход упрощает повторное использование компонентов (реализует компонентный подход к программированию). Это обусловлено тем, что гораздо проще выявить (идентифицировать) общие части проекта, когда они уже частично разработаны, чем пытаться выделить их в самом начале проекта. Анализ проекта после проведения нескольких начальных итераций позволяет выявить общие многократно используемые компоненты, которые на последующих итерациях будут совершенствоваться.

6. Спиральная модель позволяет получить более надежную и устойчивую систему. Это связано с тем, что по мере развития системы ошибки и слабые места обнаруживаются и исправляются на каждой итерации. Одновременно могут корректироваться критические параметры эффективности, что в случае каскадной модели доступно только перед внедрением системы.

7. Итерационный подход дает возможность совершенствовать процесс разработки — анализ, проводимый в конце каждой итерации, позволяет проводить оценку того, что должно быть изменено в организации разработки, и улучшить ее на следующей итерации.

**Недостатки спиральной модели.** Основная проблема спирального цикла — определение момента перехода на следующий этап. Для ее решения необходимо ввести временные ограничения на каждый из этапов жизненного цикла. Иначе процесс разработки может превратиться в бесконечное совершенствование уже сделанного. При итерационном подходе полезно следовать принципу «лучшее — враг хорошего». Поэтому завершение итерации необходимо проводить строго в соответствии с планом, даже если не вся запланированная работа закончена. Планирование работ обычно проводится на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков.

### **3.3. Процессы жизненного цикла информационной системы**

Международный стандарт ISO/IES 12207 определяет действия, которые могут быть выполнены на протяжении жизненного цикла программного обеспечения. Каждый процесс разделен на набор действий, каждое действие — на набор задач.

Согласно стандарту структура жизненного цикла основывается на трех группах процессов:

1) основные процессы жизненного цикла (приобретение, поставка, разработка, эксплуатация, сопровождение);

2) вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, разрешение проблем);

3) организационные процессы (управление проектами, создание инфраструктуры проекта, усовершенствование, обучение).

Рассмотрим каждую из указанных групп более подробно.

### **3.3.1. Основные процессы жизненного цикла**

Выделяют пять основных процессов жизненного цикла программного обеспечения. Каждый процесс характеризуется определенными задачами и методами их решения, исходными данными и результатами. Под основным участником процесса понимается сторона, которая инициирует или выполняет разработку, эксплуатацию или сопровождение программного изделия. Это покупатель, поставщик, разработчик, персонал эксплуатации и персонал сопровождения программных изделий.

*Процесс приобретения* определяет действия предприятия-покупателя, которое приобретает информационную систему.

*Процесс поставки* определяет действия предприятия-поставщика, снабжающего покупателя информационной системой.

*Процесс разработки* определяет действия предприятия-разработчика, включает в себя стратегическое планирование, анализ, проектирование и реализацию (программирование), т.е. все работы по созданию информационного программного обеспечения и его компонентов в соответствии с заданными требованиями. Включает также оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для тестирования разработанных программных продуктов и для обучения персонала. Разработка является одним из важнейших процессов жизненного цикла информационной системы.

*Процесс эксплуатации* определяет действия персонала, обеспечивающие обслуживание информационной системы в процессе ее функционирования в интересах пользователей. Основные эксплуатационные работы включают непосредственно эксплуатацию, локализацию проблем и устранение причин их возникновения, модификацию программного обеспечения, подготовку предложений по совершенствованию системы, развитие и модернизация.

цию системы. В процесс эксплуатации входят также конфигурирование базы данных и рабочих мест пользователей, обеспечение пользователей эксплуатационной документацией, обучение персонала.

*Процесс сопровождения* определяет действия персонала, обеспечивающие сопровождение программного продукта, что представляет собой управление модификациями программного продукта, поддержку его текущего состояния и функциональной пригодности и включает в себя инсталляцию и удаление программного обеспечения системы. Сопровождение — процесс выпуска и внедрения новых версий программного обеспечения информационной системы. Причинами выпуска новых версий могут быть:

- необходимость исправления ошибок, выявленных в процессе эксплуатации предыдущих версий;
- необходимость совершенствования предыдущих версий, например, улучшения интерфейса или расширения состава выполняемых функций;
- изменение среды функционирования, например, появление новых технических средств и/или программных продуктов.

В процессе сопровождения в программное обеспечение вносятся необходимые изменения, которые могут потребовать пересмотра проектных решений, принятых на любом этапе жизненного цикла. В спиральной модели жизненного цикла информационной системы роль этапа сопровождения существенно возросла, так как продукты теперь создаются итерационно: сначала выпускается сравнительно простая версия, затем следующая с большими возможностями, затем следующая и т. д. Именно это и послужило причиной выделения этапа сопровождения в отдельный процесс жизненного цикла в соответствии со стандартом ISO/IEC 12207.

В жизни любой корпоративной информационной системы весьма заметную роль играют службы технической поддержки. Наличие квалифицированного технического обслуживания на этапе эксплуатации информационной системы является необходимым условием решения поставленных перед ней задач, причем ошибки обслуживающего персонала могут приводить к явным или скрытым финансовым потерям, сопоставимым со стоимостью самой информационной системы. Основными предварительными действиями при подготовке к организации технического обслуживания информационной системы являются:

- выделение наиболее ответственных узлов системы и определение для них критичности простоя (это позволит выделить наиболее критичные составляющие информационной системы и оптимизировать распределение ресурсов для технического обслуживания);

– определение задач технического обслуживания и их разделение на внутренние задачи, решаемые силами обслуживающего подразделения, и внешние задачи, решаемые специализированными сервисными организациями (таким образом производится четкое определение круга исполняемых функций и разделение ответственности);

– проведение анализа имеющихся внутренних и внешних ресурсов, необходимых для организации технического обслуживания в рамках описанных задач и разделения компетенции (основные критерии для анализа: наличие гарантии на оборудование, состояние ремонтного фонда, квалификация персонала);

– подготовка плана организации технического обслуживания, в котором необходимо определить этапы исполняемых действий, сроки их исполнения, затраты на этапах, ответственность исполнителей.

Обеспечение качественного технического обслуживания информационной системы требует привлечения специалистов высокой квалификации, которые в состоянии решать не только каждодневные задачи администрирования, но и быстро восстанавливать работоспособность системы при сбоях.

### **3.3.2. Вспомогательные процессы жизненного цикла**

Вспомогательные процессы поддерживают реализацию основных процессов, будучи неотъемлемой частью всего жизненного цикла информационной системы, с определенной целью и обеспечивают должное качество информационной системы. Вспомогательные процессы используются и выполняются по мере необходимости и иницируются другими процессами.

*Процесс документирования* определяет действия для записи информации, являющейся результатом выполнения какого-либо процесса жизненного цикла информационной системы.

*Процесс управления конфигурацией* определяет действия по управлению конфигурацией. Это тот вспомогательный процесс, который поддерживает основные процессы жизненного цикла информационной системы, прежде всего процессы разработки и сопровождения. При разработке проектов сложных информационных систем, состоящих из многих компонентов, каждый из которых может разрабатываться независимо и, следовательно, иметь несколько вариантов реализации и/или несколько версий одной реализации, возникает проблема учета их связей и функций, создания единой структуры и обеспечения развития всей системы. Управление конфигурацией позволяет организовывать, система-



тически учитывать и контролировать внесение изменений в различные компоненты информационной системы на всех стадиях ее жизненного цикла.

*Процесс обеспечения качества* определяет действия для объективной гарантии, что информационная система и процессы соответствуют определенным требованиям к ним и придерживаются установленным замыслам. Совместная оценка, верификация, проверки, аттестации могут быть использованы как способы гарантии качества.

*Процесс верификации* определяет действия (для покупателя, поставщика или независимой стороны) для верификации программного обеспечения информационной системы с различной глубиной зависимости от проекта.

*Процесс аттестации* определяет действия (покупателя, поставщика, независимой стороны) для аттестации программного обеспечения информационной системы.

*Процесс совместной оценки* определяет действия для оценки состояния и результатов какого-либо действия. Этот процесс может быть использован любыми двумя сторонами, где одна сторона (проверяющая, рецензирующая) проверяет (рецензирует) другую сторону (проверяемую) на совместном форуме.

*Процесс проверки* определяет деятельность для определения соответствия с требованиями, замыслами и контрактом. Этот процесс может быть использован любыми двумя сторонами, где одна сторона (проверяющая) проверяет программное обеспечение информационной системы или деятельность другой стороны (проверяемой).

*Процесс решения проблем* определяет процесс анализа и устранения проблем (включая несоответствия), какова бы ни была их природа или источник во время разработки, эксплуатации, сопровождения или других процессов.

### **3.3.3. Организационные процессы**

Организационные процессы жизненного цикла выполняются какой-либо организацией с целью:

- создания и обеспечения деятельности нижестоящей структуры, включающей в себя связанные процессы жизненного цикла и персонал;

- совершенствования структуры и процессов.

Они, как правило, инвариантны относительно конкретных проектов и контрактов, однако уроки, извлеченные из таких проектов и контрактов, способствуют совершенствованию организации.

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков и контроля за сроками и качеством выполняемых работ. Техническое и организационное обеспечение проекта включает в себя:

- выбор методов и инструментальных средств для реализации проекта;
- определение методов описания промежуточных состояний разработки;
- разработку методов и средств испытаний созданного программного обеспечения;
- обучение персонала.

Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования компонентов информационной системы. Верификация — это процесс определения соответствия текущего состояния разработки, достигнутого на данном этапе, требованиям этого этапа. Проверка — это процесс определения соответствия параметров разработки исходным требованиям. Проверка отчасти совпадает с тестированием, которое проводится для определения различий между действительными и ожидавшимися результатами и оценки соответствия характеристик информационной системы исходным требованиям.

Процесс создания инфраструктуры управления охватывает выбор и поддержку (сопровождение) технологии, стандартов и инструментальных средств, выбор и установку аппаратных и программных средств, используемых для разработки, эксплуатации или сопровождения программного обеспечения ИС. Инфраструктура, в свою очередь, является одним из объектов управления конфигурацией.

Процесс усовершенствования предусматривает оценку, измерение, контроль и модернизацию процессов жизненного цикла. Последнее направлено на повышение производительности труда всех участвующих в них специалистов за счет совершенствования используемой технологии, методов управления, выбора инструментальных средств и обучения персонала. Усовершенствование основано на анализе достоинств и недостатков каждого процесса. Процесс обучения охватывает первоначальное обучение и последующее постоянное повышение квалификации персонала.

### **Контрольные вопросы**

1. Что такое жизненный цикл, из каких стадий он состоит?
2. Чем регламентируется жизненный цикл информационных систем?
3. Какие группы процессов входят в состав жизненного цикла, какие процессы входят в состав каждой группы?

4. Какие из процессов наиболее часто используются в реальных проектах, какие в меньшей степени и почему?
5. Что понимают под стадией жизненного цикла?
6. Какие этапы входят в состав жизненного цикла информационных систем?
7. Каковы принципиальные особенности каскадной модели?
8. В чем состоят преимущества и недостатки каскадной модели?
9. Каковы принципиальные особенности спиральной модели?
10. В чем состоят преимущества и недостатки спиральной модели?

## МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

### 4.1. Понятие модели предметной области

Под моделью предметной области понимается некоторая система, имитирующая структуру или функционирование исследуемой предметной области и отвечающая основному требованию — быть адекватной этой области. Модель должна отражать все аспекты функционирования предприятия и быть необходимой на всех этапах жизненного цикла информационной системы. Но особую роль модели предметной области играют на стадии формирования требований к будущей информационной системе при ее создании.

Предварительное моделирование предметной области позволяет сократить время и сроки проведения проекторочных работ и получить более эффективный и качественный проект. Без проведения моделирования предметной области велика вероятность допущения большого количества ошибок в решении стратегических вопросов, приводящих к экономическим потерям и высоким затратам на последующее перепроектирование системы.

К модели предметной области предъявляются следующие требования:

- формализация, обеспечивающая однозначное описание структуры предметной области;
- понятность для заказчиков и разработчиков на основе применения графических средств отображения модели;
- реалистичность, подразумевающая наличие средств физической реализации модели предметной области в ИС;
- обеспечение возможности оценки эффективности реализации предметной области на основе определенных методов и вычисляемых показателей.

Для реализации перечисленных требований необходимо построение системы моделей:

— объектной модели, отражающей состав взаимодействующих в процессах материальных и информационных объектов предметной области;

— функциональной модели, отражающей взаимосвязь функций (действий) по преобразованию объектов в процессах;

— модели управления, отражающей события и бизнес-правила, воздействующие на выполнение процессов;

— организационной модели (структуры), отражающей взаимодействие организационных единиц предприятия и персонала в процессах;

— технической модели, описывающей топологию расположения и способы коммуникации комплекса технических средств.

Главный критерий адекватности модели предметной области заключается в функциональной полноте разрабатываемой ИС.

С моделированием непосредственно связана проблема выбора языка представления проектных решений. Язык моделирования и я — это графическая нотация, которая используется для описания проектов. Нотация представляет собой совокупность графических объектов, используемых в модели, и является синтаксисом языка моделирования. Язык моделирования, с одной стороны, должен делать решения проектировщиков понятными пользователю, с другой стороны, предоставлять проектировщикам средства достаточно формализованного и однозначного определения проектных решений, подлежащих реализации в виде программных комплексов, образующих целостную систему.

Обычно модели строятся на трех уровнях — внешнем (определение требований), концептуальном (спецификация требований), внутреннем (реализация требований).

Так, на внешнем уровне модель отвечает на вопрос «Что должна делать система?», т. е. определяется состав основных компонентов системы: объектов, функций, событий, организационных единиц, технических средств. На концептуальном уровне модель отвечает на вопрос «Как должна функционировать система?». Иначе говоря, определяется характер взаимодействия компонентов системы. На внутреннем уровне модель отвечает на вопрос: «С помощью каких программно-технических средств реализуются требования к системе?». С позиции жизненного цикла ИС описанные уровни моделей соответственно строятся на этапах анализа требований, технического и рабочего проектирования.

При работе с большими и сложными информационными системами проблема сложности является главной проблемой. Ни один разработчик не в состоянии понять всю систему в целом, поскольку это выше человеческих возможностей. Единственный эффективный подход к решению этой проблемы заключается в построении сложной системы из небольшого количества крупных

частей, каждая из которых, в свою очередь, строится из частей меньшего размера. Процесс продолжается до тех пор, пока самые небольшие части станут доступны для восприятия и понимания. Это означает, что сложную программную систему нужно разделять (декомпозировать) на небольшие подсистемы и рассматривать их независимо друг от друга.

## **4.2. Структурный подход в моделировании предметной области**

### **4.2.1. Сущность структурного подхода**

Сущность структурного подхода к разработке ИС заключается в ее декомпозиции (разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи, и т. д. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимосвязаны. При разработке системы «снизу-вверх» от отдельных задач ко всей системе целостность теряется, возникают проблемы при информационной стыковке отдельных компонентов.

Структура проектируемой системы должна быть таковой, чтобы все взаимодействие между ее подсистемами укладывалось в ограниченные стандартные рамки:

- каждая подсистема должна инкапсулировать свое содержимое (скрывать его от других подсистем);
- каждая подсистема должна иметь четкий интерфейс с другими подсистемами.

Инкапсуляция дает возможность рассматривать структуру каждой подсистемы независимо от других подсистем. Интерфейсы позволяют строить систему более высокого уровня, рассматривая каждую подсистему как единое целое.

Все наиболее распространенные методологии структурного подхода базируются на следующих принципах:

- принцип «разделяй и властвуй» — решение сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения;
- принцип иерархического упорядочивания — организация составных частей проблемы в иерархические древовидные структуры с добавлением новых деталей на каждом уровне.

В структурном анализе используют группы средств, иллюстрирующих функции, выполняемые системой, и отношения между

данными. Каждой группе средств соответствуют определенные виды моделей (диаграмм), наиболее распространенными среди которых являются следующие:

- SADT (Structured Analysis and Design Technique — метод структурного анализа и проектирования) — модели и соответствующие диаграммы;
- DFD (Data Flow Diagrams) — диаграммы потоков данных;
- ERD (Entity-Relationship Diagrams) — диаграммы «сущность-связь» (модель данных).

Перечисленные модели в совокупности дают полное описание ИС независимо от того, является ли она существующей или вновь разрабатываемой. Состав диаграмм в каждом конкретном случае зависит от необходимой полноты описания системы.

#### **4.2.2. Методология функционального моделирования SADT**

Методология SADT является методологией функционального моделирования бизнес-процессов. Метод SADT поддерживается Министерством обороны США, которое было инициатором разработки стандарта IDEF0<sup>1</sup>. Метод SADT представляет собой совокупность правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области. Функциональная модель отображает функциональную структуру объекта, т.е. производимые им действия и связи между этими действиями. Результатом применения методологии SADT является модель, которая состоит из диаграмм, фрагментов текстов и глоссария, имеющих ссылки друг на друга.

Методология SADT может использоваться:

- для моделирования широкого круга систем и определения требований и функций;
- для разработки системы, которая удовлетворяет этим требованиям и реализует эти функции.

SADT может быть использована в существующих системах:

- для анализа функций, выполняемых системой;
- для указания механизмов, посредством которых они осуществляются.

Диаграммы — главные компоненты модели. Функции системы и интерфейсы представлены на диаграммах как блоки и дуги. Место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, в то время как

---

<sup>1</sup> IDEF0 — методология функционального моделирования (представление бизнес-системы в виде взаимосвязанных функций). Фактически это целое семейство методологий от IDEF0 до IDEF14.

информация, которая подвергается обработке, показана с левой стороны блока, а результаты выхода показаны с правой стороны. Механизм (человек или автоматизированная система), который осуществляет операцию, представлен дугой, входящей в блок снизу (рис. 4.1).

Одна из наиболее важных особенностей методологии SADT — постепенное введение все больших уровней детализации по мере создания диаграмм, отображающих модель.

Построение SADT-модели начинается с изображения всей системы в виде простейшей компоненты — одного блока и дуг, обозначающих интерфейсы с функциями вне системы. Поскольку единственный блок характеризует всю систему как единое целое, имя, указанное в блоке, является общим. Это верно и для интерфейсных дуг — они также представляют полный набор внешних интерфейсов системы в целом.

Затем блок, который характеризует систему как единый модуль, детализируется на другой диаграмме с помощью нескольких блоков, соединенных интерфейсными дугами. Эти блоки отражают основные подфункции исходной функции. Данная декомпозиция выявляет полный набор подфункций, каждая из которых дана как блок, границы которого определены интерфейсными дугами. Каждая из этих подфункций может быть декомпозирована подобным образом для более детального изображения. Во всех случаях каждая подфункция может содержать только те элементы, которые входят в исходную функцию. Кроме того, модель не может опустить какие-либо элементы, т. е. как уже отмечалось, родительский блок и его интерфейсы обеспечивают контекст. К нему нельзя ничего добавить, и из него не может быть ничего удалено.

Модель SADT представляет собой серию диаграмм с сопроводительной документацией, разбивающих сложный объект на составные части в виде блоков. Детали каждого из основных блоков показаны в виде блоков на других диаграммах. Каждая детальная диаграмма является декомпозицией блока из более общей диа-

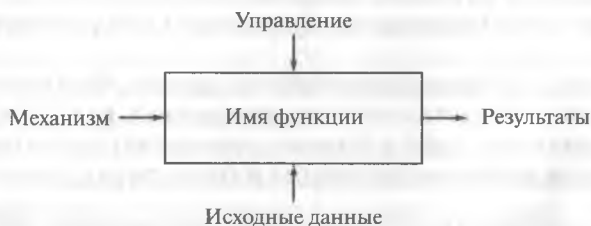


Рис. 4.1. Функциональный блок и интерфейсные дуги





Рис. 4.2. Функциональная диаграмма начального уровня

граммы. На каждом шаге декомпозиции более общая диаграмма называется родительской для более детальной диаграммы.

Дуги, входящие в блок и выходящие из него на диаграмме верхнего уровня, являются теми же самыми, что и дуги, входящие в диаграмму нижнего уровня и выходящие из нее, потому что блок и диаграмма представляют одну и ту же часть системы.

Стрелки, приходящие с родительской диаграммы или уходящие на нее, нумеруют, используя символы и числа. Символ обозначает тип связи: I — входные, С — управляющие, М — механизмы, R — результаты. Число — номер связи по соответствующей стороне родительского блока, считая сверху вниз и слева направо. Все диаграммы связывают друг с другом иерархической нумерацией блоков: первый уровень — A0, следующий — A1, A2, ..., A11, A12, A13 и т. д., где «A1» — номер родительского блока, а «1» — номер конкретного субблока родительского блока. Детализацию завершают при получении функций, назначение которых хорошо понятно как заказчику, так и разработчику. Эти функции описывают, используя естественный язык или псевдокоды. В процессе построения иерархии диаграмм фиксируют всю уточняющую информацию и строят словарь данных, в котором определяют структуры и элементы данных, показанных на диаграммах. Таким образом, в результате получают спецификацию, которая состоит из иерархии функциональных диаграмм, описаний функций нижнего уровня и словаря, имеющих ссылки друг на друга.

На SADT-диаграммах не указаны явно ни последовательность, ни время. Обратные связи, итерации, продолжающиеся процессы и перекрывающиеся (по времени) функции могут быть изображены также с помощью дуг. Обратные связи могут выступать в виде комментариев, замечаний, исправлений и т. д.

Приведем пример построения функциональной диаграммы для информационной системы приема и зачисления студентов. Диаграмма, показанная на рис. 4.2, является диаграммой верхнего

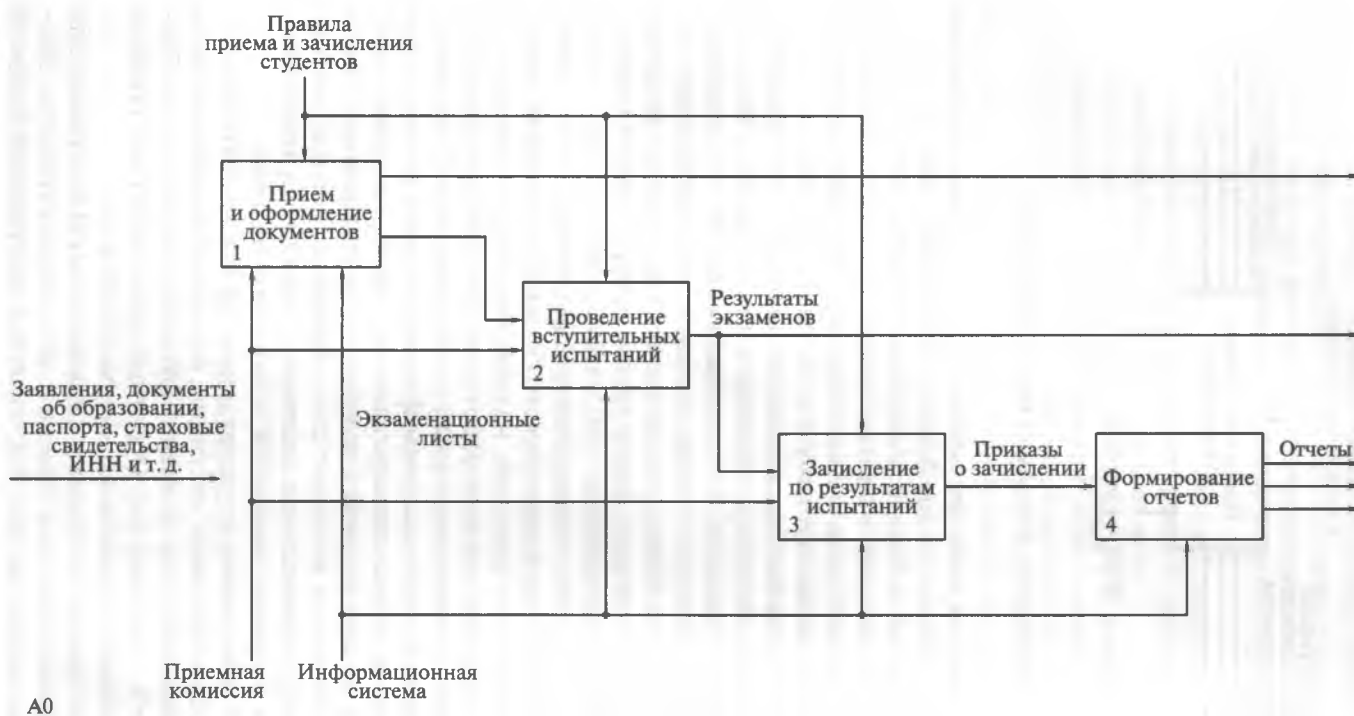


Рис. 4.3. Функциональная диаграмма нулевого уровня (более подробный вариант)

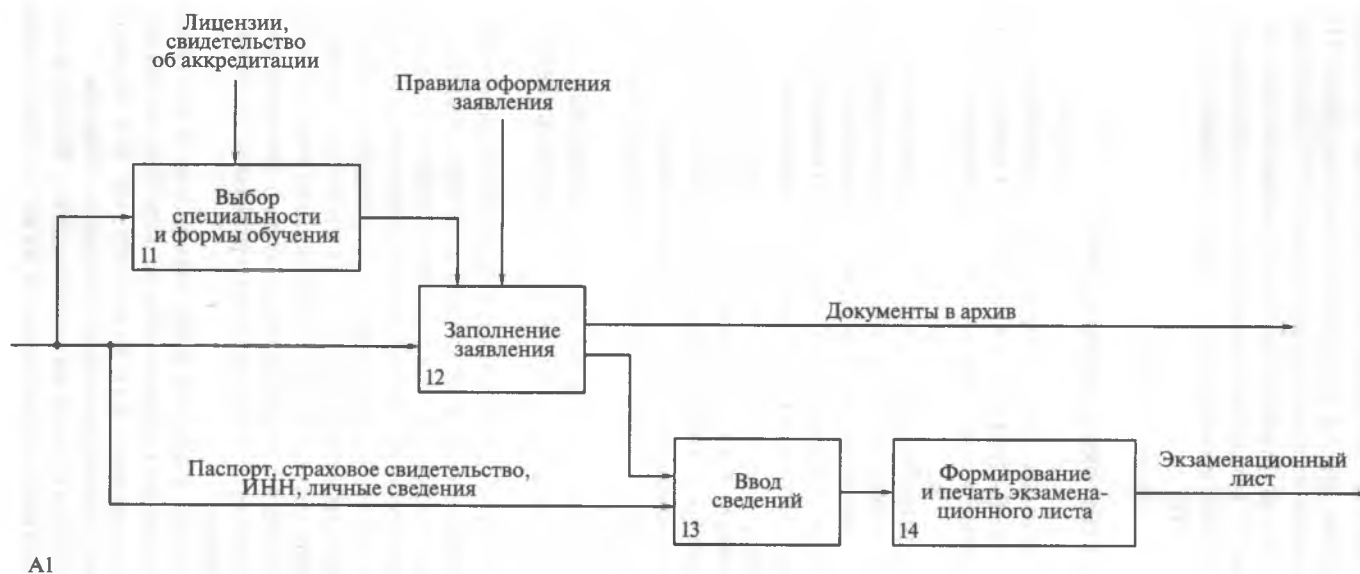


Рис. 4.4. Функциональная диаграмма первого уровня — «Прием и оформление документов»

уровня. На ней хорошо видно, что является исходными данными для системы, и получения каких результатов мы ожидаем. На функциональной диаграмме нулевого уровня (рис. 4.3) родительский блок A0 разбивается на функциональные блоки A1, A2, A3, A4. В свою очередь блок A1 может быть представлен на функциональной диаграмме первого уровня как четыре дочерних блока A11, A12, A13, A14 (рис. 4.4).

#### 4.2.3. Диаграммы потоков данных DFD

Диаграмма потоков данных (DFD) состоит из узлов обработки данных, средств хранения данных и внешних по отношению к используемой диаграмме источников или потребителей данных.

Диаграмма потоков данных является основным средством моделирования функциональных требований к системе — проектируемой или реально существующей. В основе модели лежат понятия внешней сущности, процесса, хранилища (накопителя данных) потока данных. Источники информации (внешние сущности) порождают информационные потоки (потоки данных), переносящие информацию к подсистемам или процессам. Те, в свою очередь, преобразуют информацию и порождают новые потоки, которые переносят информацию к другим процессам или подсистемам, накопителям данных или внешним сущностям — потребителям информации.

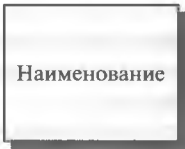
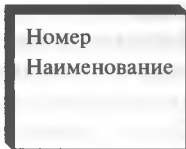






Для изображения диаграмм потоков данных традиционно используют два вида нотаций: нотацию Йордана и нотацию Гейна-Сарсона (табл. 4.1).

*Внешняя сущность* — это материальный предмет или физическое лицо, представляющее собой источник или приемник информации, — заказчики, персонал, поставщики, клиенты, склад. Определение некоторого объекта или системы в качестве внешней сущности указывает на то, что она находится за пределами границ анализируемой информационной системы. В процессе анализа некоторые внешние сущности могут быть перенесены внутрь, если это необходимо, или, наоборот, часть процессов информационной системы может быть вынесена за пределы диаграммы и представлена как внешняя сущность.

При построении модели сложной информационной системы она может быть представлена в самом общем виде на так называемой контекстной диаграмме как одна система, т.е. единое целое, либо может быть декомпозирована на ряд подсистем.

Номер подсистемы служит для ее идентификации. В поле имени вводится наименование подсистемы в виде предложения

Таблица 4.1

Понятие	Нотация Йордана	Нотация Гейна-Сарсона
Внешняя сущность		
Система, подсистема или процесс		
Накопитель данных		
Поток		

с подлежащим и соответствующими определениями и дополнениями.

*Процесс* представляет собой преобразование входных потоков данных в выходные в соответствии с определенным алгоритмом. Физически процесс может быть реализован различными способами: это может быть подразделение организации (отдел), выполняющее обработку входных документов и формирование отчетов, программа, аппаратно реализованное логическое устройство и т. д. Номер процесса служит для его идентификации. В поле имени вводится наименование процесса, при этом использование таких глаголов, как «обработать», «модернизировать» или «отредактировать», означает, как правило, недостаточно глубокое понимание данного процесса и требует дальнейшего анализа. Информация в поле физической реализации показывает, какое подразделение организации, программа или аппаратное устройство выполняет данный процесс.

*Накопитель данных* представляет собой абстрактное устройство для хранения информации, которую можно в любой момент поместить в накопитель и через некоторое время извлечь, причем

способы помещения и извлечения могут быть любыми. Накопитель данных может быть реализован физически в виде ящика в картотеке, таблицы в оперативной памяти, файла на магнитном носителе и т. д.

Накопитель данных в общем случае является прообразом базы данных, и описание хранящихся в нем данных должно быть связано с информационной моделью.

*Поток данных* определяет информацию, передаваемую через некоторое соединение от источника к приемнику. Реальный поток данных может быть информацией, передаваемой по кабелю между двумя устройствами, пересылаемыми по почте письмами, магнитными лентами или дискетами, переносимыми с одного компьютера на другой и т. д. Каждый поток данных имеет имя, отражающее его содержание.

Построение иерархии диаграмм потоков данных начинают с контекстной диаграммы, которая определяет общий вид системы. На такой диаграмме описывают интерфейс системы с внешним миром, т. е. показывают, как разрабатываемая система будет взаимодействовать с потребителями и источниками информации. Обычно начальная контекстная диаграмма имеет форму звезды.

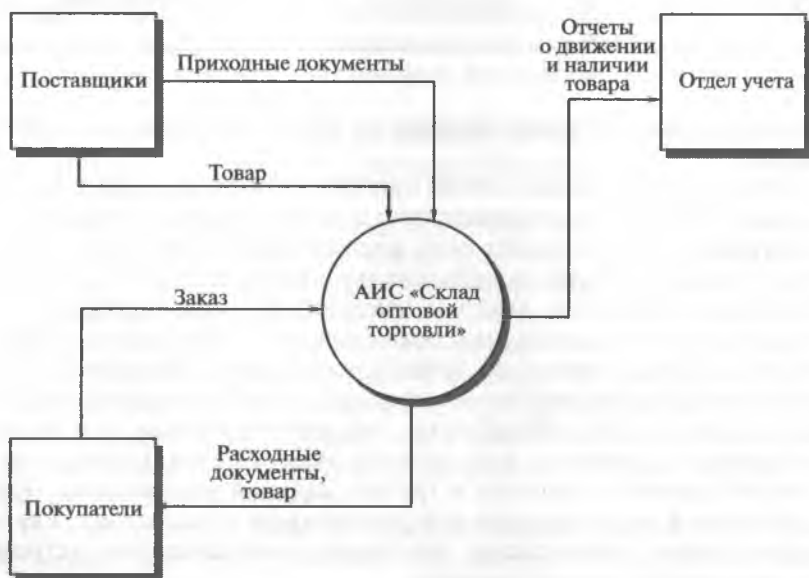


Рис. 4.5. Начальная контекстная диаграмма (диаграмма нулевого уровня) в нотации Йордана для АИС «Склад оптовой торговли»

Если система содержит большое число внешних сущностей (более 10), имеет распределенную природу или включает уже существующие подсистемы, то строят иерархии контекстных диаграмм. В процессе детализации соблюдают правило балансировки — при детализации подсистемы могут использоваться компоненты только тех подсистем, с которыми у анализируемой подсистемы существует информационная связь (т.е. с которыми она связана потоками данных). На недетализируемые процессы составляют спецификации, которые должны содержать текстовое описание функций данного процесса.

Приведем пример построения диаграммы потоков данных автоматизированной информационной системы (АИС) «Склад оптовой торговли». АИС «Склад оптовой торговли» предназначена для получения данных о движении и наличии товаров, приобретенных для оптовой продажи.

Построение иерархии диаграмм потоков данных начнем с контекстной диаграммы, т.е. определим, как разрабатываемая система будет взаимодействовать с приемниками и источниками информации (рис. 4.5).

Первичные документы по приходу товаров (приходные документы) фиксируются в журнале поступления товаров. Оформление и учет реализации товаров зависят от способа расчета за приобретаемые товары между покупателем и продавцом. Менеджер склада ведет журнал учета закупок и отпуска товаров. Данные первичных документов сохраняются в соответствующих накопителях.

Внешними сущностями для разрабатываемой системы являются поставщики, покупатели, менеджер склада, отдел учета и контроля, отдел приема и оформления заказов. Сведения о них сохраняются в соответствующих таблицах (справочниках). Поставщик передает товар на склад. Документы на поставку товара (накладные, счета-фактуры) вводят в базу данных. Покупатель подает заказ на приобретение товаров. В отделе приема и оформления заказов проверяется каждая строка поступившего заказа. При отсутствии на складе какой-либо позиции оформляется заявка поставщику, т.е. инициируется поставка необходимого товара. На основании заказа заполняются документы на реализацию товара, которые сохраняются в базе данных, распечатываются и выдаются покупателю. В конце каждого дня все приходные документы передаются бухгалтеру. На основании сведений о приходе и реализации товара менеджеры склада и работники отдела учета формируют отчеты по оборотам и остаткам товара на складе.

Диаграмма потоков данных АИС «Склад оптовой торговли» представлена на рис. 4.6.

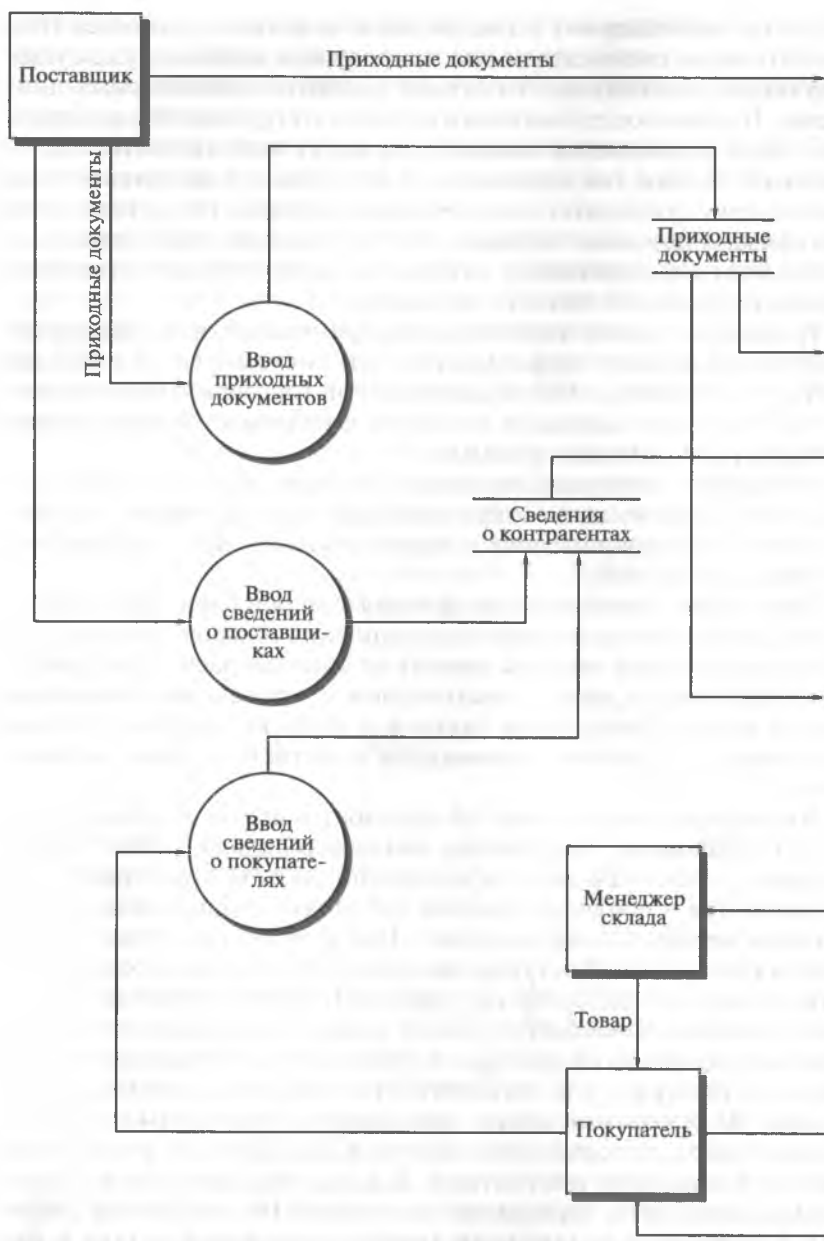
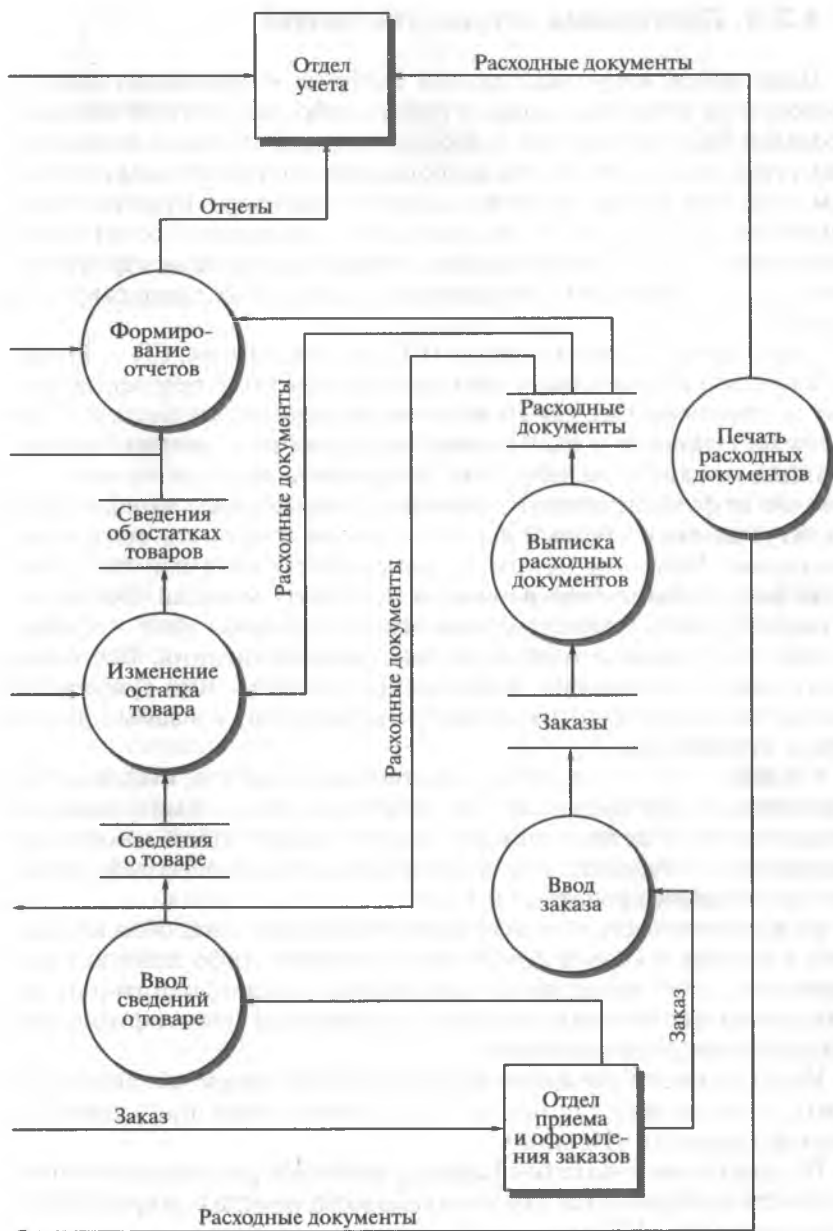


Рис. 4.6. Диаграмма потоков данных АИС «Склад оптовой торговли»





#### 4.2.4. Диаграмма «сущность-связь»

Цель любой информационной системы — обработка данных об объектах реального мира в какой-либо предметной области. Создавая базу данных для информационной системы, пользователь стремится упорядочить информацию по различным признакам, для того чтобы по необходимости извлекать нужную совокупность данных, получать выборку с желаемым сочетанием признаков. Сделать это возможно, только если данные структурированы, т.е. отвечают соглашениям о способах представления данных.

Диаграмма «сущность-связь» (ER-модель данных, ER — Entity-Relationship) обеспечивает стандартный способ определения данных и отношений между ними в информационной системе. Она включает сущности и взаимосвязи, отражающие основные бизнес-правила предметной области. Диаграммы «сущность-связь» в отличие от функциональных диаграмм определяют спецификации структур данных. Первый вариант модели «сущность-связь» был предложен Питером Ченом. В дальнейшем многими авторами были разработаны свои варианты подобных моделей. Все варианты диаграмм «сущность-связь» исходят из одной идеи — графическое изображение нагляднее текстового описания. Все такие диаграммы используют графическое изображение сущностей предметной области, их свойств (атрибутов) и взаимосвязей между сущностями.

**С у щ н о с т ь** — это класс однотипных объектов, информация о которых имеет существенное значение для рассматриваемой предметной области. Сущность представляет собой множество экземпляров реальных или абстрактных объектов (людей, событий, состояний, предметов и т.п.).

Каждая сущность должна иметь уникальное имя, обладать одним или несколькими атрибутами, которые либо принадлежат сущности, либо наследуются через связь, обладать одним или несколькими атрибутами, которые однозначно идентифицируют каждый экземпляр сущности.

Имя сущности должно отражать тип или класс объекта (студент), а экземпляр сущности — это конкретный представитель данной сущности (Иванов).

На диаграмме в нотации Баркера, наиболее распространенной, сущность изображается прямоугольником, иногда с закругленными углами (рис. 4.7, а).

**А т р и б у т** — любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности (рис. 4.7, б).



Рис. 4.7. Обозначения сущности в нотации Баркера:

*a* — без атрибутов; *б* — с указанием атрибутов; *в* — с уточнением атрибутов и их типов: # — ключевой; \* — обязательный; o — необязательный

Атрибут, таким образом, представляет собой некоторый тип характеристик или свойств, ассоциированных с множеством реальных или абстрактных объектов. Экземпляр атрибута — определенная характеристика конкретного экземпляра сущности, значение атрибута (например, «цвет» — это атрибут, а «зеленый» — экземпляр атрибута).

Атрибуты делятся на ключевые, т. е. входящие в состав уникального идентификатора, который называют первичным ключом, и описательные — прочие. Первичный ключ — это атрибут или совокупность атрибутов и/или связей, предназначенная для уникальной идентификации каждого экземпляра сущности (совокупность признаков, позволяющих идентифицировать объект). Ключевые атрибуты помещают в начало списка и помечают символом «#» (рис. 4.7, *в*). Описательные атрибуты могут быть обязательными или необязательными. Обязательные атрибуты для каждой сущности всегда имеют конкретное значение, необязательные — могут быть не определены. Обязательные и необязательные описательные атрибуты помечают символами «\*» и «o» соответственно.

Связь — это отношение одной сущности к другой или к самой себе. Если любой экземпляр одной сущности связан хотя бы с одним экземпляром другой сущности, то связь является обязательной (рис. 4.8, *а*). Необязательная связь представляет собой условное отношение между сущностями (рис. 4.8, *б*).

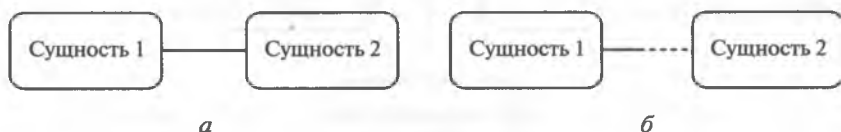


Рис. 4.8. Модальность связи:

*a* — обязательная; *б* — необязательная (пунктир до середины линии)

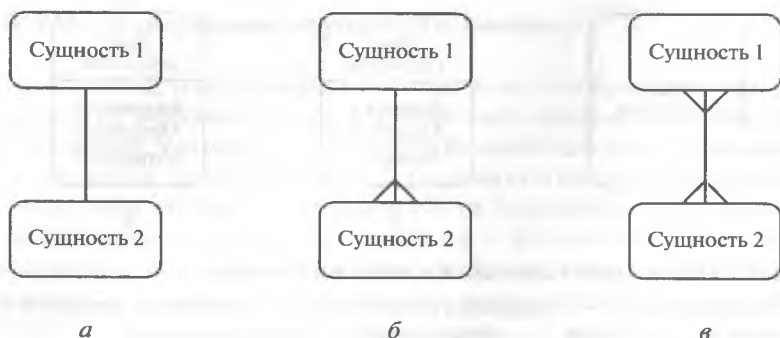


Рис. 4.9. Обозначение отношений в нотации Баркера:

*a* — «один-к-одному»; *б* — «один-ко-многим»; *в* — «многие-ко-многим»

Связь может иметь разную модальность с разных концов. Каждая связь может быть прочитана как слева направо, так и справа налево.

Каждая сущность может обладать любым количеством связей с другими сущностями модели. Различают три типа отношений:

- $1*1$  — «один-к-одному»;
- $1*n$  — «один-ко-многим»;
- $n*m$  — «многие-ко-многим» (рис. 4.9).

Связь типа «один-к-одному» означает, что один экземпляр первой сущности связан только с одним экземпляром второй сущности. Такая связь, скорее всего, свидетельствует о том, что была неверно разделена одна сущность на две (хотя иногда к такому типу связи прибегают в случае, если есть необходимость «засекретить» часть данных). Связь типа «один-ко-многим» означает, что каждый экземпляр первой сущности связан с несколькими экземплярами второй сущности. Связь типа «многие-ко-многим» означает, что каждый экземпляр первой сущности может быть связан с несколькими экземплярами второй сущности и наоборот. Такой тип связи является временным типом связи, допустимым на ранних этапах разработки модели. В дальнейшем такую связь необ-



Рис. 4.10. Обозначение ассоциированной сущности в нотации Баркера



Рис. 4.11. Обозначение идентификации посредством другой сущности в нотации Баркера

ходимо заменить двумя связями типа «один-ко-многим» путем выделения еще одной дополнительной сущности.

Независимая сущность представляет независимые данные, которые всегда присутствуют в системе. Они могут быть как связаны с другими сущностями, так и не связаны.

Зависимая сущность представляет данные, зависящие от других сущностей системы, поэтому она всегда должна быть связана с другими сущностями.

Ассоциированная сущность представляет данные, которые ассоциируются с отношениями между двумя и более сущностями. Обычно данный вид сущностей появляется в модели для разрешения отношения «многие-ко-многим» (рис. 4.10).

Если экземпляр сущности полностью идентифицируется своими ключевыми атрибутами, то говорят о полной идентификации сущности. В противном случае идентификация сущности осуществляется с использованием атрибутов связанной сущности, что указывается черточкой на линии связи (рис. 4.11).

Рассмотрим структуру базы данных для АИС «Склад оптовой торговли». Основными сущностями для решения указанной задачи являются: «поставщик», «покупатель», «товар».

Сразу возникает очевидная связь между сущностями — «покупатели могут приобретать много товаров», «товары могут приобретаться многими покупателями». Отношение между ними относится к типу «многие-ко-многим» (рис. 4.12). Для разрешения этого отношения введем две ассоциированные сущности «Приходная накладная» и «Расходная накладная», которые отражают приобретение/продажу товара покупателем/поставщиком (рис. 4.13).

Проанализируем атрибуты сущностей. Каждый поставщик и покупатель является юридическим лицом и имеет наименование, адрес, банковские реквизиты. Каждый товар имеет наименование, цену, характеризуется единицей измерения. Каждая накладная имеет уникальный номер, дату выписи, список товаров с количе-

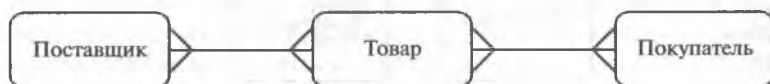


Рис. 4.12. Первый вариант ER-диаграммы



Рис. 4.13. Промежуточный вариант ER-диаграммы

ствами и ценами, а также общую сумму накладной. Покупатели покупают товары, получая при этом расходные накладные, в которые внесены данные о количестве и цене приобретенного товара. Каждый покупатель может получить несколько накладных. Каждая накладную необходимо выписывать на одного покупателя. Каждая накладная должна содержать не менее одного товара (не может быть «пустой» накладной). Каждый товар, в свою очередь, может быть продан нескольким покупателям по нескольким накладным. Ана-

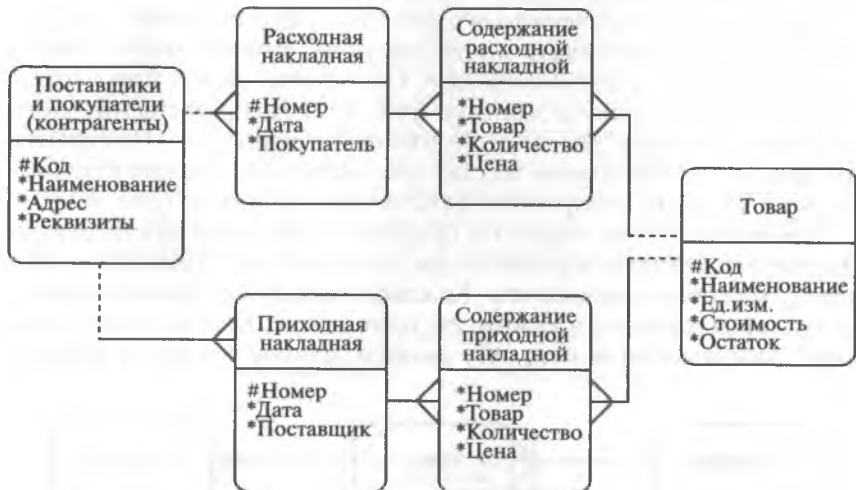


Рис. 4.14. Окончательный вариант ER-диаграммы

логичную цепь рассуждений можно выстроить для определения связей между сущностями «Товар» и «Поставщик». Покупатель может быть одновременно и поставщиком, поэтому эти две сущности объединены в одну сущность «Контрагент». Теперь можно все это внести в диаграмму. Таким образом, после уточнения диаграмма будет выглядеть следующим образом (рис. 4.14).

Данная диаграмма должна быть проверена с точки зрения возможности получения всех выходных данных, показанных на диаграмме потоков данных разрабатываемой системы.

### **4.3. Объектно-ориентированный подход в моделировании систем**

#### **4.3.1. Сущность объектно-ориентированного подхода**

Принципиальное различие между структурным и объектно-ориентированным подходами заключается в способе декомпозиции системы. Объектно-ориентированный подход использует объектную декомпозицию, при этом структура системы описывается в терминах объектов и связей между ними, а поведение системы — в терминах обмена сообщениями между объектами. Каждый объект системы обладает своим собственным поведением, моделирующим поведение объекта реального мира.

Концептуальной основой объектно-ориентированного подхода является объектная модель. Основными ее элементами являются абстрагирование, инкапсуляция, модульность, иерархия.

*Абстрагирование* (абстракция) — предписывает включать в модель только те аспекты проектируемой системы, которые имеют непосредственное отношение к выполнению системой своих функций или своего целевого назначения. При этом все второстепенные детали опускаются, чтобы чрезмерно не усложнять процесс анализа и исследования полученной модели (так же, как, например, при проектировании жилого дома не следует отвлекаться на подбор обоев для оклеивания его внутренних стен).

*Инкапсуляция* подразумевает скрытие элементов объекта, определяющих его устройство и поведение. Правило инкапсуляции — для обеспечения надежности нежелателен прямой доступ к полям объекта, чтение и обновление их содержимого должно производиться посредством вызова соответствующих методов. Абстракция и инкапсуляция — взаимодополняющие понятия: абстракция выделяет внешнее поведение объекта, а инкапсуляция содержит и

скрывает реализацию, которая обеспечивает это поведение. Инкапсуляция достигается с помощью информационной закрытости. Обычно скрываются структура объектов и реализация их методов.

*Модульность* — это свойство системы, связанное с возможностью ее декомпозиции на ряд модулей. Модули служат физическими контейнерами, в которых объявляются классы и объекты логической обработки. Общая цель декомпозиции на модули — уменьшение сложности программного обеспечения за счет выделения модулей, которые проектируются и изменяются независимо друг от друга. Изменение реализации модулей должно проводиться без знания реализации других модулей и без влияния на их поведение.

*Иерархия* (иерархическая организация) — это формирование из абстракций иерархической структуры, т. е. расположение их по уровням. Основными видами иерархических структур применительно к сложным системам являются:

- структура классов (один класс использует структурную или функциональную часть соответственно одного или нескольких других классов);

- структура объектов (агрегация, например структура типа «запись»).

Важное качество объектного подхода — согласованность моделей деятельности организации и моделей проектируемой информационной системы от стадии формирования требований до стадии реализации. По объектным моделям может быть прослежено отображение реальных сущностей моделируемой предметной области (организации) в объекты и классы информационной системы.

Объектно-ориентированный подход обладает следующими преимуществами.

1. Объектная декомпозиция дает возможность создавать модели меньшего размера путем использования общих механизмов, обеспечивающих необходимую экономию выразительных средств. Использование объектного подхода существенно повышает уровень унификации разработки и пригодность для повторного использования, что ведет к созданию среды разработки и переходу к сборочному созданию моделей.

2. Объектная декомпозиция позволяет избежать создания сложных моделей, так как она предполагает эволюционный путь развития модели на базе относительно небольших подсистем.

3. Объектная модель естественна, поскольку ориентирована на человеческое восприятие мира.

К недостаткам объектно-ориентированного подхода относятся высокие начальные затраты. Этот подход не дает немедленной отдачи. Эффект от его применения сказывается после разработки двух—трех проектов и накопления повторно используемых ком-

понентов. Диаграммы, отражающие специфику объектного подхода, менее наглядны.

### **4.3.2. UML — язык универсального моделирования**

Большинство существующих методов объектно-ориентированного подхода включают язык моделирования и описание процесса моделирования. В качестве языка моделирования в объектном подходе используется унифицированный язык моделирования UML, который содержит стандартный набор диаграмм для моделирования. В настоящее время UML является общепринятым стандартом документирования процесса создания информационных систем и их программного обеспечения.

Разработка универсального языка моделирования UML началась с середины 1990-х годов на базе нескольких объектно-ориентированных методов и нотаций описания информационных систем. Причиной, побудившей к созданию универсального языка описания программного обеспечения, явилась постоянно возрастающая сложность проектируемых информационных систем, которая в свою очередь диктуется усложнением решаемых задач. Когда количество объектов информационной системы не превышает 7—8 (психологический барьер, за которым человек не в состоянии оперировать информацией без дополнительных записей), сложности, возникающие при проектировании системы, преодолимы и без специальных средств. Такую информационную систему (для одного рабочего места или для небольшой компании) способен создать один человек. Когда же число объектов, состояний и переходов между ними доходит до нескольких тысяч, а то и миллионов, то ни один специалист, каким бы опытным и образованным он не был, не в состоянии охватить всю систему в целом. Такие информационные системы создаются группами разработчиков с разделением функциональных ролей. Тогда становится необходимым инструмент общения, каковым является UML.

Конструктивное использование языка UML основывается на понимании общих принципов моделирования сложных систем и особенностей процесса объектно-ориентированного анализа и проектирования, абстрагирования, многомодельности, иерархического построения моделей сложных систем. Эти принципы уже упоминались выше.

*Принцип абстрагирования* заключается в выделении существенных аспектов системы и отвлечении их от несущественных.

*Принцип многомодельности* представляет собой утверждение о том, что никакая единственная модель не может с достаточной степенью адекватности описывать различные аспекты сложной системы. Применительно к методологии объектно-ориентиро-



ванного анализа это означает, что достаточно полная модель сложной системы допускает некоторое число взаимосвязанных представлений, каждое из которых адекватно отражает некоторый аспект поведения или структуры системы.

*Принцип иерархического построения* предписывает рассматривать процесс построения модели на разных уровнях абстрагирования или детализации в рамках фиксированных представлений. При этом исходная, или первоначальная, модель сложной системы имеет наиболее общее представление (метапредставление). Такая модель строится на начальном этапе проектирования и может не содержать многих деталей и аспектов моделируемой системы. Концептуальная модель UML включает в себя три составные части: основные строительные блоки языка, правила их сочетания и некоторые общие для всего языка механизмы.

UML — это язык, состоящий из словаря и правил, позволяющих комбинировать входящие в него слова и получать осмысленные конструкции. В языке моделирования словарь и правила ориентированы на концептуальное и физическое представление системы. Словарь языка UML включает три вида строительных блоков: сущности, отношения, диаграммы.

Сущности — это абстракции, являющиеся основными элементами модели. Отношения связывают различные сущности. Диаграммы группируют представляющие интерес совокупности сущностей.

С помощью языка UML можно построить структурные модели и модели поведения.

Структуру сущностей или компонентов некоторой системы, включая их классы, интерфейсы, атрибуты и отношения, описывают структурные модели.

Модели поведения описывают поведение или функционирование объектов системы, включая их методы, взаимодействие и сотрудничество между ними, а также процесс изменения состояний отдельных компонентов и системы в целом. В рамках языка UML все представления о модели сложной системы фиксируются в виде специальных графических конструкций, получивших название диаграмм. В терминах языка UML определены следующие виды диаграмм:

- диаграмма вариантов использования (use case diagram) — для моделирования бизнес-процессов организации (требований к системе);

- диаграмма классов (class diagram) — для моделирования статической структуры классов системы и связей между ними;

- диаграммы поведения системы (behavior diagrams):

- 1) диаграмма состояний (statechart diagram) — для моделирования поведения объектов системы при переходе из одного состояния в другое;

2) диаграмма деятельности (activity diagram) — для моделирования поведения системы в рамках различных вариантов использования или моделирования деятельности;

3) диаграммы взаимодействия (interaction diagrams) — для моделирования процесса обмена сообщениями между объектами — диаграмма последовательности (sequence diagram), диаграмма кооперации (collaboration diagram);

– диаграммы реализации (implementation diagrams):

1) диаграмма компонентов (component diagram) — для моделирования иерархии компонентов (подсистем) системы;

2) диаграмма развертывания (deployment diagram) — для моделирования физической архитектуры системы.

Создание диаграмм аналогично созданию проекта в строительстве — можно обойтись и без него, если, например, строить сарай на дачном участке. Но выстроить большое здание гораздо сложнее. Без использования проекта будет неопределенным конечный результат. Информационная система, описанная UML-диаграммами, показывает разработчику результат, который необходимо достичь в процессе проектирования.

### **4.3.3. Диаграммы вариантов использования**

Разработку спецификаций информационной системы начинают с анализа требований к функциональности, указанных в техническом задании. В процессе этого анализа выявляют внешних пользователей разрабатываемого программного обеспечения ИС и перечень отдельных аспектов его поведения в процессе взаимодействия с конкретными пользователями. Аспекты поведения программного обеспечения были названы «вариантами использования», или «прецедентами» (use cases).

Вариант использования представляет собой последовательность действий, выполняемых системой в ответ на событие, инициируемое некоторым внешним объектом (действующим лицом), в качестве которого могут выступать не только люди, но и другие системы и устройства. Вариант использования описывает типичное взаимодействие между пользователем и системой.

Каждый вариант использования связан с некоторой целью, имеющей самостоятельное значение. Например, для текстового редактора «Формирование оглавления» — это вариант использования, а «Связывание заголовков со специальными стилями» — операция, которую необходимо выполнить, чтобы стало возможно автоматическое построение оглавления.

В зависимости от цели выполнения процедуры различают следующие варианты использования:

— основные (базовые) — обеспечивают требуемую функциональность разрабатываемого ПО;

— вспомогательные — обеспечивают выполнение необходимых настроек системы и ее обслуживание (например, архивирование информации и т. п.);

— дополнительные — обеспечивают дополнительные удобства для пользователя (как правило, реализуются в том случае, если не требуют серьезных затрат каких-либо ресурсов ни при разработке, ни при эксплуатации).

Следует отметить, что одним из требований языка UML является самодостаточность диаграмм для представления информации о моделях проектируемых систем. Однако изобразительных средств языка UML явно не хватает для того, чтобы учесть на диаграммах вариантов использования особенности функционального поведения сложной системы. С этой целью рекомендуется дополнять этот тип диаграмм текстовыми сценариями, которые уточняют или детализируют последовательность действий, совершаемых системой при выполнении ее вариантов использования.

В контексте языка UML сценарий используется для дополнительной иллюстрации взаимодействия актеров и вариантов использования. Предлагаются различные способы представления или написания подобных сценариев. Один из таких шаблонов рассматривается ниже и может быть рекомендован для применения на начальных этапах концептуального моделирования. В зависимости от уровня абстракции вариант использования может описываться кратко или более подробно. Краткая форма описания содержит название варианта использования, его цель, действующих лиц, тип варианта использования (основная, второстепенная или дополнительная) и его краткое описание. Ниже приводится шаблон для написания сценария отдельного варианта использования (табл. 4.2).

Таблица 4.2

Имя варианта использования	Типичный ход событий, приводящий к успешному выполнению варианта использования	Исключение 1	Примечание 1
Актеры		Исключение 2	Примечание 2
Цель		Исключение 3	Примечание 3
Краткое описание		...	...
Тип		...	...
Ссылки на другие варианты использования		Исключение n	Примечание n



Рис. 4.15. Основные условные обозначения диаграмм вариантов использования:

*а* — действующее лицо; *б* — вариант использования; *в* — связь

При написании сценариев вариантов использования важно понимать, что текст сценария должен дополнять или уточнять диаграмму вариантов использования, но не заменять ее полностью. В противном случае будут потеряны достоинства визуального представления моделей.

Диаграммы вариантов использования позволяют наглядно представить ожидаемое поведение системы. Основными понятиями диаграмм вариантов использования являются: действующее лицо, вариант использования и связь.

Условные обозначения, применяемые при изображении диаграмм вариантов использования, приведены на рис. 4.15.

*Действующее лицо (актер)* — внешняя по отношению к разрабатываемой системе сущность, которая взаимодействует с ней в целях получения или предоставления какой-либо информации. Как уже упоминалось выше, действующими лицами могут быть пользователи, другое ПО или какие-либо технические средства, взаимодействующее с системой.

*Вариант использования* в сценарии — некоторая очевидная для действующего лица процедура, решающая его конкретную задачу. Все варианты использования так или иначе связаны с требованиями к функциональности разрабатываемой системы и могут сильно различаться по объему выполняемой работы.

*Связь* — взаимодействие действующих лиц и соответствующих вариантов использования.

Варианты использования также могут быть связаны между собой. При этом фиксируют связи использования и расширения.

Использование (uses/include) подразумевает, что существует некоторый фрагмент поведения разрабатываемого ПО, который повторяется в нескольких вариантах использования. Этот фрагмент оформляют как отдельный вариант использования и указывают связь с ним типа «использование».

Расширение (extends) применяют, если имеются два подобных варианта использования, различающиеся наличием в одном из них некоторых дополнительных действий. В этом случае дополнитель-

ные действия определяют как отдельный вариант использования, который связан с основным вариантом связью типа «расширение».

Главное назначение диаграммы вариантов использования заключается в формализации функциональных требований к системе и возможности согласования полученной модели с заказчиком на ранней стадии проектирования. Любой из вариантов использования может быть подвергнут дальнейшей декомпозиции на множество подвариантов использования отдельных элементов, которые образуют исходную сущность.

Для иллюстрации особенностей спецификации функциональных требований на диаграмме вариантов использования можно рассмотреть модель системы «Склад оптовой торговли». Для первоначального понимания структуры программной системы выявляются действующие лица (люди или системы, между которыми происходит взаимодействие). Рассматриваемая система имеет пять актеров, двое из которых являются контрагентами, а другие менеджерами склада, осуществляющими выполнение всех операций. Каждый из этих актеров взаимодействует с системой, хотя главными актерами, пожалуй, являются поставщики и покупатели (контр-

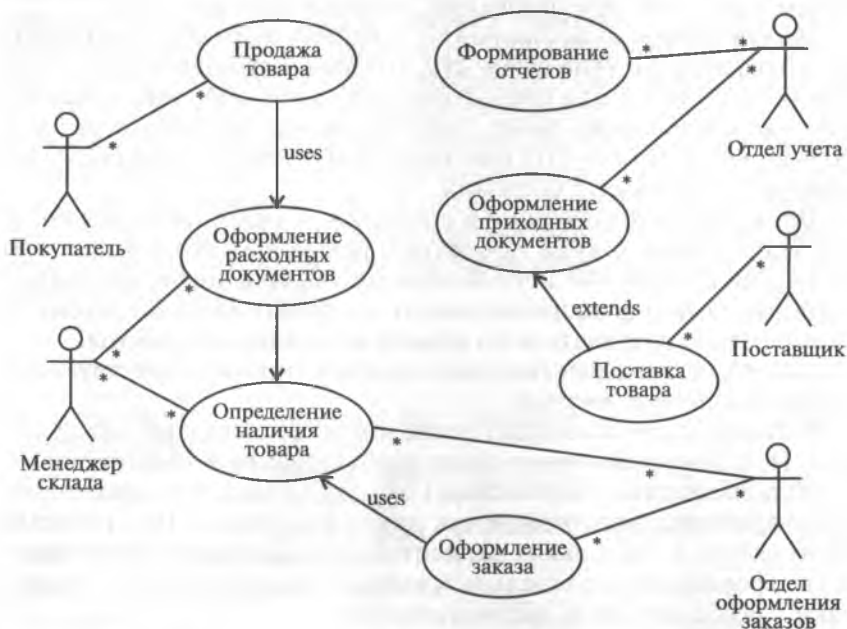


Рис. 4.16. Диаграмма вариантов использования для проектирования программного обеспечения АИС «Склад оптовой торговли»

агенты), поскольку именно они инициируют функциональность системы. Далее формулируются варианты использования, т. е. действия, выполняемые системой для реализации общения действующих лиц (актеров).

Каждый из актеров преследует определенные цели по отношению к системе: поставщик — сдать товар на склад, покупатель — приобрести товар, менеджер склада — принять и отпустить товар, менеджер учетного отдела — определить объемы поступления и продаж и проанализировать товарный запас. На основании этих целей можно сформулировать базовые варианты использования и проанализировать взаимосвязи между ними (рис. 4.16).

На самом деле вариантов использования может быть гораздо больше. Например, проверить платежеспособность клиента, получить информацию о товаре, оценить запасы товара на складе, получить оплату, вывести информацию на экран и т. д. Однако эта диаграмма дает понять, что будет делать система, как она будет функционировать.

На следующем этапе разработки модели вариантов использования для рассматриваемой системы следует дополнить данную диаграмму текстовым сценарием, написанным на основе предложенного ранее шаблона. Этот сценарий будет дополнять диаграмму, раскрывая содержание и логическую последовательность отдельных действий, которые выполняются системой и актерами в процессе поступления и реализации товаров. В этом случае сценарий удобно представить в виде таблиц, каждая из которых описывает отдельный раздел шаблона.

Таблица 4.3

Вариант использования	Продажа товара
Актеры	Покупатель. Менеджер отдела оформления заказов. Менеджер склада
Краткое описание	Покупатель запрашивает товар. Менеджер отдела оформления заказов резервирует товар, оформляет заказ, передает заказ менеджеру склада. Покупатель оплачивает товар, получает товар на складе
Цель	Получение необходимого товара
Тип	Базовый
Ссылки на другие варианты использования	Включает в себя варианты использования: определить наличие товара; оформить заказ

Таблица 4.4

Действия актеров	Отклик системы
1. Покупатель запрашивает товар	2. Менеджер отдела оформления заказов проверяет наличие необходимого товара на складе 3. Менеджер отдела оформления заказов резервирует нужный товар
Исключение № 1. На складе нет необходимого количества запрашиваемого товара	
4. Покупатель оплачивает товар	5. Менеджер отдела оформления заказов выдает разрешение на получение товара 6. Менеджер отдела оформления заказов передает заказ на склад 7. Менеджер склада выдает товар и расходную накладную покупателю
Исключение № 2. Покупатель не оплатил товар	

В главном разделе сценария (табл. 4.3) указываются имя рассматриваемого варианта использования, имена взаимосвязанных с ним актеров, цель выполнения варианта, условный тип и ссылки на другие варианты использования.

В следующем разделе сценария (табл. 4.4) изложена последовательность действий, приводящая к успешному выполнению рассматриваемого варианта использования. При этом инициатором действий должен выступать актер Покупатель. Для удобства последующих ссылок каждое действие помечается порядковым номером.

В третьем разделе сценария (табл. 4.5) описывается последовательность действий, выполняемых при возникновении исключительных ситуаций или исключений.

Таблица 4.5

Действия актеров	Отклик системы
Исключение № 1. На складе нет необходимого количества запрашиваемого товара	
4. Покупатель оплачивает товар	3. Менеджер отдела оформления заказов инициирует поставку нужного товара
Исключение № 2. Покупатель не оплатил товар	
	6. Менеджер оформления заказов блокирует получение товара покупателем

Можно дополнить данный сценарий, аналогичным образом описав не только варианты использования «Оформление заказа» и «Определение наличия товара», но и рассмотрев другие исключения, например оформление скидок постоянным покупателям, и т. п. При этом полнота сценариев и модели вариантов использования будет определяться теми функциональными требованиями, которые сформулированы в рамках конкретного проекта.

Отдельные, небольшие по своему объему, сценарии могут быть размещены на диаграмме в форме примечаний. Примечание (note) предназначено для включения в модель произвольной текстовой информации, имеющей непосредственное отношение к контексту разрабатываемого проекта. В качестве такой информации могут быть комментарии разработчика (например, дата и версия разработки диаграммы или ее отдельных компонентов), ограничения (например, на значения отдельных связей или экземпляры сущностей) и помеченные значения. Применительно к диаграммам вариантов использования примечание может иметь уточняющую информацию, относящуюся к контексту тех или иных вариантов использования.

Графически примечания на всех типах диаграмм обозначаются прямоугольником с «загнутым» верхним правым углом (рис. 4.17).

Собственно текст примечания размещается внутри этого прямоугольника. Примечание может относиться к любому элементу диаграммы, в этом случае их соединяет пунктирная линия. Если примечание относится к нескольким элементам, то от него проводят соответственно несколько линий. Как уже отмечалось, примечания могут присутствовать не только на диаграмме вариантов использования, но и на других канонических диаграммах.

Поскольку одно из главных назначений диаграммы вариантов использования — формализация функциональных требований к системе, она может служить основой для согласования с заказчи-

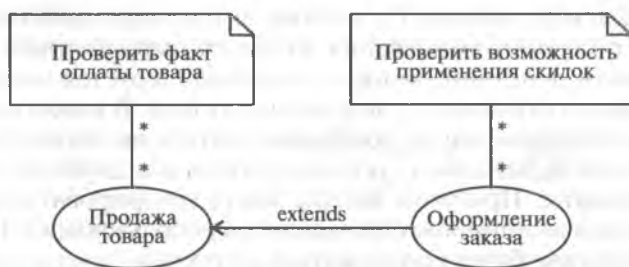


Рис. 4.17. Пример примечаний на диаграммах вариантов использования



ком этих требований на ранней стадии проектирования. Любой из базовых вариантов использования в последующем может быть подвергнут декомпозиции на частные варианты использования. При этом рекомендуется, чтобы общее количество актеров в модели не превышало 20, а вариантов использования — 50. В противном случае модель теряет свою наглядность и, возможно, заменяет собой одну из некоторых других диаграмм.

Для разработки диаграммы вариантов использования рекомендуется некоторая последовательность действий:

- определить главных или первичных и второстепенных актеров;
- определить цели главных актеров по отношению к системе;
- сформулировать основные варианты использования, которые специфицируют функциональные требования к системе;
- упорядочить варианты использования по степени убывания риска их реализации;
- рассмотреть все базовые варианты использования в порядке убывания их степени риска;
- выделить участников, интересы, предусловия и постусловия выполнения выбранного варианта использования;
- написать успешный сценарий реализации выбранного варианта использования;
- определить исключения или неуспех в выполнении сценария варианта использования;
- написать сценарии для всех исключений;
- выделить общие варианты использования и изобразить их взаимосвязи с базовыми со стереотипом `uses/include`;
- выделить варианты использования для исключений и изобразить их взаимосвязи с базовыми со стереотипом `extend`;
- проверить диаграмму на отсутствие дублирования вариантов использования и актеров.

Семантика построения диаграммы вариантов использования должна определяться следующими особенностями рассмотренных выше элементов модели. Отдельный экземпляр варианта использования по своему содержанию является выполнением последовательности действий, которая инициализируется посредством экземпляра сообщения от экземпляра актера. В качестве отклика или ответной реакции на сообщение актера выполняется последовательность действий, установленная для данного варианта использования. При этом актеры могут генерировать новые сообщения для инициирования вариантов использования. Подобное взаимодействие будет продолжаться до тех пор, пока не закончится выполнение требуемой последовательности действий экземпляром варианта использования и указанный в модели экземпляр актера не получит требуемый экземпляр сервиса. Окончание

взаимодействия означает отсутствие инициализации сообщений от актеров для базовых вариантов использования.

Варианты использования могут быть дополнительно специфицированы примечаниями с текстом, которые в последующем могут стать прототипами операций и методов совместно с атрибутами. Дальнейшая разработка моделей связана с реализацией вариантов использования в виде графа деятельности, посредством конечного автомата или любого другого механизма логического представления поведения, включающего предусловия и постусловия. Взаимодействие между вариантами использования и актерами может уточняться на диаграмме кооперации, когда описываются взаимосвязи между системой, содержащей эти варианты использования, и окружением или внешней средой этой системы.

#### 4.3.4. Диаграммы деятельности

Если диаграмма вариантов использования дает «вид сверху» на функциональность системы, диаграмма деятельности, напротив, позволяет подробно иллюстрировать отдельный вариант использования и его сценарии.

Под деятельностью в данном случае понимают задачу (операцию), которую необходимо выполнить вручную или с помощью средств автоматизации. Каждому варианту использования соответствует своя последовательность задач. В теоретическом плане диаграммы деятельности — обобщенное представление алгоритма, реализующего анализируемый вариант использования. На диаграмме деятельность обозначается прямоугольником с закругленными углами (рис. 4.18, *а*).

Диаграммы деятельности позволяют описывать альтернативные и параллельные процессы. Для обозначения альтернативных процессов используют ромб (рис. 4.18, *б*), условие указывают рядом, а альтернативы «да», «нет» — рядом с соответствующими выходами. С помощью этого же блока можно построить циклический

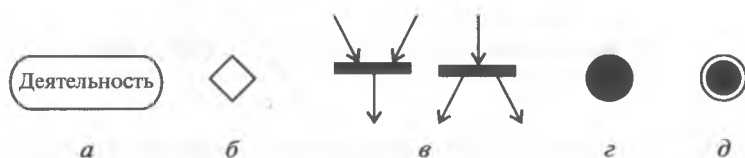


Рис. 4.18. Условные обозначения диаграммы деятельностей:

*а* — деятельность; *б* — выбор; *в* — линейки синхронизации; *г* — начало; *д* — конец

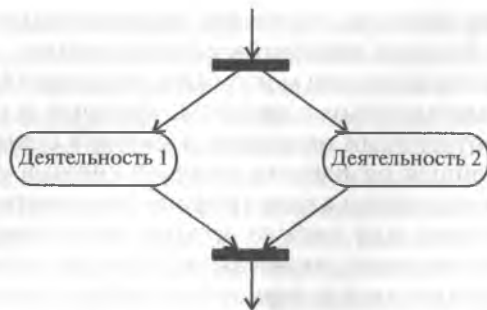


Рис. 4.19. Пример диаграммы деятельности с указанием параллельности процессов

процесс. Множественность активации деятельности обозначают символом «\*», помещенным рядом со стрелкой активации деятельности, и при необходимости уточняют надписью вида «для каждой строки».

Для обозначения параллельных процессов используют линии синхронизации (рис. 4.18, в), причем условие синхронизации можно уточнить, указав его на диаграмме (рис. 4.19).

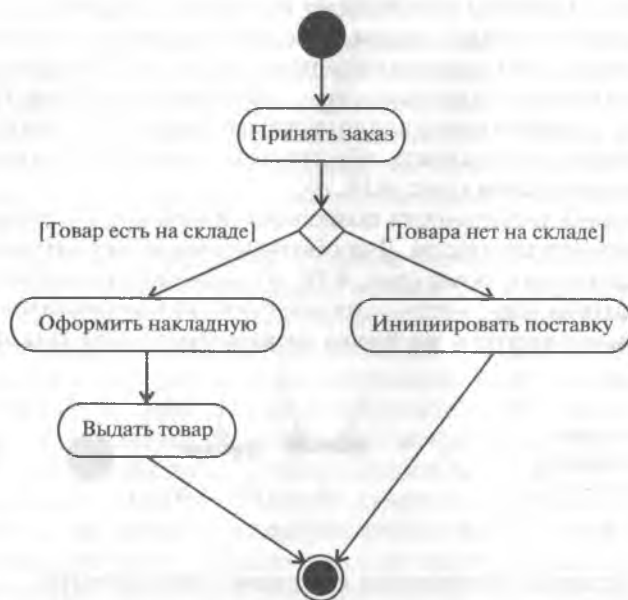


Рис. 4.20. Пример диаграммы деятельности

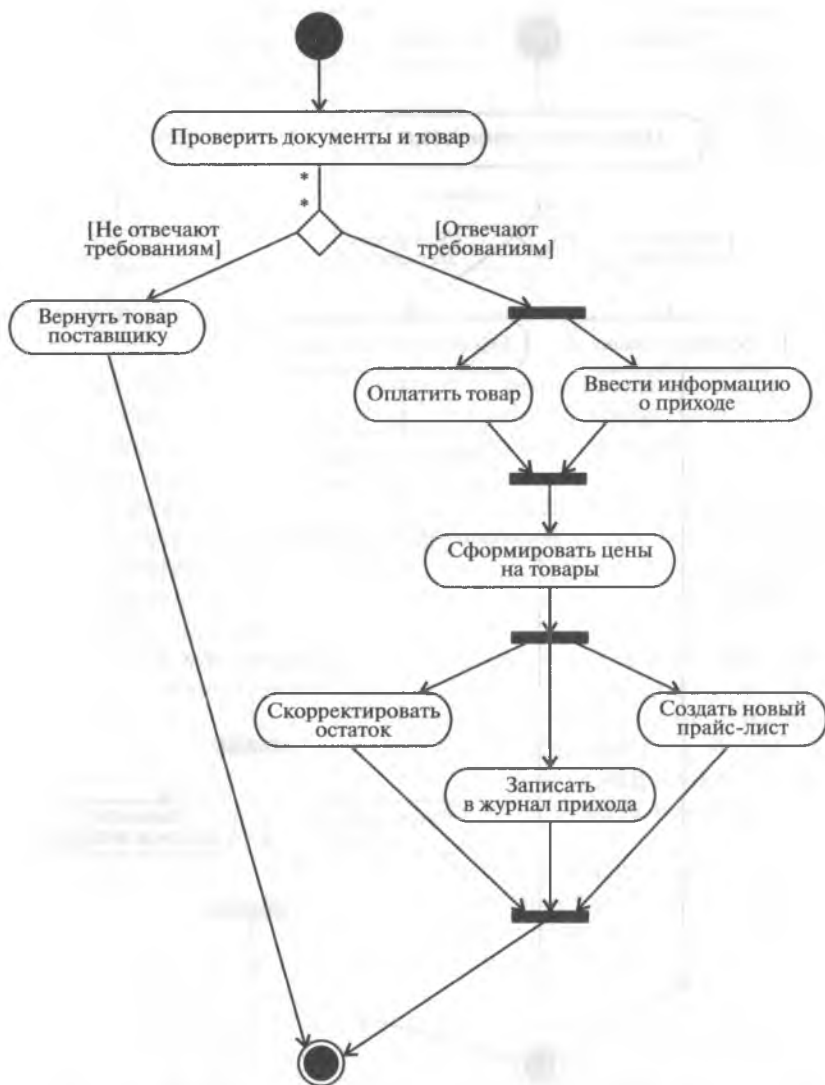


Рис. 4.21. Диаграмма деятельности для варианта использования «Поставка товара»

Теперь рассмотрим пример построения диаграммы деятельности. В предыдущем примере для конкретизации описания вариантов использования в АИС «Склад оптовой торговли» был разработан текстовый сценарий. Этот сценарий дополняет диаграмму, раскрывая содержание отдельных действий, выполняемых

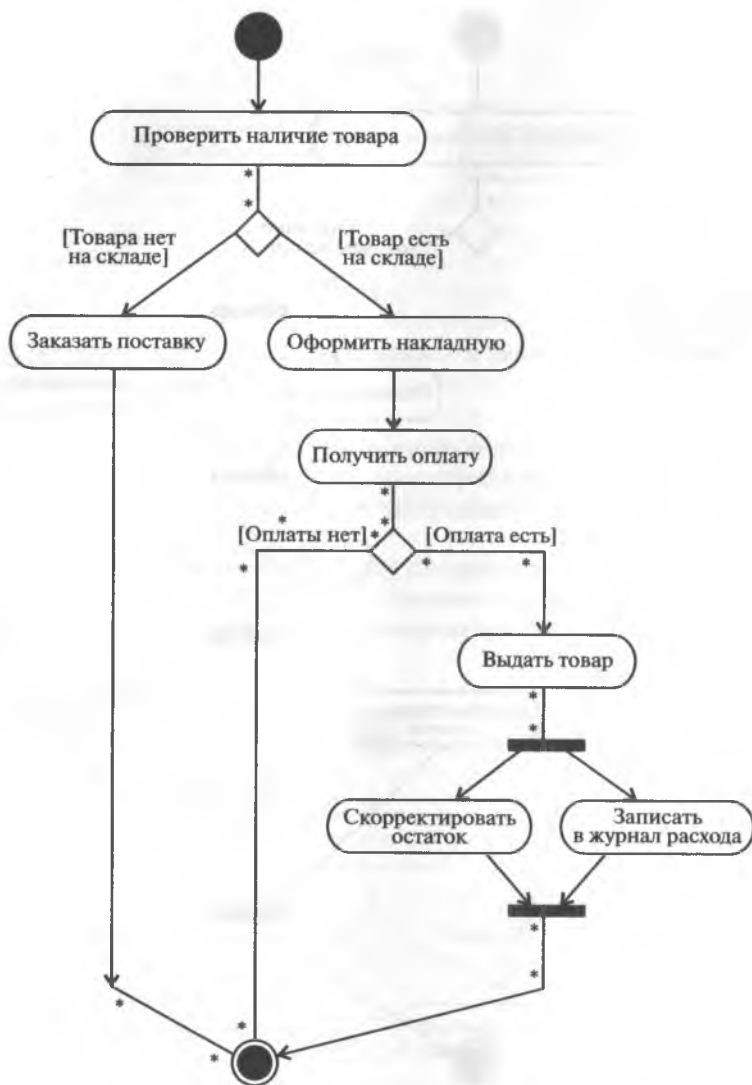


Рис. 4.22. Диаграмма деятельности для варианта использования «Продажа товара»

системой и актерами. Однако вместо описания вариантов использования или в дополнение к ним можно использовать диаграммы деятельности. Диаграмма деятельности позволяет проиллюстрировать вариант использования с требуемой степенью подробности. Имеет смысл уточнять только те варианты использования, краткое

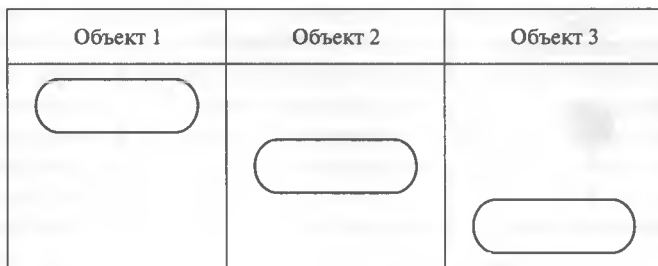


Рис. 4.23. Вариант диаграммы деятельности с дорожками

описание которых недостаточно для понимания сущности решаемых проблем. Диаграммы деятельностей можно использовать вместо текстового описания вариантов использования или в дополнение к ним.

В рамках разрабатываемой модели построим диаграммы деятельности для реализации вариантов использования «Продажа товара» и «Поставка товара». На рис. 4.20 показан вариант диаграммы деятельности для описания последовательности действий при поставке товаров.

Однако диаграмма деятельности позволяет проиллюстрировать вариант использования с различной степенью подробности, и диаграмму деятельности для варианта использования «Поставка товара» можно представить следующим образом (рис. 4.21).

Проанализировав вариант использования «Продажа товара», построим диаграмму деятельности для варианта использования «Продажа товара» (рис. 4.22).

Полная модель системы может содержать несколько диаграмм деятельности, каждая из которых описывает последовательность реализации либо наиболее важных вариантов использования (типичный ход событий и все исключения), либо нетривиальных операций классов.

Помимо стандартного формата описания UML предлагает вариант с «плавательными дорожками». Этот формат удобен для описания случая, когда в варианте использования участвуют несколько действующих лиц. При этом все состояния действия на диаграмме деятельности делятся на отдельные группы, отделяющие друг от друга вертикальными линиями (рис. 4.23). Применение дорожек открывает дополнительные возможности для наглядного представления бизнес-процессов, позволяя специфицировать деятельность подразделений предприятий (рис. 4.24).

В случае типового проекта большинство деталей реализации действий могут быть известны заранее на основе анализа существующих систем или предшествующего опыта разработки систем-

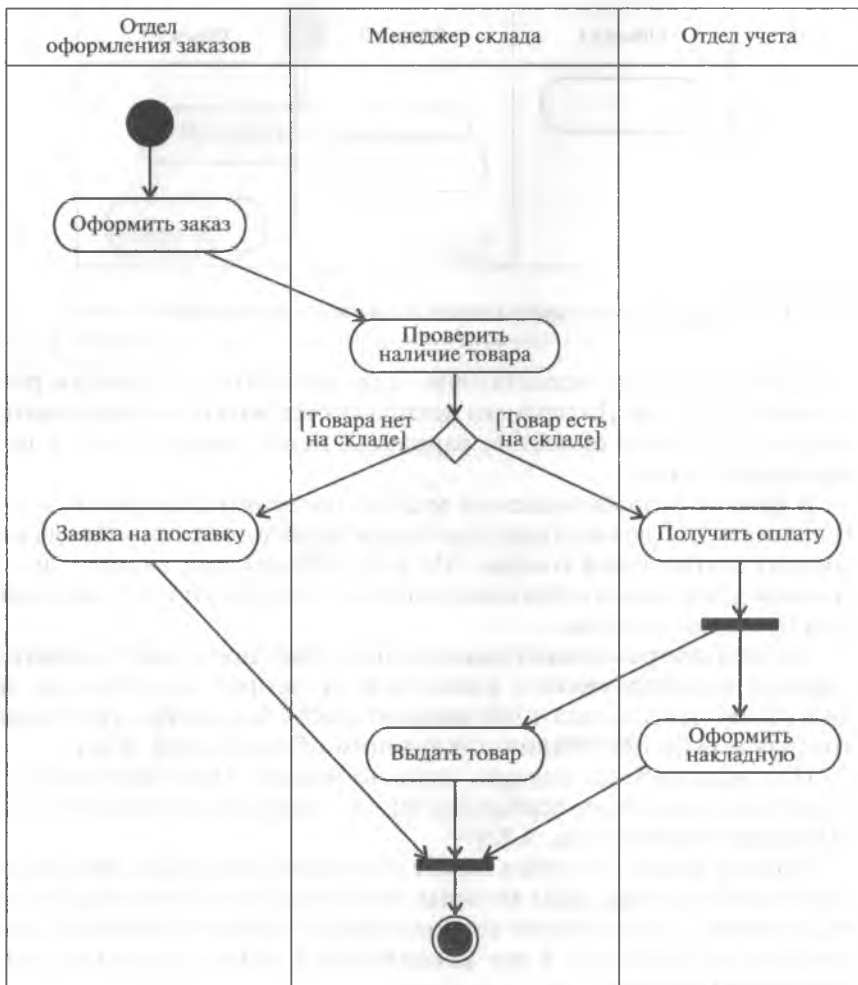


Рис. 4.24. Пример применения диаграммы деятельности с дорожками для варианта использования «Оформление заказа»

прототипов. Использование типовых решений может существенно сократить время разработки и избежать возможных ошибок при реализации проекта.

#### 4.3.5. Диаграммы последовательности

Рассмотренные диаграммы деятельности используются для спецификации динамики поведения систем, время в явном виде

в них не присутствует. Однако временной аспект поведения может иметь существенное значение при моделировании синхронных процессов, описывающих взаимодействия объектов. Именно для этой цели в языке UML используются диаграммы последовательности — графические модели, которые для определенного сценария варианта использования показывают динамику взаимодействия объектов во времени.

Для построения диаграммы последовательностей системы необходимо:

- идентифицировать каждое действующее лицо (объект) и изобразить для него линию жизни;
- из описания варианта использования определить множество системных событий и их последовательность;
- изобразить системные события в виде линий со стрелкой на конце между линиями жизни действующих лиц и системы, а также указать имена событий и списки передаваемых значений.

На диаграмме последовательности изображаются исключительно те объекты, которые непосредственно участвуют во взаимодействии и не показываются возможные статические ассоциации с другими объектами. Для диаграммы последовательности ключевым моментом является именно динамика взаимодействия объектов во времени. При этом диаграмма последовательности имеет как бы два измерения.

Первое измерение — слева направо в виде вертикальных линий, каждая из которых изображает линию жизни отдельного объекта, участвующего во взаимодействии. Графически каждый объект изображается прямоугольником и располагается в верхней части своей линии жизни. Внутри прямоугольника записывают имя объекта и имя класса, разделенные двоеточием. При этом вся запись подчеркнута, что является признаком объекта, который, как известно, представляет собой экземпляр класса (рис. 4.25).

Не исключается ситуация, когда имя объекта может отсутствовать на диаграмме последовательности. В этом случае указывается только имя класса, а сам объект считается анонимным.

Крайним слева на диаграмме изображается объект, который является инициатором взаимодействия (объект 1). Правее изображается другой объект, который непосредственно взаимодействует с первым. Таким образом, все объекты на диаграмме последовательности образуют некоторый порядок, определяемый степенью активности этих объектов при взаимодействии друг с другом.

Второе измерение диаграммы последовательности — вертикальная временная ось, направленная сверху вниз. Начальному моменту времени соответствует самая верхняя часть диаграммы. При этом взаимодействия объектов реализуются посредством со-



общений, которые посылаются одними объектами другим. Сообщения изображаются в виде горизонтальных стрелок с именем сообщения. Кроме того, их показывают в определенном порядке в соответствии со временем своего возникновения. Другими словами, сообщения, расположенные на диаграмме последовательности выше, инициируются раньше тех, которые расположены ниже. При этом масштаб на оси времени не указывается, поскольку диаграмма последовательности моделирует лишь временную упорядоченность взаимодействий типа «раньше-позже».

Линия жизни объекта служит для обозначения периода времени, в течение которого объект существует в системе и, следовательно, может потенциально участвовать во всех ее взаимодействиях. На диаграмме линия жизни изображается пунктирной вертикальной линией, ассоциированной с единственным объектом. Если объект существует в системе постоянно, то и его линия жизни должна продолжаться по всей плоскости диаграммы последовательности.

Построение диаграммы последовательности целесообразно начинать с выделения из всей совокупности тех и только тех классов, объекты которых участвуют в моделируемом взаимодействии. После этого все объекты наносят на диаграмму с соблюдением некоторого порядка инициализации сообщений. Здесь необходимо установить, какие объекты будут существовать постоянно, а какие временно — только на период выполнения ими требуемых действий. Когда объекты определены, можно приступить к спецификации сообщений. При этом следует учитывать те роли, которые играют сообщения в системе.



Рис. 4.25. Различные графические примитивы диаграммы последовательности

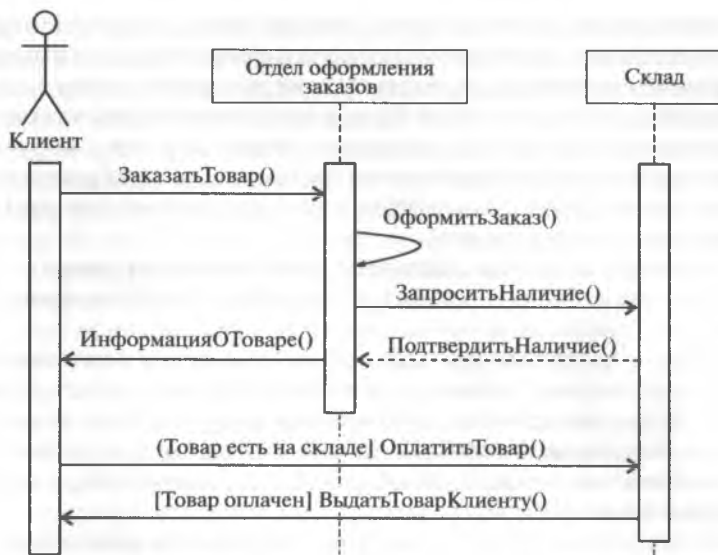


Рис. 4.26. Вариант диаграммы последовательности для моделирования операции продажи товара

В качестве примера построим диаграмму последовательности для реализации варианта использования «Продажа товара» в АИС «Склад оптовой торговли» (рис. 4.26).

Данная диаграмма содержит два объекта и одного актера. Объекты не являются постоянно активными, что показано с помощью соответствующих фокусов управления. В качестве имен сообщений указаны имена операций, которые специфицированы у соответствующих классов. Предложения-условия некоторых сообщений записаны обычным текстом в квадратных скобках. Эти условия отражают возможность ветвления процесса продажи и выполнения исключительного сценария соответствующего варианта использования, однако другие варианты использования на данной диаграмме не показаны.

#### 4.4. Сравнение структурного и объектно-ориентированного методов моделирования

В функциональных моделях (DFD-диаграммах потоков данных, IDEF-диаграммах) главными структурными компонентами являются функции (операции, действия, работы), которые на диаграммах связываются между собой потоками информации.

Несомненным достоинством функциональных моделей является реализация структурного подхода к проектированию информационной системы по принципу «сверху-вниз», когда каждый функциональный блок может быть декомпозирован на множество подфункций, задач и т. д., выполняя, таким образом, модульное проектирование информационной системы. Для функциональных моделей характерны процедурная строгость декомпозиции ИС и наглядность представления.

При функциональном подходе объектные модели данных в виде ER-диаграмм «объект — свойство — связь» разрабатываются отдельно.

Главный недостаток функциональных моделей заключается в том, что процессы и данные существуют отдельно друг от друга — помимо функциональной декомпозиции существует структура данных, находящаяся на втором плане. Кроме того, не ясны условия выполнения процессов обработки информации, которые динамически могут изменяться.

Перечисленные недостатки функциональных моделей отсутствуют в объектно-ориентированных моделях, где главным структурообразующим компонентом выступает класс объектов с набором функций, которые могут обращаться к атрибутам этого класса. Для классов объектов характерна иерархия обобщения, позволяющая осуществлять наследование не только атрибутов (свойств) объектов от вышестоящего класса объектов к нижестоящему классу, но и функций (методов).

В случае наследования функций можно абстрагироваться от конкретной реализации процедур (абстрактные типы данных), которые различаются для определенных подклассов ситуаций. Это дает возможность обращаться к подобным программным модулям по общим именам (полиморфизм) и повторно использовать программный код при модификации программного обеспечения. Таким образом, адаптивность объектно-ориентированных систем к изменению предметной области по сравнению с функциональным подходом значительно выше.

При объектно-ориентированном подходе изменяется и принцип проектирования ИС. Сначала выделяют классы объектов, а далее в зависимости от возможных состояний объектов (жизненного цикла объектов) определяют методы обработки (функциональные процедуры), что обеспечивает наилучшую реализацию динамического поведения информационной системы.

Для объектно-ориентированного подхода разработаны графические методы моделирования предметной области, обобщенные в языке унифицированного моделирования UML. Однако по наглядности представления модели пользователю-заказчику объектно-ориентированные модели явно уступают функциональным моделям.

При выборе методики моделирования предметной области обычно в качестве критерия выступает степень ее динамичности. Для регламентированных задач больше подходят функциональные модели, для более адаптивных бизнес-процессов (управления рабочими потоками, реализации динамических запросов к информационным хранилищам) — объектно-ориентированные. Однако в рамках одной и той же ИС для различных классов задач могут требоваться различные виды моделей, описывающих одну и ту же проблемную область. В таком случае нужно использовать комбинированные модели предметной области.

Таким образом, каждая из рассмотренных методик позволяет решить задачу построения формального описания рабочих процедур исследуемой системы. Все методики позволяют построить модели «как есть» и «как должно быть». В то же время каждая из этих методик обладает существенными недостатками. В целом их можно охарактеризовать следующим образом: недостатки применения отдельной методики лежат не в области описания реальных процессов, а в неполноте методического подхода.

Функциональные методики нагляднее дают представление о существующих функциях в организации, о методах их реализации. Причем, чем выше степень детализации исследуемого процесса, тем лучше они позволяют описать систему. Под лучшим описанием в данном случае понимается наименьшая ошибка при попытке по полученной модели предсказать поведение реальной системы. На уровне отдельных рабочих процедур их описание практически однозначно совпадает с фактической реализацией в потоке работ.

На уровне общего описания системы функциональные методики допускают значительную степень свободы в выборе общих интерфейсов системы, ее механизмов и т. д., т. е. в определении границ системы. Хорошо описать систему на этом уровне позволяет объектный подход, основанный на понятии сценария использования. Ключевым является понятие о сценарии использования как о сеансе взаимодействия действующего лица с системой, в результате которого действующее лицо получает нечто, имеющее для него ценность. Использование критерия ценности для пользователя дает возможность отбросить не имеющие значения детали потоков работ и сосредоточиться на тех функциях системы, которые оправдывают ее существование. Однако и в этом случае задача определения границ системы, выделения внешних пользователей является сложной.

Технология потоков данных, исторически возникшая первой, легко решает проблему границ системы, поскольку дает возможность за счет анализа информационных потоков выделить внешние сущности и определить основной внутренний процесс. Одна-

ко отсутствие выделенных управляющих процессов, потоков и событийной ориентированности не позволяет предложить эту методику в качестве единственной.

Наилучшим способом преодоления недостатков рассмотренных методик является формирование синергетической методики, объединяющей различные этапы отдельных методик. При этом из каждой методики необходимо взять часть методологии, наиболее полно и формально изложенную, и обеспечить возможность обмена результатами на различных этапах применения синергетической методики.

### **Контрольные вопросы**

1. Каковы цели моделирования предметной области?
2. Назовите три уровня построения моделей предметной области.
3. Какой существует подход к решению проблемы сложных систем?
4. Объясните сущность структурного подхода к разработке ИС.
5. В чем сущность объектно-ориентированного подхода?
6. На каких принципах базируются методологии структурного подхода?
7. Какие требования предъявляют к модели предметной области?
8. Назовите особенности методологии SADT.
9. Какие компоненты составляют диаграмму потоков данных (DFD)?
10. Для чего предназначена диаграмма «сущность-связь»?
11. Назовите базовые понятия ER-модели данных.
12. Что является концептуальной основой объектно-ориентированного подхода?
13. Назовите основные элементы объектной модели.
14. В чем заключается принцип абстрагирования?
15. Что представляет собой принцип многомодельности?
16. Что представляет собой принцип иерархического построения систем?
17. Какие виды диаграмм определены в UML?
18. Назовите преимущества объектно-ориентированный подхода.
19. Что представляет собой вариант использования в UML?
20. Почему диаграммы вариантов использования рекомендуется дополнять текстовыми сценариями?
21. Для чего используют диаграммы деятельности?
22. В чем особенность диаграммы последовательности системы?
23. Дайте сравнительную характеристику структурного и объектно-ориентированного методов моделирования.

## **ИНФОРМАЦИОННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ**

### **5.1. Понятие информационной технологии управления**

Наиболее очевидным способом повышения эффективности управления бизнес-процессами на предприятии является использование новейших методов работы с информацией, внедрение методов управления, использующих современные информационные технологии.

Информационная технология управления — совокупность методов и средств сбора, регистрации, передачи, накопления и обработки информации на базе программно-аппаратного обеспечения для решения управленческих задач. Основная цель информационной технологии управления — преобразовать первичные данные для получения информации нового качества, на основе которой вырабатываются оптимальные управленческие решения.

Информационные технологии управления в их современном понимании представляют собой весьма важную и сложную область управленческой теории и практики. Само понятие управления в наши дни приобрело особенные черты. Наука управления сегодня непосредственно связана с такими дисциплинами, как системный анализ, исследование операций, теория принятия решений, математическое моделирование и др. Внедрение в управленческую деятельность исследовательского подхода базируется на применении современных достижений в области информационных технологий, обеспечивающих полноту, своевременность информационного отображения управляемых процессов, возможность их моделирования, анализа, прогнозирования. Но то что действительно для строго формализованного процесса, отнюдь не очевидно для такой сложной сферы, как управление. Трудности, возникающие при решении задачи автоматизированной поддержки управления, связаны с его спецификой. Управленческий труд отличается сложностью и многообразием, наличием большого числа форм и видов,

многосторонними связями с различными явлениями и процессами. Это прежде всего творческий и интеллектуальный труд. На первый взгляд, большая его часть вообще не поддается какой-либо формализации. Поэтому автоматизация управленческой деятельности изначально связывалась только с автоматизацией некоторых вспомогательных, рутинных операций. Но бурное развитие информационных технологий, совершенствование технической платформы и появление принципиально новых классов программных продуктов привело в наши дни к изменению подходов к автоматизации управления предприятием. При рассмотрении технологии управления необходимо четко разделять три понятия: система управления, объект управления, внешняя среда, воздействующая на объект управления (рис. 5.1)

Под системой управления подразумевается совокупность информации, экономико-математических методов и моделей, технических, программных, других технологических средств, предназначенных для обработки информации и принятия управленческих решений.

Объектом управления является некий экономический субъект — предприятие, организация, подразделение и т. д.

Внешняя среда — это внешние факторы, влияющие на экономическое состояние субъекта (законодательство, стабильность государственной экономической политики, состояние рынка сбыта продукции или услуг, обеспеченность поставками и т. д.).

Одним из основных элементов информационных систем управления является лицо, принимающее решение (ЛПР), — человек или группа людей, оказывающих управляющее воздействие на объект.

В информационных системах управления на нижних уровнях действует детальная информация о состоянии объекта управления. По мере движения информации от нижних уровней управления к высшим она должна быть избавлена от лишних подроб-

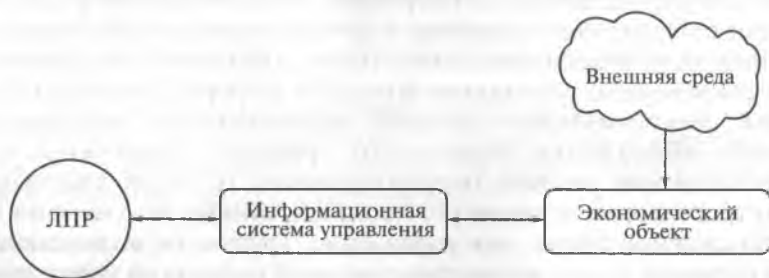


Рис. 5.1. Общая схема процесса управления

ностей, бесполезной детализации. Этот процесс называется интеграцией или сжатием информации. Первичная информация — информация о состоянии внешней среды и объекта управления. Информация, получаемая в результате обработки первичной информации (сортировки, группировки, выделения, вычисления и т.д.), называется промежуточной, или вторичной, информацией. Информация, полученная для выработки управляющих воздействий, называется управляющей, требующей реакции объекта управления.

Основные функции системы управления — функции принятия решений, рутинные функции обработки информации, функции обмена информацией.

*Функции принятия решений* выражаются в создании новой информации в ходе анализа, планирования (прогнозирования) и оперативного управления. Это связано с преобразованием информации о состоянии объекта управления и внешней среды в управляющую информацию при решении логических задач и выполнении аналитических расчетов, проводимых ЛПР. Эта группа функций является главной, поскольку обеспечивает выработку информационных воздействий для перевода экономического объекта в новое состояние или удержания в существующем состоянии. Без автоматизации этой функции информационная система не может считаться полноценной.

*Рутинные функции обработки информации* охватывают учет, контроль, хранение, поиск, отображение информации и т.д. Эта группа функций преобразует информацию, не изменяя ее смысл, т.е. это функции, не связанные с содержательной обработкой информации.

*Функции обмена информацией* связаны с доведением выработанных воздействий до объекта управления и обменом информацией между лицами, принимающими решения (ограничение доступа, сбор, передача информации по управлению в различных формах).

Совокупность функций управления, выполняемых в системе при изменении состояния внешней среды, принято называть *циклом управления*. Выполняя цикл за циклом, система приближается к сформулированной цели. Одно из представлений цикла управления показано на рис 5.2.

От экономического объекта в систему управления поступает информация о текущем состоянии дел. ЛПР контролируют ее истинность, учитывают и анализируют ее в целях выявления отклонений от требуемого состояния и оценки текущего состояния. Учет является необходимым дополнительным средством контроля. По результатам анализа выбирают задачи управления, осуществляется выработка решений по удержанию объекта в требуемом



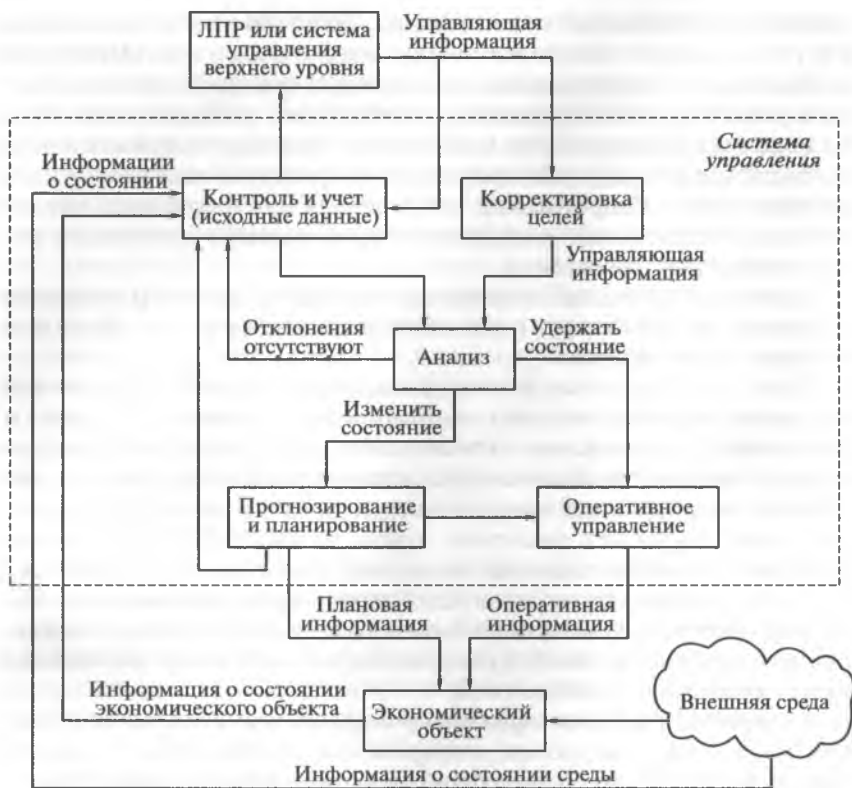


Рис. 5.2. Цикл управления

состоянии или корректируются цели, после чего объект управления переводится в новое состояние на основе прогнозирования и планирования. Информация о состоянии экономического объекта передается лицам, принимающим решения (менеджерам, руководителям) или системе управления верхнего уровня. Используя оперативную информацию, полученную в ходе функционирования автоматизированной информационной системы, руководитель может спланировать и сбалансировать ресурсы предприятия (материальные, финансовые, кадровые), просчитать и оценить результаты управленческих решений, наладить оперативное управление сбалансированностью производства (товаров, услуг), ходом выполнения плана, использованием ресурсов и т. д.

Информационная технология управления должна удовлетворять информационные потребности менеджеров, принимающих решения по повышению эффективности деятельности предприятия. Она применима на любом уровне управления. Информация

должна быть представлена в агрегированном виде так, чтобы просматривались тенденции изменения данных, причины возникших отклонений и возможные решения. Для принятия решения необходимо оценить планируемое состояние объекта управления. В рамках данной оценки определяют отклонения от планируемого состояния и выявляют причины. После чего анализируют возможные решения и действия.

Принятие решения может осуществляться менеджером двумя способами:

- на основе анализа информации из управленческих отчетов;
- выбором из возможных альтернативных решений.

Информационные системы, создающие управленческие отчеты, ориентированы главным образом на обработку данных (поиск, сортировку, агрегирование, фильтрацию). Они обеспечивают информационную поддержку пользователя, т.е. предоставляют доступ к информации в базе данных и ее частичную обработку. Используя сведения, содержащиеся в этих отчетах, управляющий принимает решение. Регулярные отчеты создаются в соответствии с установленным графиком, определяющим время их создания (например, анализ продаж компании за декаду или месяц). Специальные отчеты создаются по запросам менеджеров или в случаях, когда на предприятии (в организации) произошло что-то незапланированное. И те, и другие виды отчетов могут иметь форму суммирующих, сравнительных и чрезвычайных отчетов. Суммирующие отчеты содержат данные, объединенные в отдельные группы, отсортированные и представленные в виде промежуточных и окончательных итогов по отдельным полям. Сравнительные отчеты содержат данные, полученные из разных источников или классифицированные по нескольким признакам и используемые для целей сравнения. Чрезвычайные отчеты содержат данные исключительного (чрезвычайного) характера.

Принятие решений в информационных системах, разрабатывающих альтернативы решений, сводится к выбору одной из предложенных альтернатив. Система предлагает различные варианты готовых решений в виде моделей или описаний, из которых менеджеру необходимо выбрать оптимальное. Эти системы могут быть модельными или экспертными.

*Модельные информационные системы* предоставляют пользователю математические, статистические, финансовые и другие модели, использование которых облегчает выработку и оценку альтернатив решения. Пользователь может получить недостающую ему для принятия решения информацию путем установления диалога с моделью в процессе ее исследования.

Основными функциями модельной информационной системы являются:

— работа в среде типовых математических моделей, включая решение основных задач моделирования типа «как сделать, чтобы?», «что будет, если?», анализ чувствительности и др.;

— достаточно быстрая и адекватная интерпретация результатов моделирования;

— оперативная подготовка и корректировка входных параметров и ограничений модели;

— графическое отображение динамики модели;

— объяснение пользователю хода работы модели.

*Экспертные информационные системы* обеспечивают выработку и оценку возможных альтернатив пользователем за счет создания экспертных систем, связанных с обработкой знаний.

Экспертная поддержка принимаемых пользователем решений реализуется на двух уровнях.

Работа первого уровня экспертной поддержки исходит из концепции «типовых управленческих решений», в соответствии с которой часто возникающие в процессе управления проблемные ситуации можно свести к некоторым однородным классам управленческих решений, т.е. к некоторому типовому набору альтернатив. Для реализации экспертной поддержки на этом уровне создается информационный фонд хранения и анализа типовых альтернатив.

Если возникающая проблемная ситуация не ассоциируется с имеющимися классами типовых альтернатив, в работу должен вступать второй уровень экспертной поддержки управленческих решений. Этот уровень генерирует альтернативы на базе имеющихся в информационном фонде данных, правил преобразования и процедур оценки синтезированных альтернатив.

Общая цель автоматизации управления — повышение эффективности деятельности объекта управления, что обеспечивается следующими направлениями автоматизации.

1. Повышение оперативности управления — сокращение времени выполнения таких операций, как сбор, поиск, предварительная обработка и передача информации, расчеты, решение логических задач, оформление документов и т.д.

2. Снижение трудозатрат ЛПР на выполнение вспомогательных процессов. К ним относятся информационные и расчетные операции, которые носят вспомогательный характер и при этом являются весьма трудоемкими. В результате высвобождения от рутинной работы должностные лица могут сосредоточиться на творческих процессах управления.

3. Повышение объективности и обоснованности принимаемых решений. Процесс принятия решений строится на основе анализа и прогноза развития ситуации с применением математического аппарата. При этом сохраняют свое значение традиционные

методы обоснования решений, опирающиеся на опыт и интуицию.

Следует отметить, что оптимальных решений не всегда удается достичь даже в условиях автоматизированного управления, поэтому скорее всего более верно говорить о рациональных решениях.

Итак, информационная технология управления должна решать текущие задачи стратегического и тактического планирования, бухгалтерского учета и оперативного управления фирмой, т.е. управлять системой принятия решений, а именно:

- управлять материальными ресурсами;
- управлять знаниями и информационной средой;
- автоматизировать процессы управления;
- обеспечивать создание интегрированных баз данных и баз знаний;
- обеспечивать сокращение персонала и повышение производительности труда;
- обеспечивать интеграцию всех звеньев управления;
- автоматизировать обработку данных и обеспечить соответствующий мониторинг основных показателей экономического состояния предприятия (организации);
- создавать новую информационную культуру деятельности;
- усиливать интеллектуальные возможности ЛПР.

## **5.2. Локальные и корпоративные информационные системы**

Информационную систему управления (ИСУ) можно охарактеризовать как систему процессов, использующую комплексный набор взаимодействующих элементов и их связей для сбора, обработки, хранения и предоставления информации для принятия решений и достижения установленных целей. Сейчас уже невозможно найти предприятие, на котором в той или иной степени не применялись бы автоматизированные информационные системы управления. Информационная система является частью организации. При этом эффективность работы всей организации часто находится в прямой зависимости от эффективности функционирования ее ИСУ.

*Локальные информационные системы управления* — простейшие информационные системы, реализующие отдельные функции управления предприятием (бухгалтерский учет, сбыт, снабжение, учет готовой продукции, складское хозяйство, управление автотранспортом, учет кадров, маркетинг и т.д.).

В настоящее время автоматизация отдельной функции, например бухгалтерского учета или сбыта готовой продукции, считается уже пройденным этапом для многих предприятий. Предприятие, ограничивающееся при автоматизации управления своей деятельностью лишь локальными информационными системами, не может быть конкурентоспособным в современных условиях. Опыт показывает, что полнофункциональная информационная система обязательно должна быть интегрированной или корпоративной.

*Интегрированные информационные системы* предназначены для автоматизации всех функций управления экономическим объектом (фирмой, предприятием, организацией). При этом фирма или корпорация может иметь территориальную разобщенность между подразделениями, филиалами, отделениями, офисами и т. д. Напомним, что корпорация называется сложным хозяйствующим субъектом, имеющим иерархическую структуру и включающий в себя предприятия самого различного масштаба и профиля деятельности.

Интегрированные информационные системы — сравнительно новый класс систем, поэтому применяемая в этой сфере терминология пока еще не устоялась. Например, некоторыми авторами термины «интегрированная» и «корпоративная» используются фактически в качестве синонимов, другие определяют корпоративные информационные системы как системы, предназначенные для территориально разобщенных предприятий (корпораций). Но это вопрос только терминологии, и для нас не является принципиальным.

*Интегрированные корпоративные информационные системы (ИКИС)* — системы, используемые для автоматизации всех функций предприятия и охватывающие весь цикл работ от планирования деятельности до сбыта продукции. Они ориентированы на автоматизацию процессов управления на всех уровнях и обладают средствами поддержки корпоративного управления. ИКИС включают в себя ряд модулей (подсистем), работающих в едином информационном пространстве и выполняющих функции поддержки соответствующих направлений деятельности предприятия (планирование, анализ и финансовый учет, производство, маркетинг, учет человеческих ресурсов и т. д.). Анализ современного состояния рынка информационных систем показывает устойчивую тенденцию роста спроса именно на интегрированные системы управления хозяйственной деятельностью.

К интегрированным корпоративным системам можно отнести системы очень большого диапазона — от информационных систем среднего предприятия, которое имеет цеха, склады, магазины и другие подразделения с той или иной степенью самостоятельности,

находящиеся в пределах одного города, до информационных систем транснациональных корпораций.

Средние ИКИС имеют такие средства поддержки корпоративного управления, как возможность ведения совместной базы данных корпорации и получения консолидированной отчетности по любому виду хозяйственной деятельности.

Крупные ИКИС помимо перечисленных средств обладают возможностями более глубокой поддержкой процессов управления многофункциональными группами предприятий. В первую очередь здесь следует отметить средства корпоративного анализа с использованием технологии интерактивной аналитической обработки данных (OLAP — On-line Analytical Processing).

Интегрированная информационная система строится на единой программно-аппаратной платформе и общей базе данных. Отдельные функциональные подсистемы в ней взаимодействуют на основе единого технологического процесса обработки информации, для чего необходимо организовать на предприятии вычислительную сеть. Способ построения сети зависит от требований управленческого аппарата к оперативности информационного обмена и управления всеми структурными подразделениями фирмы. Повышение запросов к оперативности информации в управлении экономическим объектом привело к созданию сетевых технологий, которые развиваются в соответствии с требованиями современных условий функционирования организации. Это влечет за собой организацию не только локальных вычислительных систем, но и многоуровневых (иерархических) и распределенных информационных технологий в информационных системах организационного управления. Все они ориентированы на технологическое взаимодействие, которое организуется за счет средств передачи, обработки, накопления, хранения и защиты информации. Процессы обработки информации в интегрированных информационных системах базируются, как правило, на технологии клиент-сервер.

### **5.3. Классификация информационных систем по уровню управления**

Основными функциями управленческого аппарата различных организаций являются анализ ситуаций в компании и внешней среде и принятие решений по стратегическому и краткосрочному планированию ее деятельности.

На рис. 5.3 отображены уровни управления, которые соотнесены с такими факторами, как степень возрастания ответствен-



Рис. 5.3. Иерархия уровни управления

ности и сложности решаемых задач, а также динамика принятия решений по реализации задач.

Каждый из этих уровней требует определенной информационной поддержки, реализуемой на базе информационной технологии. В соответствии с уровнями принятия решений (управления) в функционировании информационной технологии можно выделить три уровня:

- оперативный (быстрого регулирования деятельности организации);
- стратегический (долгосрочное планирование);
- тактический (среднесрочное планирование).

Каждому уровню соответствуют свои информационные системы управления.

### 5.3.1. Системы оперативного уровня

Оперативный (операционный) уровень принятия решений является основой всех автоматизированных информационных систем. Он обеспечивает решение многократно повторяющихся задач и операций и быстрое реагирование на изменения входной текущей информации. Задачи, цели и источники информации на оперативном уровне заранее определены и структурированы. Программная обработка информации выполняется по заранее разработанным алгоритмам.

Системы оперативного уровня называют еще системами обработки данных/транзакций (в зарубежной литературе использу-

ется термин OLTP — On-line Transaction Processing — технологии, ориентированные на оперативную (транзакционную) обработку данных). Задачи, решаемые системами обработки данных, носят регулярный характер, а их временные рамки, как правило, не превышают нескольких дней. На этом уровне достаточно велики как объем выполняемых операций, так и динамика принятия управленческих решений (см. рис 5.3). Этот уровень управления и называют оперативным из-за необходимости быстрого реагирования на изменение ситуации. На уровне оперативного управления выполняется огромное количество текущих рутинных операций по решению различных функциональных задач предприятия.

Информационные технологии обеспечивают менеджеров оперативного уровня ежедневной информацией о состоянии дел как внутри фирмы, так и во внешней среде, необходимой для принятия оперативных решений. Через оперативный уровень предоставляются данные для верхних уровней управления о результатах деятельности исполнителей. Инструментальные средства на оперативном уровне управления имеют небольшие аналитические возможности. Их достаточно для периодического формирования строго структурированных сводных типовых отчетов, удовлетворения запросов менеджеров о текущем состоянии фирмы и контроля информационных потоков организации (например, представление сведений о текущей дебиторской задолженности, о состоянии склада, о денежных остатках в кассе, о состоянии расчетного счета и т.д.). Информационная технология, поддерживающая управление на оперативном уровне, является также связующим звеном между организацией и внешней средой (рис. 5.4).

Оперативное управление ориентировано на достижение целей, сформулированных на стратегическом уровне, и использует потенциал, определенный на тактическом уровне. Главная функция оперативных систем — регистрация в базе данных и обработка элементарных событий, сопутствующих бизнес-процессам. Напомним, что бизнес-процесс — это совокупность взаимосвязанных действий или операций, которые используют ресурсы предприятия для создания или получения продукции (работ, услуг), необходимой потребителю. Бизнес-процессами являются прием и увольнение сотрудников, приход и расход материальных ценностей на складах и производственных подразделениях, оплата материальных ценностей и оказанных предприятию услуг через банк, ведение табеля учета рабочего времени и т.д.

Задачи системы оперативной обработки данных решаются непосредственными исполнителями бизнес-процессов (кладовщиками, рабочими, кассирами, бухгалтерами, администраторами



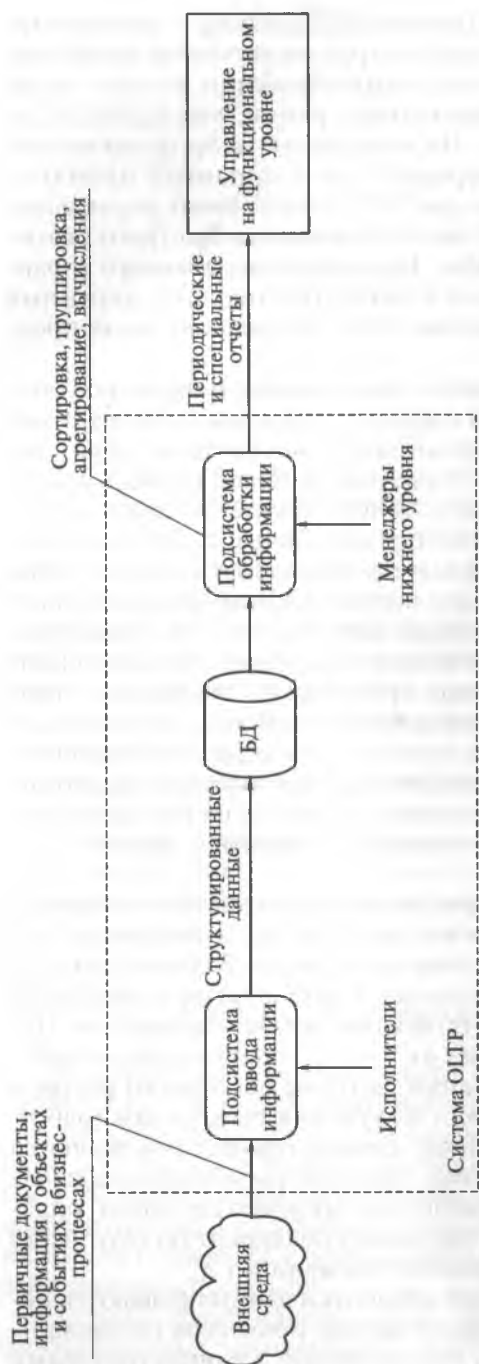


Рис. 5.4. Система оперативного уровня

торговых залов) параллельно с выполнением основных обязанностей. От них требуется элементарная компьютерная грамотность, в первую очередь умение пользоваться рабочим местом локальной вычислительной сети предприятия и оконным интерфейсом для заполнения экранных форм. Основная задача, стоящая перед информационной системой оперативного уровня, — обеспечить высокую скорость прохождения информационных потоков, связывающих участников бизнес-процессов. При необходимости системы обработки данных должны быть синхронизированы с информационными системами управления технологическими процессами на предприятии.

Системы оперативного уровня позволяют сделать контроль за исполнением бизнес-процессов прозрачным, поскольку в каждый момент времени можно оперативно выяснить состояние предприятия на всех рабочих местах.

К числу важнейших приоритетов оперативного управления следует отнести следующие:

- получение прибыли за счет реализации запланированных заранее мероприятий с использованием накопленного потенциала;

- регистрацию, накопление и анализ отклонений хода производства от запланированного;

- выработку и реализацию решений по устранению или минимизации нежелательных отклонений.

На основе накопленной базы оперативных данных решаются задачи более высоких уровней управления — тактического и стратегического.

### **5.3.2. Системы тактического (функционального) уровня**

Информационные системы управления функционального (тактического) уровня служат для решения задач управления бизнес-процессами предприятия. К тактическому уровню относятся процедуры среднесрочного планирования, анализа и организации работ. Если, например, на оперативном уровне мы имеем дело с отдельным заказом и сопутствующими его выполнению транзакциями, то на тактическом уровне рассматривается уже свод заказов для формирования производственной программы, либо делается прогноз прибыли и разработка плана выпуска продукции на ближайший период (неделю, месяц, квартал) на основании анализа статистических данных по спросу на продукцию и т. д. Результаты решения подобных задач предназначены для менеджеров среднего звена — начальников производственных и транс-

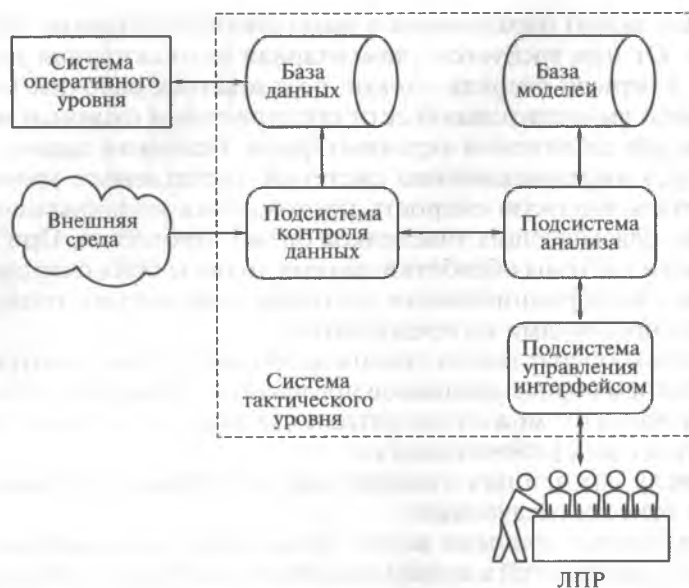


Рис. 5.5. Система функционального уровня

портных цехов, руководителей служб снабжения и маркетинга, планово-финансовых отделов и т. д.

Функциональный уровень управления обеспечивает решение задач, требующих предварительного анализа информации, подготовленной на первом уровне. Объем решаемых задач уменьшается, но возрастает их сложность (см. рис. 5.3). При этом не всегда удастся выработать нужное решение оперативно, требуется дополнительное время на осмысление, сбор недостающих сведений и т. п. Управление связано с некоторой задержкой от момента поступления информации до принятия решений и их реализации, а также от момента реализации решений до получения реакции на них. Результаты принимаемых управленческих решений проявляются спустя некоторое время (месяц, квартал, год).

Основные функции систем тактического уровня управления:

- мониторинг и контроль деятельности исполнителей бизнес-процессов;
- администрирование;
- сравнение текущих показателей с прошлыми;
- составление аналитических отчетов за определенный период;
- планирование и прогнозирование;
- обеспечение доступа к архивной информации;
- анализ отклонений;

– принятие тактических управленческих решений и т.д.

Для поддержки принятия тактического решения в информационной технологии фирмы используют такие инструментальные средства, как базы данных, базы моделей, системы обработки знаний, системы поддержки принятия решений и т.д. (рис. 5.5). Так, с помощью использования моделей можно анализировать продажи, при этом учитывая в значительной степени внешние требования поставщиков и потребителей. Модели, базируясь на математической интерпретации проблемы, при помощи определенных алгоритмов способствуют нахождению информации, полезной для принятия наиболее правильных решений.

### 5.3.3. Системы стратегического уровня

Основным инструментарием для поддержки работы высшего руководящего звена являются информационные системы для реализации стратегических перспективных планов развития организации. Системы стратегического уровня обеспечивают выработку управленческих решений, направленных на достижение долгосрочных целей организации. На стратегическом уровне рассматриваются вопросы выпуска и продвижения на рынок новой продукции, поиска новых рынков сбыта, выбора источников финансирования, привлечения инвесторов, реинжиниринга бизнес-процессов. Часто стратегический уровень управления называют долгосрочным планированием. Правомерность принятого на этом уровне решения может быть подтверждена спустя достаточно длительное время — могут пройти месяцы или даже годы. Ответственность за принятие управленческих решений чрезвычайно велика и определяется не только результатами анализа с использованием математического аппарата, но и профессиональной подготовкой и интуицией менеджеров верхнего уровня. Итак, стратегический уровень ориентирован на руководителей высшего ранга. За счет организации информационной технологии обеспечивается доступ к информации, отражающей текущее состояние дел в фирме, внешней среде, их взаимосвязь.

Основные цели стратегического уровня управления:

- определение системы приоритетов развития организации;
- оценка перспективных направлений развития организации;
- выбор и оценка необходимых ресурсов для достижения поставленных целей.

В соответствии с этими направлениями информационная технология обеспечивает высшему руководству оперативный удобный доступ и сортировку информации по ключевым факторам, кото-

рые позволяют оценивать степень достижения стратегических целей фирмы и прогнозировать ее деятельность на длительную перспективу. Особенностью информационной технологии систем долгосрочного планирования и анализа является построение агрегированных моделей развития организации с учетом деятельности смежных производств.

При выработке стратегических управленческих решений учитываются следующие факторы:

- особенности развития рыночных отношений в стране;
- возможные перспективные виды продукции (товары и услуги), относящиеся к профилю деятельности организации или предприятия;
- потенциальные виды производственных ресурсов, возможные для использования при создании новых видов продукции (товаров, услуг);
- перспективные технологические процессы изготовления новых видов продукции (товаров и услуг).

Учет перечисленных факторов в модели функционирования информационной технологии базируется преимущественно на использовании внешней по отношению к деятельности организации информации. Таким образом, информационная технология должна располагать развитой коммуникационной средой (включая Internet) для получения, накопления и обработки внешней информации.

Отличительной особенностью функционирования информационной технологии в долгосрочном стратегическом планировании, базирующемся на использовании агрегированных моделей, следует считать решающую роль самого управленческого персонала в процессе принятия решений. Высокий уровень неопределенности и неполноты информации повышает значение субъективного фактора как основы принятия решений. При этом автоматизированная информационная технология выступает в роли вспомогательного средства, обеспечивающего главную предпосылку для организации деятельности аппарата управления.

Таким образом, информационные технологии поддержки стратегического уровня принятия решений помогают высшему звену управления организацией решать неструктурированные задачи, основной из которых является сравнение изменений, происходящих во внешней среде, с имеющимся потенциалом фирмы.

Можно выделить следующие подходы к использованию систем стратегического планирования:

- первый заключается в том, что сначала формулируются цели и стратегии их достижения, а только затем автоматизированная информационная технология приспособливается к выработанной заранее стратегии;

— второй основывается на том, что организация использует стратегическую информационную систему при формулировании целей и стратегическом планировании;

— третий характеризуется применением методологии синтеза двух предыдущих подходов.

В российских условиях решение стоящих перед ЛПР задач стратегического управления осложняется постоянными изменениями условий, в которых работают российские предприятия (в первую очередь следует отметить частые поправки, принимаемые к налоговому законодательству), да и для всей мировой экономики последние годы характеризуются появлением многочисленных факторов нестабильности. В связи с этим в настоящее время в применяемых на российских предприятиях интегрированных корпоративных системах, как правило, реализована автоматизация оперативного и тактического уровней управления. Что же касается задач стратегического уровня управления, то они в большинстве случаев решаются на основе имеющихся данных нижнего уровня, а также благодаря профессиональному опыту и интуиции топ-менеджеров предприятий.

#### **5.4. Системы поддержки принятия решений**

Ранее были рассмотрены концепции информационных систем принятия решений на различных уровнях управления. Системы поддержки принятия решений (СППР) представляют собой системы, максимально приспособленные к решению задач повседневной управленческой деятельности. С помощью СППР можно решать некоторые неструктурированные и слабоструктурированные задачи, в том числе и многокритериальные. Современные системы поддержки принятия решения возникли в результате слияния информационных систем и систем управления базами данных для оптимального решения задач управления бизнес-процессами. С момента появления первых разработок по созданию СППР им не было дано четкого определения. И в настоящее время нет общепринятого определения СППР, поскольку конструкция СППР существенно зависит от вида задач, от доступности данных, информации и знаний, а также от пользователей системы. Необходимо подчеркнуть, что СППР поддерживают, а не заменяют выработку решений менеджерами.

Основные характеристики систем поддержки принятия решений следующие:

— возможность решения проблем, развитие которых трудно прогнозировать;

- наличие инструментальных средств моделирования и анализа;
- возможность легко менять постановки решаемых задач и входных данных;
- гибкость и адаптируемость к изменению условий;
- использование технологии, максимально ориентированной на пользователя.

Информационная технология поддержки принятия решений применяется на разных уровнях управления.

Любой процесс принятия решений включает в себя:

- определение проблемы и критериев принятия решения;
- выбор методики принятия решений;
- выявление и анализ альтернативных решений;
- выбор решения;
- информирование о результатах анализа.

Системы поддержки принятия решений обслуживают частично структурированные задачи, результаты которых трудно спрогнозировать заранее. СППР имеют достаточно мощный аналитический аппарат с множеством моделей. База моделей состоит из стратегических, тактических, оперативных и математических моделей в виде совокупности модельных блоков, модулей и процедур, используемых как элементы для их построения.

*Стратегические модели* (детерминистские, описательные, специализированные для использования на одной определенной фирме) используют на высших уровнях управления для установления целей организации, объемов ресурсов, необходимых для их достижения, а также политики приобретения и использования этих ресурсов. Они могут быть также полезны при выборе вариантов размещения предприятий, прогнозировании политики конкурентов и т. п. Для стратегических моделей характерны значительная широта охвата, множество переменных, представление данных в сжатой агрегированной форме. Часто эти данные базируются на внешних источниках и могут иметь субъективный характер. Горизонт планирования в стратегических моделях, как правило, измеряется в годах.

*Тактические модели* (детерминистские, оптимизационные и универсальные, т. е. могут быть использованы в различных организациях) применяются управляющими среднего уровня для распределения и контроля использования имеющихся ресурсов. Среди возможных сфер использования следует указать финансовое планирование, планирование требований к работникам, планирование увеличения продаж, построение схем компоновки предприятий. Эти модели применимы обычно лишь к отдельным частям фирмы (например, к системе производства и сбыта) и могут также включать в себя агрегированные показатели. Временной горизонт, охватываемый тактическими моделями — от одного

месяца до двух лет. Здесь также могут потребоваться данные из внешних источников, но основное внимание при реализации данных моделей должно быть уделено внутренним данным фирмы.

*Оперативные модели* (детерминистские, оптимизационные и универсальные) используют на низших уровнях управления для поддержки принятия оперативных решений с горизонтом, измеряемым днями и неделями. Возможные применения этих моделей включают в себя ведение дебиторских счетов и кредитных расчетов, календарное производственное планирование, управление запасами и т. д. В оперативных моделях для расчетов обычно используют внутрифирменные данные.

*Математические модели* состоят из совокупности модельных блоков, модулей и процедур, реализующих математические методы. Сюда могут входить процедуры линейного программирования, статистического анализа временных рядов, регрессионного анализа и т. п. — от простейших процедур до сложных пакетов прикладных программ. Модельные блоки, модули и процедуры могут быть использованы как поодиночке, так и комплексно для построения и поддержания моделей.

Система управления базой моделей должна обладать следующими возможностями:

- создавать новые модели или изменять существующие;
- поддерживать и обновлять параметры моделей;
- манипулировать моделями.

Использование моделей в составе информационных систем началось с применения статистических методов и методов финансового анализа, которые реализовывались командами обычных алгоритмических языков. Позже были созданы специальные языки, позволяющие моделировать ситуации типа «что будет, если?» или «как сделать, чтобы?». Такие языки, созданные специально для построения моделей, дают возможность построения моделей определенного типа, обеспечивающих нахождение решения при гибком изменении переменных.

Существует множество типов моделей и способов их классификации, например, по цели использования, области возможных приложений, способу оценки переменных и т. п.

По цели использования:

— оптимизационные модели, связанные с нахождением точек минимума или максимума некоторых показателей (например, управляющие часто хотят знать, какие их действия ведут к максимизации прибыли или минимизации затрат);

— описательные модели, описывающие поведение некоторой системы и не предназначенные для целей управления (оптимизации).

По способу оценки:



– детерминистские модели, использующие оценку переменных одним числом при конкретных значениях исходных данных;

– стохастические модели, оценивающие переменные несколькими параметрами, так как исходные данные заданы вероятностными характеристиками.

Детерминистские модели более популярны, чем стохастические, — они менее дорогие, их легче строить и использовать. К тому же часто с их помощью можно получить вполне достаточную информацию для принятия решения.

По области возможных приложений:

– специализированные модели, предназначенные для использования только одной системой;

– универсальные модели, предназначенные для использования различными системами.

Специализированные модели более дорогие, чем универсальные. Их обычно применяют для описания уникальных систем, и они обладают большей точностью.

Для задач СППР характерны трудности формализации целей и ограничений. Имеющаяся в распоряжении ЛПР информация часто бывает неполной, нечеткой и противоречивой, поэтому задачи планирования решают с помощью различных методов статистического анализа, экспертных систем, математического и имитационного моделирования. Возможно также применение методов интеллектуального анализа данных (искусственные нейронные сети, методы нечеткой логики, системы поиска логических правил в данных и т. д.).

СППР реализуются обычно в виде диалоговых человеко-машинных систем. Для них характерна высокая активность ЛПР, варьирующего в процессе решения задач своими целями и ограничениями. При создании СППР можно использовать Web-технологии.

## **5.5. Основные методологии создания экономических информационных управляющих систем**

### **5.5.1. Планирование потребности в материалах (Material Requirements Planning, MRP) и мощностях (Capacity Requirements Planning, CRP)**

MRP-системы — прямые предшественники современных ИКИС. Считается, что с появлением этой методологии можно

говорить о возникновении современных экономических корпоративных информационных систем. Первые мысли о возможности использования средств вычислительной техники для планирования деятельности предприятий (в том числе, планирования производственных процессов) возникли еще в начале 1960-х годов. Тогда была разработана методология планирования потребности в материалах — MRP. Суть концепции MRP состоит в том, чтобы минимизировать издержки, связанные с запасами на складах и на различных участках в производстве. Метод MRP позволяет обеспечивать компоненты, требуемые в соответствии с планом производства конечной продукции точно в том количестве и в те сроки, которые требуются для изготовления конечных изделий.

Собственно методология MRP — это реализация двух известных принципов JIT (Just In Time — «вовремя заказать») и KanBan («вовремя произвести»). JIT — это система управления материалами в производстве, при которой комплектующие с предыдущего этапа производства или от внешнего поставщика доставляются именно в тот момент, когда они требуются, но не раньше. Партии комплектующих настолько малы, насколько это возможно (это зависит от времени выполнения и стабильности цикла доставки). Наиболее успешно система JIT работает на предприятиях среднего масштаба с серийным типом производства, непрерывным потоком материалов и комплектующих, высокой скоростью производства стандартных изделий. В данной ситуации процедуры планирования и контроля в достаточной степени стандартизованы и просты. На крупных высокотехнологичных предприятиях, где процедура планирования и контроля производственных процессов является сложной, JIT практически не используют.

Концепция MRP легла в основу построения так называемых MRP-систем. MRP-система, получая на входе данные о наличии материалов на складе и их свойствах, «зная», что именно требуется для производства конечного продукта, а также имея возможность соотнести производственный цикл с временной шкалой, способна предоставить в руки менеджера информацию, которая позволит оптимально (относительно сроков закупки и производства) спланировать процесс производства.

MRP-программа отслеживает движение материалов, чтобы оптимизировать процесс выработки решений о заказе новых поставок, автоматизирует этот процесс, генерируя заказы автоматически (в ее ведении находится вся информация, требуемая для своевременного оформления новых требований). В результате создается глобальный план заказов поставщикам, в котором должно быть расписано, что, когда и у кого необходимо заказать.

Логику работы MRP-системы можно представить с помощью диаграммы (рис. 5.6). На диаграмме отображены основные информационные элементы MRP-системы.

*В описании состояния материалов* отражается максимально полная информация о всех материалах, необходимых для производства изделия, — остаток данного материала на складе, информация о том, зарезервирован ли он для других целей, присутствует ли он в заказах или заказ на него только планируется, регулярность поставок. Следует учитывать необходимость страхового запаса материала для поддержания процесса производства в случае возникновения непредвиденных задержек в его поставках.

*Программа производства* представляет собой оптимизированный график распределения времени для производства необходимой партии готовой продукции за планируемый период.

*Спецификация конечного продукта (изделия)* — это перечень материалов и их количество, необходимых для производства определенного объема готового продукта (изделия). Здесь же содержится полная информация о технологии сборки готового изделия.

На основе плана выпуска продукции, спецификации изделия и учета особенностей технологической цепочки и рассчитывают



Рис. 5.6. Входные элементы и результаты работы MRP-системы производства

потребности производства в материалах, привязанные к конкретным срокам.

Основными результатами работы MRP-системы являются план заказов и изменения к нему.

*План заказов* определяет, какой материал и в каком количестве должен быть заказан в рассматриваемый период времени в планируемые сроки. Генеральный план заказов является руководством для дальнейшей работы с поставщиками.

*Изменения к плану заказов* — модификация ранее запланированных заказов (отмена, изменение, перенесение на другой период).

Кроме того, MRP-система формирует ряд отчетов для проведения анализа текущего состояния дел и прогнозирования дальнейшей деятельности.

*Исполнительный отчет* информирует об ошибках в работе системы и возникших критических ситуациях при планировании (например, информация об отсутствии страхового запаса какого-либо отдельного материала).

*Отчет об узких местах планирования* предназначен для получения информации о сроках, которые требуют особого внимания и в которые могут возникнуть необходимость управленческих решений (непредвиденное запоздание заказов, избыток материалов на складах и т. д.).

*Прогноз* — информация, полученная в результате анализа производственного процесса, о возможном будущем изменении объемов и характеристик выпускаемой продукции.

Программные системы, реализованные на базе MRP-методологии, позволили оптимально регулировать поставки материалов для производства продукции, контролировать складские запасы и саму технологию производства. Кроме того, использование MRP-систем позволило уменьшить объем постоянных складских запасов.

В целях повышения эффективности планирования в конце 1970-х годов в MRP-системах была реализована идея воспроизведения замкнутого цикла (Closed Loop Material Requirement Planning). Идея заключалась в рассмотрении более широкого спектра при планировании путем введения дополнительных функций. Подразумевалось составление производственной программы и ее контроль на цеховом уровне. К базовым функциям планирования производственных мощностей и планирования потребностей в материалах были добавлены дополнительные функции (например, контроля соответствия количества произведенной продукции количеству использованных в процессе сборки комплектующих, составления регулярных отчетов о задержках заказов, об объемах и динамике продаж продукции, о

поставщиках и др.). Созданные в процессе работы модифицированной MRP-системы отчеты анализировались и учитывались на дальнейших этапах планирования, изменяя при необходимости программу производства и план заказов, обеспечивая тем самым гибкость планирования по отношению к таким внешним факторам, как уровень спроса, текущее состояние дел у поставщиков комплектующих и др.

MRP-системы фактически просто формировали на основе утвержденной производственной программы план заказов на определенный период. Это не могло удовлетворить потребности усложняющегося производства. Постепенно становилось очевидным, что накопленный опыт управления материальными ресурсами может быть перенесен на решение задач, связанных с другими видами ресурсов — производственными. В результате появилась концепция CRP, в которой методы MRP перенесены на управление производственными мощностями.

Основные этапы реализации методологии CRP в экономических информационных системах можно представить в следующем виде.

1. Разрабатывается план распределения производственных мощностей для обработки каждого конкретного цикла производства в течение планируемого периода.

2. Устанавливается технологический план последовательности производственных процедур и в соответствии с пробной программой производства определяется степень загрузки каждой производственной единицы на срок планирования.

3. Если после цикла работы CRP-методологии программа производства признается реально осуществимой, то она становится основной для MRP-системы. В противном случае в нее вносятся изменения, и она подвергается повторному тестированию с помощью CRP-методологии.

### **5.5.2. Система MRP II**

После появления концепции MRP, казалось бы, все основные проблемы производства были решены, активно создавались и продавались компьютерные программы, реализующие ее несложные принципы. Однако в процессе дальнейшего анализа существующей ситуации в мировом бизнесе и ее развития выяснилось, что в концепции MRP есть серьезный недостаток. При расчете потребности в материалах не учитываются производственные мощности, их загрузка, стоимость рабочей силы и т.д. В 1980-х годах MRP-система с замкнутым циклом была трансформирована в систему планирования производственных ресурсов (manufac-

tory resource planning), которая получила название MRP II (вследствие идентичности аббревиатур).

MRP II позволяет управлять всем производственным процессом, а не только отдельными его фрагментами (рис. 5.7). Эта си-

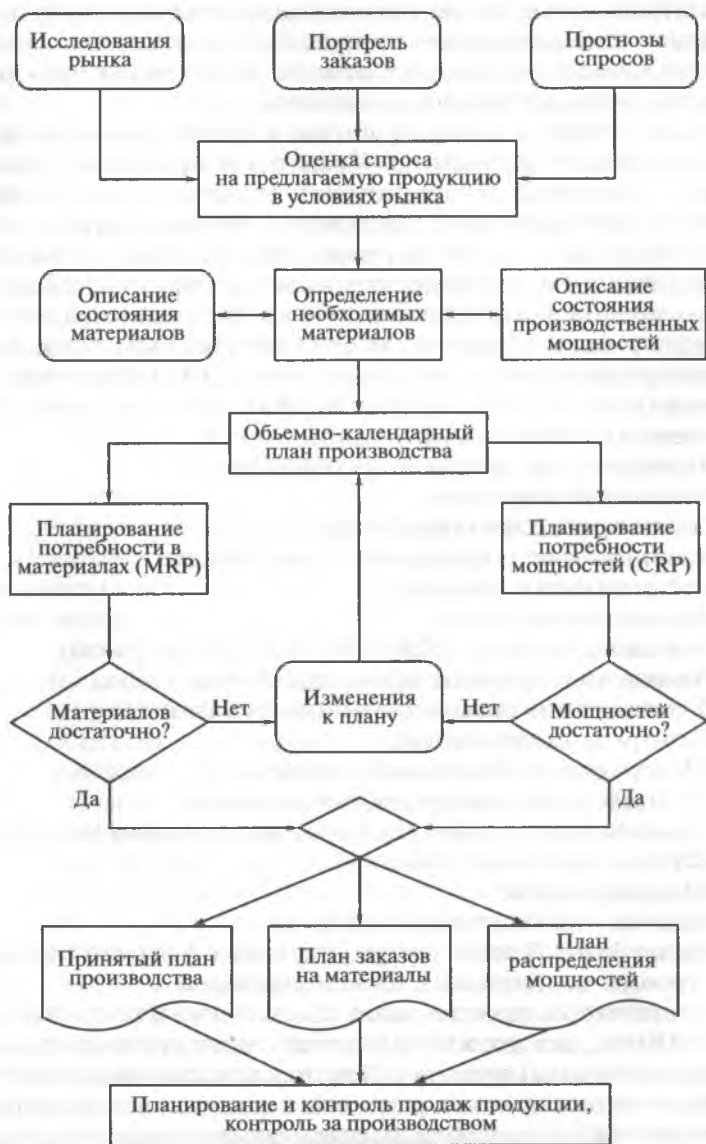


Рис. 5.7. Схема функционирования методологии MRP II

стема была создана для эффективного планирования всех ресурсов производственного предприятия, в том числе финансовых и кадровых. К базовым функциям планирования производственных мощностей и планирования потребностей в материалах были добавлены ряд дополнительных:

- контроль соответствия количества произведенной продукции количеству использованных в процессе сборки комплектующих;
- составление регулярных отчетов о задержках заказов, об объемах и динамике продаж продукции.

Стандарт MRP II был разработан в США и поддерживается Американским обществом по управлению производством и запасами — American Production and Inventory Control Society (APICS). В свое время APICS регулярно издавало документ «MRP II Standart System» в котором приводили основные требования к информационным производственным системам (последнее издание этого документа вышло в 1989 г.). В соответствии с «MRP II Standart System» в системе MRP II должны быть реализованы 16 основных групп функций (для того чтобы ПО управления предприятием было отнесено к классу MRP II, оно должно выполнять определенный объем этих функций).

1. Планирование продаж и производства.
2. Управление спросом.
3. Составление плана производства.
4. Планирование потребностей в материалах.
5. Спецификация продуктов.
6. Управление складом.
7. Плановые поставки (Scheduled Receipts Subsystem).
8. Управление на уровне производственного цеха.
9. Планирование производственных мощностей.
10. Контроль входа/выхода.
11. Материально-техническое снабжение.
12. Планирование распределения ресурсов.
13. Планирование и контроль производственных операций.
14. Финансовое планирование.
15. Моделирование.
16. Оценка результатов деятельности.

Стандарт MRP II делит сферы отдельных функций (процедур) на два уровня: необходимый и опциональный.

Для того чтобы программное обеспечение было отнесено к классу MRP II, оно должно выполнять определенный объем необходимых основных функций. Некоторые поставщики программного обеспечения приняли различные опции этого стандарта.

Система MRP II способна адаптироваться к внешней ситуации. Как видно из рис. 5.7, для MRP II характерны обратные связи, поскольку созданные в процессе ее работы отчеты учитываются

на дальнейших этапах планирования. В случае необходимости может быть изменена программа производства. Эти дополнительные функции обеспечивают гибкость планирования по отношению к внешним факторам — уровню спроса, надежности поставок материалов и комплектующих и т. д.

Результаты работы каждого модуля анализируются всей системой в целом, что, собственно, и обеспечивает ее гибкость по отношению к изменению внешних факторов. Это особенно важно для многочисленных предприятий, производящих продукцию с коротким жизненным циклом, требующую частых доработок. Лучшие образцы систем MRP II представляют собой автоматизированные программные комплексы, позволяющие оптимизировать объемы выпуска и потребительские характеристики выпускаемой продукции на основе анализа ситуации на рынке. Системы MRP II ориентированы на производство, причем в большинстве случаев именно на промышленное, и призваны интегрировать все основные процессы, реализуемые предприятием, — такие как снабжение, запасы, производство, продажа, планирование, контроль за выполнением плана, затраты, финансы и т. д.

Задача информационных систем класса MRP II — оптимальное формирование потоков материалов (сырья), полуфабрикатов (в том числе находящихся в производстве) и готовых изделий.

В результате применения MRP II должны быть реализованы:

- оперативное получение информации о текущих результатах деятельности предприятия как в целом, так и с полной детализацией по отдельным заказам, видам ресурсов, выполнению планов;

- долгосрочное, оперативное и детальное планирование деятельности предприятия с возможностью корректировки плановых данных на основе оперативной информации;

- оптимизация производственных и материальных потоков со значительным сокращением непроизводственных затрат и реальным сокращением материальных ресурсов на складах;

- планирование и контроль за всем циклом производства с возможностью влияния на него в целях достижения максимальной эффективности в использовании производственных мощностей, всех видов ресурсов и удовлетворения потребностей заказчиков;

- сокращение непроизводственных затрат;

- отражение финансовой деятельности предприятия в целом.

В соответствии со стандартом MRP II система должна на любой момент времени показать потребность в материальных ресурсах для выполнения плана производства, а также адекватно реагировать на замену материалов в спецификации в соответствии с изменениями требований к технологическому процессу и к процессу снабжения.



При тестировании программы необходимо попытаться ввести варианты использования различных материалов, а также их прогнозных комбинаций для получения полноценного MRP-отчета.

Проверка соответствия стандарту MRP II должна подтвердить наличие ряда свойств системы:

- предоставление структур данных и запросов, позволяющих построить план снабжения предприятия в соответствии с реальными возможностями поставщиков, включая ограничения по срокам исполнения заказов, минимальные партии отгрузки, условия транспортировки сырья и материалов;

- возможность отслеживания полного цикла исполнения заказа клиента (план его обеспечения ресурсами на складе, план производства и закупок, фактический процесс выполнения этого плана на всех стадиях цикла материально-технического снабжения и производства);

- резервирование ресурсов за соответствующими заказами клиентов;

- выдача информации по произвольно взятому заказу клиента относительно «состояния позиций на складе», плана производства и плана закупок, а также информации о ходе исполнения производственного плана в части обеспечения данного заказа клиента, о заказах поставщикам, их состоянии на всех стадиях жизненного цикла, включая процессы оплаты счетов;

- возможность планирования загрузки оборудования (мощностей) предприятия (CRP-отчет) в соответствии с планом производства и технологическими спецификациями. При этом система должна адекватно отражать режимы работы оборудования при использовании различного инструмента, спецоснастки и быть связана с подсистемой планирования потребностей в ресурсах (MRP);

- возможность расчета сроков выполнения заказов и заявок клиентов с учетом реальной загрузки производственных мощностей и имеющегося материально-технического обеспечения. Система должна позволять «экспериментирование» при замене собственных подразделений субподрядчиками, при увеличении сменности работы с учетом изменения себестоимости выпускаемой продукции (увеличение сменности приводит как к дополнительным выплатам за сверхурочную работу, так и к росту затрат на ремонт оборудования).

Итак, системы MRP II предназначены для эффективного планирования всех ресурсов предприятия. Основная суть концепции MRP II состоит в том, что прогнозирование, планирование и контроль производства осуществляются по всему жизненному циклу продукции, начиная от закупки сырья и заканчивая отгрузкой продукции потребителю. В системе MRP II интегрирова-

но большое число модулей, результаты работы которых анализируются системой в целом, что и обеспечивает ее гибкость по отношению к различным внешним факторам (как, например, текущему спросу на продукцию и др.).

### 5.5.3. ERP-системы

В последней четверти прошлого века производство «на склад» стало вытесняться производством «под заказ клиента». Важный удельный вес стали приобретать транспортные, телекоммуникационные, финансовые, учебные подразделения корпораций. Обладая определенной автономностью, эти подразделения работают в тесной связи. Возникают так называемые «виртуальные предприятия» — географически распределенные либо временные объединения предприятий, работающих над одним проектом и связанных между собой сложной цепочкой внутрикорпоративных материальных, финансовых, кадровых потоков. Попытка соответствующего усовершенствования систем MRP II привела к появлению ERP-систем.

Задача ERP-системы (Enterprise Resource Planning, ERP — Планирование ресурсов корпорации) — интегрировать все подразделения и функции корпорации в единой информационной системе. Все стороны производственной и коммерческой деятельности охватываются ERP: производство, планирование, управление договорами, материально-техническое снабжение, финансы, бухгалтерия, управление кадрами, сбыт, управление запасами. Таким образом, главная задача ERP сводится к распространению принципов MRP II на управление современными корпорациями.

Основа ERP — единая база данных, которой пользуются в равной степени бухгалтерия, производство, служба маркетинга, отдел кадров, склады. Введенная в эту базу данных информация мгновенно становится доступной самым различным подразделениям корпорации. Возникает инфраструктура электронного обмена данными как между подразделениями и предприятиями корпорации, так и между корпорацией и ее поставщиками и потребителями.

Существует несколько определений ERP системы:

— класс интегрированных систем управления, представляющий собой унифицированную централизованную базу данных, единое приложение и общий пользовательский интерфейс для управления финансово-экономической деятельностью: производственной, экономической и финансовой, сбытовой, закупочной, хранения продукции и материалов и множества других;

— комплекс программных средств и управленческих решений, который позволяет в необходимые сроки с максимальной точностью выполнить заказ потребителя путем правильного планирования, перераспределения и направления всех ресурсов предприятия.

Системы класса ERP — это набор интегрированных приложений, позволяющих создать единую среду для автоматизации планирования, учета, контроля и анализа всех основных бизнес-операций предприятия. Концепция ERP до настоящего времени не стандартизирована. Зачастую при оценке возможностей той или иной конкретной информационной системы и решении вопроса об отнесении ее к классу развитых систем MRP II или к классу ERP-систем аналитики расходятся во мнениях.

Зарубежные авторы выделяют следующие отличительные особенности ERP-систем:

— поддержка различных типов производств (сборочного, обрабатывающего и др.) и видов деятельности предприятий и организаций (например, ERP-системы могут быть установлены не только на промышленных предприятиях, но и в организациях сферы услуг — банках, страховых и торговых компаниях и др.);

— поддержка планирования ресурсов по различным направлениям деятельности предприятия (а не только производства продукции);

— ERP-системы ориентированы на управление «виртуальным предприятием», отражающим взаимодействие производства, поставщиков, партнеров и потребителей в рамках интегрированной информационной системы;

— в ERP-системах больше внимания уделено финансовым подсистемам;

— добавлены механизмы управления транснациональными корпорациями, включая поддержку нескольких часовых поясов, языков, валют, систем бухгалтерского учета и отчетности;

— повышенные требования к инфраструктуре (Интернет/Инtranет), масштабируемости (до нескольких тысяч пользователей), гибкости, надежности и производительности программного обеспечения и различных платформ;

— повышены требования к интегрируемости ERP-систем с приложениями, уже используемыми предприятием (CAD/CAM/CAE/PDM-системами, АСУ ТП, системами управления документооборотом и др.), а также с новыми приложениями (например, электронного бизнеса). При этом именно на базе ERP-системы осуществляется интеграция всех приложений, используемых на предприятии;

— больше внимания уделено программным средствам поддержки принятия решений и средствам интеграции с хранилищами

данных (иногда включаемых в ERP-систему в виде нового модуля);

— в ряде ERP-систем разработаны развитые средства настройки (конфигурирования), интеграции с другими приложениями и адаптации (в том числе, применяемые динамически в процессе эксплуатации систем).

Эксперты определяют ERP как систему управления всеми бизнес-процессами предприятия, увязывающую функции отдельных подразделений с движением финансовых и товарных потоков по всей технологической цепочке управленческих процедур.

Преимущества ERP покажем на примере процесса выполнения заказа. В случаях отсутствия автоматизации либо частичной («локутной») автоматизации принятый заказ клиента начинается длительное, в основном «бумажное», путешествие по предприятию. Заказ часто заново вводится в различные компьютерные программы в разных подразделениях. В одном подразделении вводятся параметры заказа в виде необходимых материалов и комплектующих, в других — в виде производственного плана. При этом информационные системы соответствующих подразделений имеют между собой только «бумажную» связь. В результате зачастую никто на предприятии не знает, на каком этапе находится процесс выполнения заказа, готов ли заказ и не пора ли выставить счет на оплату.

На предприятии, где внедрена ERP-система, отсутствует проблема «информационной стыковки» различных подразделений. Для того чтобы определить состояние выполнения заказа, достаточно войти в систему и набрать номер заказа. В данном случае номер заказа является тем информационным полем, которое «просвечивает» все службы предприятия, связывая воедино их деятельность.

Внедрение ERP-системы повышает требования к оперативности работы с информацией и взаимодействию между подразделениями. Например, если работник склада (кладовщик) по нескольку дней не актуализирует информацию о поступивших на склад материалах и комплектующих, то отдел сбыта считает, что необходимого для выполнения заказа количества на складе нет. В результате выполнение заказа задерживается. По мнению специалистов российского представительства SAP AG, внедрение ИКИС класса ERP, как правило, меняет управленческую стратегию на предприятии. Они отмечают, что «... наши клиенты покупают не «систему». Они покупают порядок». Здесь имеется в виду, что, как правило, внедрение на предприятии ИКИС сопровождается реинжинирингом бизнес-процессов, т. е. их переосмыслением и перепроектированием (вплоть до изменения управленческой структуры) для достижения кардинальных улучшений в таких

целевых показателях бизнеса, как затраты, прибыль, оперативность.

Проверка соответствия системы стандарту ERP включает несколько этапов. При тестировании следует проверить наличие связи между модулем оперативного планирования производства и модулем управления персоналом. Например, в случае болезни какого-либо сотрудника должна быть обеспечена автоматическая либо полуавтоматическая корректировка сменных заданий.

Система должна увязывать все виды затрат ресурсов с бюджетом предприятия. Это должно проводиться в системе на стадии возникновения обязательств, а не в момент фактического осуществления платежей. Система должна содержать встроенный механизм связи плана закупок с бюджетом. Тестирование состоит в получении информации о состоянии бюджетных статей на различных стадиях возникновения обязательств (обязательств на стадии договоров, заказов и выставленных счетов-фактур), а также в проведении тестов на возможность формирования плана снабжения и заказов поставщикам при отсутствии достаточных средств на соответствующий период. Система должна предоставлять информацию о фактических затратах на производство отдельных видов продукции и затратах на содержание подразделений в разрезе статей, режимов работы, факторов отклонений и центров ответственности, т.е. структурной единицы предприятия, ответственной за определенные доходы либо расходы. В качестве теста системе может быть предложен такой запрос: «В какой мере повлияли на себестоимость продукции отклонения от планового уровня загрузки оборудования, относящегося к конкретному центру ответственности по конкретному подразделению?»

#### **5.5.4. BPM (Business Performance Management) — автоматизация процессов управленческого планирования и контроля**

В настоящее время остро встает необходимость в специализированных приложениях для автоматизации управленческих задач бюджетирования, финансового планирования, анализа и контроля. Международная компания IDC, специализирующаяся на независимом мониторинге рынка программного обеспечения, объединила такие приложения в новое семейство — BPM (Business Performance Management — управление эффективностью бизнеса). BPM-системы позволяют связывать воедино такие понятия, как миссия компании, стратегия развития, цели, долгосрочные планы, среднесрочные перспективы и конкретные бюджеты на ближайший период. Топ-менеджеры могут предоставлять плановые бюд-

жеты начальникам отделов, которые, исходя из оценки ситуации (могут ли они выполнить эти задачи, какие ресурсы им для этого нужны), их корректируют. Система позволяет им видеть и использовать в своей работе отчетность смежных подразделений: на основе планов поставок сырья оценивать свои возможности по объемам производства и т. п. Далее откорректированные и дополненные на нижнем уровне цифры агрегируются вновь до общекорпоративного уровня. Весь этот процесс «двунаправленного» бюджетирования итеративно повторяется до тех пор, пока не будет составлен наиболее «реальный» бюджет.

Благодаря единой среде сотрудничества каждый работник начинает более четко осознавать свою роль в процессе управления организацией. Достоверность бюджета повышается за счет вовлечения рядовых исполнителей в процесс его составления.

Аналитическая функциональность ВРМ-приложений обеспечивает возможность составления отчетности «на лету». Система ВРМ обладает возможностью своевременного обнаружения отклонения фактических показателей от их плановых величин и оповещения об этом менеджеров. В настоящее время ВРМ-приложения набирают все большую популярность. Предприятия различных сфер деятельности заинтересованы в использовании ВРМ-приложений как мощного механизма консолидации отчетности.

## **5.6. Аналитическая обработка данных для поддержки принятия решений**

Можно выделить три основные технологии поддержки принятия управленческих решений на основе накопленной информации:

- технологии, ориентированные на оперативную (транзакционную) обработку данных и реализованные в большинстве транзакционных систем (OLTP). Сфера действия таких технологий — область детализированных данных. Классические реляционные СУБД нормально справляются с подобными задачами, поэтому в подробном их рассмотрении нет необходимости;

- технологии OLAP (On-line Analytical Processing — интерактивная аналитическая обработка данных), ориентированные на область агрегированных показателей;

- технологии интеллектуальной обработки данных, ориентированные на область закономерностей. Интеллектуальная обработка проводится методами интеллектуального анализа данных (ИАД, в западной литературе — Data Mining). С помощью этих

технологий решаются задачи поиска функциональных и логических закономерностей в накопленной информации.

Рассмотрим подробнее перечисленные технологии.

### **5.6.1. OLAP (On-line Analytical Processing) — оперативный анализ данных**

Механизм OLAP является на сегодня одним из популярных методов анализа данных. OLAP — это не отдельно взятый программный продукт, не язык программирования и даже не конкретная технология. Если постараться охватить OLAP во всех его проявлениях, то это совокупность концепций, принципов и требований, лежащих в основе программных продуктов, которые облегчают аналитикам доступ к данным. Для начала выясним, зачем аналитикам надо специально облегчать доступ к данным.

Дело в том, что аналитики — это особые потребители корпоративной информации. Их задача — найти закономерность в большом массиве данных. Поэтому аналитик не будет обращать внимания на отдельно взятый факт (например, продажа конкретному покупателю отдельного товара). Одиночные факты в базе данных могут заинтересовать, к примеру, бухгалтера или начальника отдела продаж, в компетенции которого находится сделка. Аналитику одной такой записи мало — ему могут понадобиться все сделки за месяцы или годы. При этом аналитик отбрасывает ненужные ему подробности вроде ИНН покупателя, его точного адреса и номера телефона.

Технологии OLAP основаны на понятии хранилище данных. Приведем определение, сформулированное «основателем» хранилищ данных Биллом Инмоном. OLAP — это предметно-ориентированное, привязанное ко времени и неизменяемое собрание данных для поддержки процесса принятия управляющих решений.

Данные в хранилище поступают из оперативных систем (OLTP) и из внешних источников — статистических отчетов, Интернета, прайс-листов и т. п. (рис. 5.8). Оперативные данные «очищаются», интегрируются и сохраняются в реляционном хранилище. При этом они уже доступны для анализа, их можно использовать с помощью различных средств построения отчетов. Затем данные (полностью или частично) подготавливаются для OLAP-анализа. Они могут быть загружены в специальную БД OLAP или оставлены в реляционном хранилище. Важнейшим элементом информационно-аналитической системы являются метаданные, т. е. информация о структуре, размещении и трансформации данных.



Рис. 5.8. Структура корпоративной информационно-аналитической системы

Благодаря метаданным обеспечивается эффективное взаимодействие различных компонентов хранилища.

Возникает вопрос — зачем создавать хранилища данных, содержащие заведомо избыточную информацию, если необходимые данные уже имеются в файлах оперативных систем? Еще несколько лет назад в качестве главных причин называли различие форматов хранящихся данных, их разрозненность, локализация в разных местах корпоративной сети. Действительно, раньше хранение всех данных на центральном сервере базы данных было редким явлением для крупного предприятия. Сейчас в связи с интенсивным внедрением интегрированных корпоративных информационных систем положение меняется. Но остаются принципиальными три причины необходимости создания хранилищ данных. Во-первых, сложные аналитические запросы к оперативным данным «забирают» ресурсы сервера и тормозят работу информационной системы, надолго блокируя таблицы данных. Во-вторых, оперативные данные малопригодны для непосредственного сложного анализа. В-третьих, системы OLTP предназначены для оперативной обработки данных, поэтому они не приспособлены для хранения информации за длительный период, в то время как для OLAP интересен многолетний и многоаспектный анализ объекта.

Итак, OLAP предоставляет «сырье» для анализа в простой, понятной структуре, используя удобные быстродействующие сред-



ства доступа, просмотра и анализа деловой информации. Традиционные отчеты, даже построенные на основе единого хранилища, лишены гибкости. Их нельзя «покрутить», «развернуть» или «свернуть», чтобы получить желаемое представление данных. OLAP является удобным инструментом для многомерного анализа данных, накопленных в хранилище.

В основе OLAP лежит наглядная модель данных, организуемая самим пользователем, в виде многомерных кубов (гиперкубов). Осями многомерной системы координат служат атрибуты анализируемого бизнес-процесса — измерения (Dimensions). Например, для продаж это могут быть вид товара, регион, контрагент. В качестве одного из измерений используется время. Данные, количественно характеризующие бизнес-процесс, называются мерами (Measures). Это могут быть объемы продаж в количественном или денежном выражении, товарные остатки на складе, издержки и т. п. Пользователь получает естественную, интуитивно понятную модель данных. Анализируя информацию, он может «разрезать» куб по разным направлениям. Это дает возможность получать сводные или детальные сведения о бизнес-процессе и осуществлять прочие манипуляции, которые придут в голову пользователю в процессе анализа.

В качестве мер в трехмерном кубе, изображенном на рис. 5.9, использованы суммы продаж, а в качестве измерений — время, товар и склад. Измерения представлены на определенных уровнях группировки: товары группируются по видам, а данные о времени совершения операций — по месяцам. Чуть позже мы рассмотрим уровни группировки (иерархии) подробнее.

На самом деле с точки зрения математики кубом такой массив будет далеко не всегда: у настоящего куба количество элементов во всех измерениях должно быть одинаковым, а у кубов OLAP такого ограничения нет. Тем не менее, несмотря на эти детали, термин «кубы OLAP» ввиду своей краткости и образности стал

		Февраль			
		Январь			
Товар	Количество	Склад 1	Склад 2	Склад 3	Склад 4
	Мебель		1 000	200	100
Аксессуары		500	50	25	50
Ткань		500	50	25	50

Рис. 5.9. Пример OLAP-куба

общепринятым. Куб OLAP совсем не обязательно должен быть трехмерным. Он может быть и двух-, и многомерным — в зависимости от решаемой задачи. Особо искусственным аналитикам может понадобиться порядка 20 измерений, и серьезные OLAP-продукты именно на такое количество и рассчитаны. Более простые настольные приложения поддерживают около 6 измерений.

Однако куб сам по себе для анализа не пригоден. Даже трехмерный куб сложно отобразить на экране компьютера так, чтобы были видны значения интересующих мер. Что уж говорить о кубах с количеством измерений больше трех? Для визуализации данных, хранящихся в кубе, применяют, как правило, привычные двумерные табличные представления со сложными иерархическими заголовками строк и столбцов. Поэтому перед употреблением из многомерного куба извлекают обычные двумерные таблицы. Эта операция образно называется «разрезанием» куба. Аналитик как бы берет и «разрезает» измерения куба по интересующим его меткам. Этим способом аналитик получает двумерный срез куба и с ним работает. Примерно так же лесорубы считают годовые кольца на спиле. Соответственно «неразрезанными», как правило, остаются только два измерения — по числу измерений таблицы.

Определяющие принципы OLAP были сформулированы в 1993 г. Э. Коддом — «изобретателем» реляционных БД. Позже его определение было переработано в так называемый тест FASMI, требующий, чтобы OLAP-приложение предоставляло возможности быстрого анализа разделяемой многомерной информации. Расшифруем аббревиатуру FASMI:

- Fast (быстрый) — анализ должен производиться одинаково быстро по всем аспектам информации. Приемлемое время отклика — 5 с или менее;

- Analysis (анализ) — должна быть возможность осуществлять основные типы числового и статистического анализа, предопределенного разработчиком приложения или произвольно определяемого пользователем;

- Shared (разделяемой) — множество пользователей должно иметь доступ к данным, при этом необходимо контролировать доступ к конфиденциальной информации;

- Multidimensional (многомерной) — это основная, наиболее существенная характеристика OLAP;

- Information (информация) — приложение должно иметь возможность обращаться к любой нужной информации, независимо от ее объема и места хранения.

Работа с OLAP-системами может быть построена на основе двух описанных ниже схем.

1. Для «легковесного» применения подойдут OLAP-средства, встроенные в настольные приложения. Такие средства, как пра-

вило, имеют множество ограничений: на количество измерений, на допустимые иерархии и т. д. К подобным средствам, например, относится модуль Pivot Table, позволяющий работать с кубами в Microsoft Excel. Pivot Table входит в Microsoft Office с незапамятных времен и до недавнего времени был единственным OLAP-продуктом в его составе. В этом случае данные извлекаются модулем-клиентом непосредственно из реляционной СУБД.

2. В «тяжелых» случаях применяют двухступенчатую схему «клиент-сервер». Сервер обеспечивает непосредственно извлечение информации из СУБД и все прочее, необходимое для создания кубов. Специализированное же приложение-клиент предназначено для удобного (а главное — эффективного) просмотра кубов и выявления тех самых аналитических закономерностей. В линейке продуктов Microsoft серверная часть представлена продуктами Microsoft Analysis Services, которые входят в MS SQL Server. Сравнительно недавно в состав MS Office включен OLAP-клиент под названием Microsoft Data Analyzer.

В России разработкой технологий OLAP занимаются несколько компаний. Наиболее известный программный продукт — аналитическая платформа Контур фирмы InterSoft Lab. Все большую известность приобретает модуль Галактика Zoom системы «Галактика». Отметим, что эти программные продукты получили международное признание и благодаря привлекательному сочетанию цена/качество заняли свою нишу на западном рынке OLAP-продуктов.

Наличие в ERP-системе «встроенной OLAP-аналитики» станет в ближайшие годы важным конкурентным преимуществом интегрированных корпоративных информационных систем.

**Архитектура OLAP-приложений.** Многомерность в OLAP-приложениях может быть разделена на три уровня, из которых два в обязательном порядке присутствуют во всех OLAP-средствах.

1. Многомерное представление данных — средства конечного пользователя, обеспечивающие многомерную визуализацию и манипулирование данными. Многомерная обработка — средство (язык) формулирования многомерных запросов (традиционный реляционный язык SQL здесь оказывается непригодным) и процессор, умеющий обработать и выполнить такой запрос.

2. Многомерное хранение — средства физической организации данных, обеспечивающие эффективное выполнение многомерных запросов.

Третий уровень, хотя и является широко распространенным, не обязателен, так как данные для многомерного представления могут извлекаться и из обычных реляционных структур; процессор многомерных запросов в этом случае транслирует многомерные запросы в SQL-запросы, которые выполняются реляционной СУБД.

Конкретные OLAP-продукты, как правило, представляют собой либо средство многомерного представления данных — например, Pivot Tables в Excel 2000 фирмы Microsoft или ProClarity фирмы Knosys, либо многомерную серверную СУБД — например, Oracle Express Server или Microsoft OLAP Services.

Слой многомерной обработки обычно бывает встроен в OLAP-клиент и/или в OLAP-сервер, но может быть выделен в чистом виде, как, например, компонент Pivot Table Service фирмы Microsoft.

**Технические аспекты многомерного хранения данных.** Как уже говорилось выше, средства OLAP-анализа могут извлекать данные и непосредственно из реляционных систем. Такой подход был более привлекательным в те времена, когда OLAP-серверы еще не существовали. Но сегодня и Oracle, и Informix, и Microsoft предлагают полноценные OLAP-серверы.

OLAP-серверы, или серверы многомерных БД, могут хранить свои многомерные данные по-разному. Прежде чем рассмотреть способы хранения, нам нужно поговорить о таком важном аспекте, как хранение агрегатов. Дело в том, что в любом хранилище данных (и в обычном, и в многомерном) наряду с детальными данными, извлекаемыми из оперативных систем, хранятся и суммарные (агрегированные) показатели — суммы объемов продаж по месяцам, категориям товаров и т. п. Агрегаты хранятся в явном виде с единственной целью — ускорить выполнение запросов. Ведь с одной стороны, в хранилище накапливается, как правило, очень большой объем данных, а с другой, — аналитиков в большинстве случаев интересуют не детальные, а обобщенные показатели. И если каждый раз для вычисления суммы продаж за год пришлось бы суммировать миллионы индивидуальных продаж, скорость была бы неприемлемой. Поэтому при загрузке данных в многомерную БД вычисляются и сохраняются все суммарные показатели или их часть.

Однако за скорость обработки запросов к суммарным данным приходится платить увеличением объемов данных и времени на их загрузку. Причем увеличение объема может стать буквально катастрофическим. Степень увеличения количества данных при вычислении агрегированных показателей зависит от количества измерений куба и структуры этих измерений. Для решения проблемы хранения агрегатов применяются подчас сложные схемы, позволяющие при вычислении далеко не всех возможных агрегированных показателей достигать значительного повышения производительности выполнения запросов.

Существуют три способа хранения данных по технологии OLAP.

Теперь о различных вариантах хранения информации. Как детальные данные, так и агрегированные могут храниться либо в

реляционных, либо в многомерных структурах. Многомерное хранение позволяет обращаться с данными как с многомерным массивом, благодаря чему обеспечиваются одинаково быстрые вычисления суммарных показателей и различные многомерные преобразования по любому из измерений. Некоторое время назад OLAP-продукты поддерживали либо реляционное, либо многомерное хранение. Сегодня, как правило, один и тот же продукт обеспечивает оба этих вида хранения, а также третий вид — смешанный. Применяются следующие термины.

1. Детальные и агрегированные данные хранятся в многомерной БД. В этом случае создается избыточность, так как многомерные данные полностью содержат реляционные MOLAP — (Multidimensional OLAP).

2. Детальные данные остаются там, где они «жили» изначально — в реляционной БД; агрегированная информация собрана в той же базе данных в специально созданных служебных таблицах ROLAP — (Relational OLAP).

3. Детальные данные остаются на месте (в реляционной базе данных), а агрегаты хранятся в многомерной базе данных HOLAP — (Hybrid OLAP).

Каждый из этих способов имеет свои преимущества и недостатки и должен применяться в зависимости от условий — объема данных, мощности реляционной СУБД и т. д.

При хранении данных в многомерных структурах возникает потенциальная проблема «разбухания» за счет хранения пустых значений. Ведь если в многомерном массиве зарезервировано место под всевозможные комбинации меток измерений, а реально заполнена лишь меньшая часть (например, ряд продуктов продается только в небольшом числе регионов), то большая часть куба будет пустовать, хотя место будет занято. Современные OLAP-продукты умеют справляться с этой проблемой.

### **5.6.2. Технологии интеллектуальной обработки данных**

Интеллектуальный анализ данных (ИАД) — это процесс поддержки принятия решений, основанный на поиске в данных скрытых закономерностей. Большинство методов ИАД было первоначально разработано в рамках теории искусственного интеллекта в 1970—1980-х гг., но получило распространение только в последние годы, когда возникла проблема обработки быстрорастущих объемов корпоративных данных. Сырьем для интеллектуального анализа данных могут быть таблицы реляционных СУБД. Именно с них и начиналась история интеллектуального анализа

данных. Во многих случаях более эффективным является применение ИАД к данным, полученным после обработки с помощью OLAP-технологий.

Задачи интеллектуального анализа данных классифицируются прежде всего по типам извлекаемой информации, т.е. по видам находимых закономерностей. Единого мнения относительно того, какие задачи следует относить к интеллектуальному анализу данных, нет. Большинство авторитетных источников относят к интеллектуальной обработке классификацию, кластеризацию, выявление ассоциаций, поиск последовательностей, прогнозирование.

*Классификация* — наиболее простая и распространенная задача интеллектуального анализа. В результате решения задачи классификации обнаруживаются признаки, которые характеризуют группы объектов исследуемого набора данных — классы; по этим признакам новый объект можно отнести к тому или иному классу. Классификация позволяет выявить признаки, характеризующие однотипные группы объектов. Предполагается, что характеристики классов заранее (до анализа) известны. В качестве методов решения задачи классификации применяют индукцию деревьев решений, нейронные сети и т.д.

*Кластеризация* распространяет идеи классификации на более сложный случай, когда сами классы не predetermined. Задача кластеризации значительно сложнее задачи классификации. В результате выполнения процедуры кластеризации исходные данные разбиваются на однородные группы (кластеры). Это позволяет выработать по отношению к каждой из групп (например, к группам покупателей) определенную политику. Кластеризация является описательной процедурой, она не делает никаких статистических выводов, но дает возможность провести разведочный анализ и изучить «структуру данных». Само понятие «кластер» определено неоднозначно: в каждом исследовании свои «кластеры». Переводится понятие кластер (cluster) как «скопление», «гроздь». Кластер можно охарактеризовать как группу объектов, имеющих общие свойства. Характеристиками кластера можно назвать два признака: внутреннюю однородность и внешнюю изолированность. Наибольшее применение кластеризация первоначально получила в таких науках, как биология, антропология, психология. Для решения экономических задач кластеризация длительное время мало использовалась из-за специфики экономических данных и явлений.

*Выявление ассоциаций* — это поиск закономерностей в данных, фиксирующих одновременное наступление двух (или более) событий. Типичный пример ассоциации исследуется в задаче определения пар одновременно покупаемых продуктов (телевизоры и

видеомагнитофоны, зубные пасты и зубные щетки и т.д.). Ассоциативные правила могут применяться в розничной торговле (определение товаров, которые стоит продвигать совместно), для сегментации клиентов (выявление общих характеристик клиентов компании, выявление групп покупателей) и т.д.

*Поиск последовательностей* — выявление закономерностей в данных, при происхождении событий с некоторым определенным разрывом во времени. Последовательность позволяет найти временные закономерности между транзакциями (множеством событий, которые произошли одновременно). Другими словами, последовательность определяется высокой вероятностью цепочки связанных во времени событий. Фактически ассоциация является частным случаем последовательности с временным лагом, равным нулю. Правило последовательности: после события X через определенное время произойдет событие Y. Так, если видеомагнитофон не был куплен вместе с телевизором, то в течение месяца после покупки нового телевизора покупка видеомагнитофона производится в 60 % случаев.

*Прогнозирование* — формализованная процедура предсказания, которая на основе исследования текущих и прошлых данных позволяет оценить будущие значения числовых показателей. В задачах подобного типа наиболее часто используют традиционные методы математической статистики, а также нейронные сети. Прогнозирование (от гр. *prognosis*) в широком понимании этого слова определяется как опережающее отражение будущего. Прогнозирование направлено на определение тенденций динамики конкретного объекта или события на основе ретроспективных данных, т.е. анализа его состояния в прошлом и настоящем. Таким образом, решение задачи прогнозирования требует некоторой обучающей выборки данных. Прогнозирование можно также определить как установление функциональной зависимости между зависимыми и независимыми переменными. Прогнозирование является распространенной и востребованной задачей во многих областях человеческой деятельности. В результате прогнозирования уменьшается риск принятия неверных, необоснованных или субъективных решений.

В самых общих чертах решение задачи прогнозирования сводится к решению таких подзадач: — выбор модели прогнозирования, анализ адекватности и точности построенного прогноза.

Примеры задач прогнозирования: движение денежных средств, урожайность агрокультуры, финансовая устойчивость предприятия. Типичной в сфере маркетинга является задача прогнозирования рынков (*market forecasting*). В результате решения данной задачи оцениваются перспективы развития конъюнктуры определенного рынка, изменения рыночных условий на будущие перио-

ды, определяются тенденции рынка (структурные изменения, потребности покупателей, изменения цен).

Помимо экономической и финансовой сферы задачи прогнозирования ставятся в самых разнообразных областях: медицине, фармакологии; популярным сейчас становится политическое прогнозирование.

## 5.7. Внедрение корпоративных информационных систем

Под бизнес-процессом принято понимать цепь логически связанных повторяющихся действий, которые совместно реализуют некую бизнес-задачу или цель предприятия. В то же время бизнес-процесс — это поток работ (событий), протекающий от одного рабочего места к другому, совершающийся в течение какого-либо периода и обладающий результатом (рис. 5.10).

Современная процессно-ориентированная организация — это совокупность специализированных функциональных отделов, с одной стороны, и совокупность бизнес-процессов, с другой. В каждом из отделов реализуются отдельные функции бизнес-процессов, а сотрудники таких организаций, помимо классического функционального подчинения, подчиняются в рамках выполняемых бизнес-процессов соответствующим владельцам этих процессов.

В настоящее время многие отечественные предприятия являются функционально-ориентированными организациями, структура которых в отличие от процессных организаций имеет вертикальную топологию, построенную в соответствии с выполняемыми

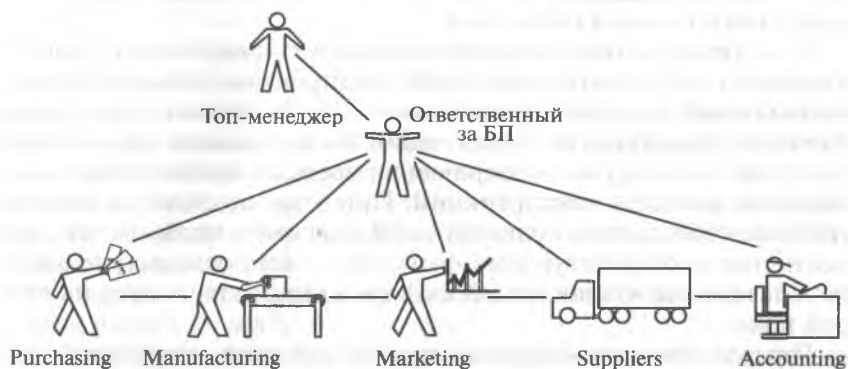


Рис. 5.10. Пример бизнес-процесса



ми функциями, и строгую иерархическую подчиненность сверху вниз.

Недостатки такой организации — это и отсутствие владельцев процессов, ответственных за конечный результат, и наличие непроизвольной разрушительной конкуренции между подразделениями, и оторванность сотрудников от конечного результата. Бизнес-процессы таких предприятий сегментированы, т. е. существуют в рамках отдельно взятых функциональных подразделений, а эффективность функций, выполняемых отдельными структурами, зачастую достигается в ущерб эффективности всего процесса. В такой организации чрезвычайно усложнены взаимодействие и обмен информацией между подразделениями, попытка внедрить на подобных предприятиях информационную систему путем последовательной автоматизации отдельных функций приводит в лучшем случае к невозможности интегрировать внедренную функциональность, а в худшем — к провалу проекта. Затратив значительные средства, предприятие не получает ожидаемой отдачи от инвестиций.

При внедрении информационных систем в функционально-ориентированных организациях весьма острой оказывается проблема перестройки деятельности предприятия в контексте осмысления и совершенствования бизнес-процессов. Изменить ситуацию призван процессный подход, т. е. применение в организации системы процессов наряду с их идентификацией и взаимодействием. Преимущества этого подхода становятся очевидны при сравнении процессной и функциональной организации, а дополнительным аргументом в его пользу является ориентация на применение процессного подхода в системе менеджмента качества.

Функциональными или процессными могут быть и подходы к внедрению информационных систем. При этом существенные различия двух разных подходов хорошо заметны уже на стадии подготовки проекта внедрения.

Если предприятие собирается автоматизировать деятельность отдельных сотрудников или служб предприятия, то налицо функциональный подход: стремление автоматизировать отдельные функции предприятия. Наилучший из возможных результатов такой автоматизации — сокращение времени выполнения и повышение качества этих функций. При этом от системы обычно требуется обеспечить пользователям максимум удобства при выполнении соответствующих функций, а вопросы дальнейшего использования возникающей информации отодвигаются на второй план.

Гораздо большего эффекта можно добиться, применив процессный подход и осуществив процессное внедрение. Объектом автоматизации в этом случае служат сквозные бизнес-процессы.

При постановке задачи очень важно правильно идентифицировать те из них, которые должны быть реализованы с использованием информационной системы. Разумеется, выбор автоматизируемых процессов должен соответствовать корпоративной стратегии повышения эффективности. Выбранные бизнес-процессы подвергаются анализу и затем проектируются с точки зрения реализации в информационной системе. При таком подходе достигается синергетический эффект от автоматизации отдельных функций, поскольку в системе организована совместная деятельность сотрудников и служб предприятия. На основании спроектированных процессов определяется объем внедряемой функциональности (конфигурация рабочих мест), которая покрывает потребности процессов, и только после этого происходит реализация выбранных процессов в системе.

Деятельность, предшествующая реализации, относится к стадии подготовки проекта внедрения. В случае функционального внедрения подготовка занимает непродолжительное время: на начальных этапах функциональный подход может принести быстрый результат. Подготовка процессного внедрения требует довольно продолжительного времени и более существенных затрат, но это внедрение обеспечивает результаты принципиально иного уровня, многократно превосходящие все возможные преимущества первого варианта.

Говоря о процессном внедрении, нельзя не упомянуть об инструментарии, применяемом для моделирования бизнес-процессов. Сам по себе процессный подход не предъявляет особых требований к инструментам описания и проектирования бизнес-процессов, однако использование специализированных инструментов вместо стандартных офисных программ имеет массу неоспоримых преимуществ, например:

- представление бизнес-процессов в виде графических моделей;
- наличие единого стандарта моделирования;
- ориентация на процессный подход;
- наличие единой информационной базы, позволяющей использовать в разных диаграммах одни и те же объекты, совмещая различные точки зрения на организацию;
- возможность генерации разнообразных отчетов по разработанной модели — в том числе и отчетов, специально разработанных пользователем;
- возможность организации совместной работы в сетях Интернет и Интранет и т. д.

Что же дает внедрение информационной системы в случае применения процессного подхода и совершенствования бизнес-процессов?

Единая информационная среда источников и потребителей информации, прежде всего, позволяет кардинально изменить назначение бумажного документа и рассматривать его не как носитель информации, а как отчет, сформированный на основе соответствующего информационного объекта базы данных. Бумажный документ становится носителем юридического статуса и представляет собой набор данных из базы, распечатанный на бланке. При этом файл (электронный документ) сохраняется в централизованном электронном архиве, являющимся неотъемлемой частью системы, и связывается с объектом базы данных, на основании которого он был получен. Создатели информации (конструкторы, финансисты, технологи и т.д.) и ее потребители работают с соответствующим информационным объектом напрямую, имея при этом доступ к электронным документам в рамках прав, предоставленных им системой.

Перечислим основные преимущества рассматриваемого способа работы — с точки зрения организации процессов на предприятии.

1. Реальная совместная работа с информацией в большинстве случаев позволяет перейти от последовательного способа обработки информации к параллельному. Другими словами, появляется возможность распараллелить бизнес-процесс, существенно сократить сроки разработки и сэкономить время для таких операций, как согласование, утверждение документации, внесение изменений.

2. Работа в единой информационной среде делает процесс прозрачным и управляемым; каждый его участник видит и результат, и собственную роль в процессе. Подобная организация работы позволяет выстроить в рамках процесса цепочки взаимодействия функциональных подразделений и отдельных сотрудников.

3. При проектировании процессов с учетом использования информационной системы, как правило, выявляется ряд документов, полностью или частично дублирующих друг друга, а также документы, которые вообще могут быть выведены из употребления, поскольку содержащаяся в них информация может быть получена гораздо более эффективным способом.

4. Документ, получаемый в виде отчета из базы данных и сохраненный в архиве, становится частью информационной базы предприятия и его интеллектуальной собственностью. Это снижает влияние человеческого фактора, а также риск искажения или утраты информации.

К сожалению, перечисленные плюсы подобного способа работы с информацией создают определенные проблемы при внедрении информационных систем. Функциональные подразделения предприятия обычно предпочитают наводить порядок в своей

функциональной области и не склонны становиться частью большого целого. Подразделение стремится оттачивать и совершенствовать собственные функции, не слишком задумывается об эффективности всего процесса и без особого энтузиазма воспринимает необходимость реорганизации деятельности в соответствии с требованиями оптимизации бизнес-процессов.

Принципы, на которых базируются современные информационные системы, предполагают организацию совместной деятельности сотрудников предприятия и являются выражением процессного подхода. Принимая процессный подход, предприятие непременно должно принять концепцию процессного внедрения и согласиться с ней. В противном случае проект будет обречен на неудачу с самых первых шагов.

Известно, что идеальных систем не бывает. При внедрении информационная система накладывает некоторые ограничения на способы реализации процессов, поэтому проектирование процессов «как должно быть» оказывается неизбежным компромиссом между требованиями процесса и возможностями информационной системы. Важно, что информационная система способна обеспечить сквозную, не сегментированную автоматизацию бизнес-процессов в целом, а также эффективное использование данных для решения задач производственного планирования и учета.

## 5.8. Управление информационными рисками

Обеспечение информационной безопасности — одна из главных задач современного предприятия. Угрозу могут представлять не только технические сбои, но и несогласованность данных в различных учетных системах, которая встречается едва ли не у каждой второй компании, а также неограниченный доступ сотрудников к информации. Учитывая сказанное, очень важно выявить и минимизировать информационные риски (IT-риски).

**И н ф о р м а ц и о н н ы е р и с к и** — это опасность возникновения убытков или ущерба в результате применения компанией информационных технологий. Иными словами, IT-риски связаны с созданием, передачей, хранением и использованием информации с помощью электронных носителей и иных средств связи.

IT-риски можно разделить на две категории:

- риски, вызванные утечкой информации и использованием ее конкурентами или сотрудниками в целях, которые могут повредить бизнесу;
- риски технических сбоев работы каналов передачи информации, которые могут привести к убыткам.

Работа по минимизации ИТ-рисков заключается в предупреждении несанкционированного доступа к данным, предотвращении аварий и сбоев оборудования. Процесс минимизации ИТ-рисков следует рассматривать комплексно: сначала выявляют возможные проблемы, а затем определяют, какими способами их можно решить. На практике способы выявления ИТ-рисков ничем не отличаются от способов определения любых других рисков: составляются карты рисков, проводится сбор экспертных мнений и т. п.

Выявить наиболее критичные информационные риски можно и более простым способом, ответив на следующие вопросы.

1. Способна ли фирма контролировать доступ к информационным системам, в которых формируется и хранится финансовая отчетность?

2. Обеспечены ли клиенты фирмы необходимой информационной поддержкой?

3. Сможет ли фирма в короткий срок интегрировать существующие технологии работы с информацией в системы предприятия, являющегося объектом слияния или приобретения? Например, в компании установлена одна или несколько учетных систем, с помощью которых финансисты получают данные для составления консолидированной отчетности. При покупке нового предприятия выясняется, что у него установлена другая учетная система. Поэтому у фирмы должен быть четкий план трансформации такой отчетности в стандарты, принятые на головном предприятии. В противном случае она может потерять оперативный контроль над ситуацией.

4. Позволяет ли организация документооборота фирмы в существующих системах продолжить ее деятельность в прежнем режиме в случае ухода ключевых сотрудников? Эта проблема чрезвычайно актуальна для российских фирм, поскольку даже финансовая и бухгалтерская информация зачастую вводится и хранится в произвольном виде, не говоря уже о сведениях, касающихся клиентов и т. п. Это ведет к дополнительным затратам времени новых сотрудников на «вхождение» в курс дела и повышает вероятность возникновения ошибок.

5. Обеспечена ли защита интеллектуальной собственности фирмы и ее клиентов?

6. Имеет ли фирма четкий алгоритм действий в критической ситуации, например в случае сбоев в работе компьютерных сетей или вирусной атаки?

7. Соответствует ли способ работы информационных систем общим задачам фирмы? (Если перед фирмой стоит задача иметь общий центр управления денежными потоками, а учетные системы, установленные в разных филиалах, не связаны между собой, то поставленная задача не будет решена.)

Точно определить возможный ущерб от большинства IT-рисков довольно сложно, но примерно оценить их вполне возможно.

Как показывает опыт многих российских компаний, наиболее успешные стратегии предупреждения IT-рисков базируются на трех основных правилах.

Во-первых, доступ сотрудников к информационным системам и документам предприятия должен быть различен в зависимости от важности и конфиденциальности содержания документа. Во-вторых, необходимо контролировать доступ к информации и обеспечивать защиту уязвимых мест информационных систем. В-третьих, информационные системы, от которых напрямую зависит деятельность предприятия (стратегически важные каналы связи, архивы документов, компьютерная сеть), должны работать бесперебойно даже в случае кризисной ситуации.

Как обеспечить приемлемый уровень безопасности для систем корпоративного уровня?

Защита информации — это обеспечение непрерывности внутренних бизнес-процессов и безопасности обмена данными с клиентами при использовании информационных систем. На случай возникновения чрезвычайных ситуаций, таких как атаки на основной сервер или сбой в его работе, журнал транзакций (т.е. перечень операций, производимых в системе) периодически должен копироваться на резервный сервер, находящийся в другом помещении. При такой организации максимальный объем информации, который может быть потерян, — это данные за последний час работы. Для доступа клиентов к данным через сети общего пользования, например Интернет, выделяется отдельный сервер. На него копируются данные с основного сервера, так что даже если нет возможности отразить атаку взломщиков, они не получат доступа к внутренней информации компании, а также к электронным архивам, которые хранятся на носителях, не имеющих выхода в Интернет. Доступ клиентов к данным должен быть обеспечен различными степенями защиты, которые гарантируют конфиденциальность и достоверность передаваемой информации.

К IT-рискам относят потерю данных из-за сбоя в работе информационных систем, хищение информации, а также передачу информации третьим лицам сотрудниками предприятия. Работа по минимизации таких рисков делится на организационную и техническую.

Организационная работа связана с ограничением доступа к данным. Для этого вся информация классифицируется на общедоступную, для служебного пользования и секретную. Кроме того, содержание информационных потоков можно разделить по назначению:

- данные, которые циркулируют внутри рабочей группы (по определенному проекту);
- данные, предназначенные для исполнителей и руководителей подразделений (зароботная плата, индивидуальные задачи и т. п.);
- данные для руководителей подразделений и топ-менеджмента (планы стратегического развития).

В итоге получается матрица информационных потоков, каждому уровню которой соответствует определенный уровень доступа.

Разработкой регламентов, касающихся информационной безопасности, занимается специальный отдел. Основываясь на этих регламентах, каждый руководитель подразделения формирует для своих сотрудников должностные инструкции и назначает ответственных за соблюдение информационной безопасности в рамках своего подразделения.

*Техническая работа* по обеспечению информационной безопасности заключается в дублировании важных функций, от которых зависят сохранность и целостность информации, а также непрерывность работы компании (например, установка запасных серверов, систем резервного копирования). Чтобы минимизировать риск сбоев, нужно использовать только технику от надежных производителей. Затраты на нее окупаются, так как убыток от простоя информационных систем в течение нескольких часов многократно превысит их стоимость, а потеря информации может вообще парализовать работу предприятия.

Обеспечение информационной безопасности — это, в первую очередь, вопрос эффективности затраченных средств, поэтому расходы на защиту не должны превышать суммы возможного ущерба.

Поскольку любые расходы на предотвращение рисков должны быть обоснованы, необходимо обязательно рассчитывать их экономическую эффективность.

Для обеспечения необходимой защиты от ИТ-рисков и контроля безопасности можно провести следующие мероприятия.

1. Определить круг лиц, отвечающих за информационную безопасность, создать нормативные документы, в которых будут описаны действия персонала компании, направленные на предотвращение ИТ-рисков, а также обеспечить резервные мощности для работы в критической ситуации.

2. Разработать единые стандарты информационных систем в рамках организации, т.е. перейти к единым отчетным формам, а также единым правилам расчета показателей, которые будут применяться во всех программных продуктах компании, используемых для этой цели.

3. Классифицировать данные по степени конфиденциальности и разграничить права доступа к ним.

4. Следить за тем, чтобы любые документы, обращающиеся внутри организации, создавались с помощью систем, централизованно установленных на компьютерах. Установка любых других программ должна быть санкционирована, иначе риск сбоев и вирусных атак резко возрастет.

5. Внедрить средства контроля, позволяющие отслеживать состояние всех корпоративных систем: в случае несанкционированного доступа система должна или автоматически запрещать вход, или сигнализировать об опасности, чтобы персонал мог принять меры.

Помимо перечисленных мер необходимо подготовиться к последствиям возможных кризисных ситуаций и описать действия компании по выходу из кризиса. Для этого следует:

- проанализировать сценарии проникновения посторонних лиц или не имеющих соответствующих полномочий сотрудников компании во внутреннюю информационную сеть, а также провести учебные мероприятия с целью отработки модели поведения сотрудников, ответственных за информационную безопасность, в кризисных ситуациях;

- разработать варианты решения проблем, связанных с кадрами, включая уход из компании ключевых сотрудников, например составить и ознакомить персонал с планом преемственности управления на предприятии;

- подготовить запасные информационные мощности (серверы, компьютеры), а также резервные линии связи.

В заключение отметим, что разработка и реализация политики по минимизации ИТ-рисков не принесет пользы, если рекомендуемые стандарты и правила неверно используются, например если сотрудники не обучены их применению и не понимают их важности. Поэтому работа по обеспечению ИТ-безопасности должна быть комплексной и продуманной.

## **5.9. Экономическая эффективность информационных систем**

В настоящее время уровень затрат на информационные технологии предприятия приближается, а иногда и превышает уровень инвестиций в другие производственные процессы, вместе взятые, поэтому обеспечение оптимальной стоимости ведения бизнеса на основе информационных систем становится одной из основных задач поддержания требуемого уровня конкурентоспособности



организации на рынке. Процесс оценки стоимости ИС является весьма сложной задачей. Он начинается на стадии анализа требований к информационной системе и продолжается даже на этапе реализации. Существует множество подходов к решению этой задачи, но до сих пор не существует универсальной методики, дающей гарантированный результат.

В сложившихся условиях проблема оценки эффективности вложений в ИС становится чрезвычайно актуальной. Какой должна стать корпоративная информационная система? Какие инвестиции требуются для ее создания и сопровождения? Как обосновать необходимость инвестиций в информационные технологии? Как добиться разумного соотношения между размерами этих инвестиций и той выгодой, которая может быть получена от использования информационных систем? В этой главе мы постараемся ответить на эти вопросы.

Эффективность системы — это степень соответствия своему назначению. Оценка экономической эффективности основана на сопоставлении затрат и результатов, выраженных в денежной форме. Основным требованием к экономической эффективности является требование, чтобы результаты превосходили затраты и соотношение результатов и затрат соответствовало требуемой рентабельности.

В настоящее время выделяют следующие показатели экономической эффективности внедрения программных проектов:

- внутренняя норма дохода (IRR — Internal Rate of Return);
- чистая приведенная стоимость (NVP — Net Present Value);
- срок окупаемости (PB — Payback Period);
- совокупная стоимость владения (TCO — Total Cost of Ownership);
- норма возврата инвестиций (ROI — Return of Investment);
- ценность возможностей для бизнеса (TVO — Total Value of Opportunity).

Зачастую организации учитывают только прямые затраты на приобретение программного обеспечения, в то время как косвенные затраты, связанные с сопровождением и поддержкой (обновление и пр.), как правило, не учитываются.

Зарубежный опыт решения задач оценки эффективности инвестиций в ИС показывает, что одним из наиболее широко распространенных методов является применение концепции оценки Совокупной Стоимости Владения информационных технологий (Total Cost of Ownership — TCO). Под TCO понимается сумма всех затрат на внедрение и обеспечение функционирования информационных систем вплоть до момента вывода ее из эксплуатации. Существует две основные модели расчета Совокупной Стоимости Владения: концепция, предложенная Gartner Group, и результат

совместных усилий Microsoft и Interpose. По методике предложенной Gartner, все затраты делятся на фиксированные и текущие.

*Фиксированные затраты* производятся один раз на этапе внедрения системы. К ним относят стоимость разработки и внедрения проекта, первоначальные закупки необходимого для внедрения ИС аппаратного и программного обеспечения, привлечение внешних консультантов.

*Текущие затраты* — расходы, обеспечивающие функционирование системы. Это те затраты, которые требуются постоянно, пока система работает. К ним относятся:

- обновление и модернизация системы;
- управление системой в целом — администрирование, обучение администрации и конечных пользователей, заработная плата персонала, привлечение внешних ресурсов;
- «активность пользователя» — доработка программного обеспечения, дополнительные настройки программного обеспечения, работа с данными, последствия некомпетентных действий пользователя.

Казалось бы все просто — нужно только подсчитать затраты по каждой из вышеперечисленных статей. Но на самом деле не все расходы легко подсчитать — значительная их часть не только не закладывается заранее, но и нигде не учитывается. Причем 75 % затрат, входящих в состав ТСО, обусловлены проблемами конечных пользователей. В модели ТСО, разработанной Microsoft и Interpose, учитываются как раз эти затраты. Согласно их методике затраты делятся на прямые и косвенные.

Совокупная стоимость владения можно подсчитать по простой формуле

$$ТСО = Пр + Кс,$$

где Пр — прямые затраты; Кс — косвенные затраты.

Прямые затраты предусматриваются бюджетом и планируются, косвенные затраты составляют более 50 % средних расходов, они не поддаются планированию и часто вообще не регистрируются. К ним относятся прежде всего пользовательские затраты (неформальное обучение, персональная поддержка, ошибки и просчеты) и простои из-за выхода оборудования из строя или плановых профилактических остановок. Часто косвенные затраты связаны с недостатками в работе информационных систем, в том числе:

- потери времени сотрудников на самообучение и настройку систем;
- потери времени сотрудников на попытки решения технических проблем с информационными системами своими силами, в обход службы технической поддержки;

— потери времени сотрудников на помощь коллегам в решении вопросов поддержки информационных систем;

— потери (реальные или возможные) предприятия от сбоев в работе IT-системы, когда системы становятся недоступными, что влияет непосредственно на конечных пользователей.

ТСО является ключевым количественным показателем эффективности процессов автоматизации компании, так как позволяет оценить совокупные затраты на информационные технологии (оборудование, инструментальные средства программного обеспечения, процессы сопровождения информационных систем, а также действия конечных пользователей), анализировать их и соответственно управлять IT-затратами (бюджетом) для достижения наилучшей отдачи от информационной системы в организации. ТСО представляет собой не просто отдельный интегральный показатель, но целую систему показателей, соответствующих различным статьям расходов.

Для количественного анализа используются специальные программные средства типа «ТСО Manager». Сам по себе расчет не сложен и фактически реализуется на уровне электронных таблиц. Большую ценность имеет возможность сравнения результатов ТСО для конкретной организации с представительной базой, созданной по результатам анализа показателей в других компаниях аналогичного профиля. Gartner ведет обширную базу данных клиентов по измерению показателей ТСО для различных индустрий и характера вычислительной среды. Сравнение может производиться как со средними показателями по выбранной группе аналогичных компаний, так и с «лучшими в группе». Средние и лучшие показатели рассчитываются и отслеживаются экспертами Gartner по многим предприятиям различных отраслей, так что база этих показателей периодически обновляется. Возможен также анализ типа «что будет, если», который позволяет смоделировать возможное влияние планируемых решений в области архитектуры — например, по консолидации серверов, на показатель ТСО.

Важный аспект заключается еще и в том, что косвенные затраты являются отражением того, насколько эффективно осуществляется управление информационными системами. Чем эффективнее IT-служба обеспечивает управление IT-инфраструктурой и системами, тем меньше косвенные затраты, связанные с простоем систем или потерями времени конечных пользователей на решение проблем и получение помощи от своих коллег. Поэтому неразумная экономия на прямых затратах может фактически привести только к лишним косвенным расходам и, как следствие, к лишним расходам для организации в целом.

Прямые затраты включают в себя следующие статьи:

— затраты на аппаратное и программное обеспечение: покупка, амортизационные отчисления и траты на обновление оборудования, запасные части и расходные материалы, расходы на покупку и обновление программного обеспечения (серверные и клиентские операционные системы, офисные приложения, лицензии на базы данных и системы электронной почты и коллективной работы, программное обеспечение для управления ИТ-инфраструктурой и приложениями);

— затраты на управление и поддержку: затраты на внутренний и внешний персонал, который обеспечивает техническую поддержку, все вопросы, связанные с планированием и управлением инфраструктурой (серверами и настольными компьютерами, сетями, прикладными системами, системами хранения данных и т. д.);

— административные расходы: затраты на внутренний и внешний персонал, который занимается административными и финансовыми вопросами, закупками, выполняет контрольные функции; обучение ИТ-персонала, обучение конечных пользователей.

Важно обратить внимание на то, что ТСО существенным образом зависит от эффективности управления инфраструктурой ИТ-службой. Это значительно влияет на экономию от перехода на более новые версии программного обеспечения.

Сами информационные системы являются только одной составляющей деятельности компании, а получение реальных преимуществ зависит от способности организации эффективно их использовать.

В состав типового проекта по оценке ТСО входят такие работы, как:

— сбор и консолидация данных по затратам на информационные системы;

— анализ затрат на информационные системы;

— моделирование ситуаций информационного развития компании;

— расчет затрат и выгод от внедрения технических средств;

— анализ возврата инвестиций в информационные системы и информационные технологии;

— сравнение показателей информационных технологий организации с показателями аналогичных компаний;

— оценка работы персонала службы информационных технологий;

— оценка влияния новых информационных технологий на деятельность персонала организации в целом;

— разработка рекомендаций по оптимизации ТСО.

Таким образом, ТСО выступает в качестве одного из инструментов при выборе информационной системы. В то же время следует учитывать, что эта роль ТСО может меняться в зависимо-

сти от изменения миссии информационных систем в организации. Наибольший вес факторы, связанные с ТСО, будут иметь в том случае, когда информационная система рассматривается как необходимая, но не критическая составляющая поддержки бизнеса организации, такая же, как, например, бухгалтерия или хозяйственная служба. В этом плане развитие архитектуры информационных систем направлено прежде всего на сокращение ТСО. Величина ТСО может быть снижена за счет таких факторов, как уменьшение административных расходов, увеличение скидок, связанных с объемами закупок, улучшение использования оборудования и программного обеспечения за счет концентрации квалифицированных кадров и повышения общей надежности систем. Кроме того, наличие стандартизованных конфигураций создает предпосылки для ускорения реализации проектов.

В то же время на определенных этапах роста или преобразований в организации величина ТСО может по объективным причинам весьма сильно превышать средние показатели по отрасли. В частности, переход от хаотичной к стандартной архитектуре, вероятно, потребует достаточно длительного «периода сосуществования» старых и новых систем, сопровождающегося увеличенными прямыми расходами и рисками.

Эффективность вложений (возврат инвестиций, ROI) — это показатель, характеризующий выгоду внедрения информационной системы. Он рассчитывается как отношение дисконтированных поступлений, ожидаемых от внедрения данной ИС к начальной стоимости инвестиций:

$$ROI = \text{Эф} / \text{И},$$

где Эф — эффект от внедрения, выраженный в денежных единицах; И — инвестиции в информационные системы.

Модель ROI также принадлежит Gartner Group. Для оценки доходной части, как правило, анализируют цели бизнеса, которые нужно достичь путем внедрения программного проекта или проявления каких-либо новых программных продуктов; используют измеримые показатели бизнеса, например сокращение операционных расходов, поддержку конкурентоспособного состояния, улучшение внутреннего контроля, и по ним делают оценки эффекта. В качестве расходной части чаще всего применяют показатель ТСО.

### **Контрольные вопросы**

1. Разъясните понятие информационной технологии управления.
2. Каковы отличия исторического развития западных и отечественных информационных экономических систем?

3. Охарактеризуйте локальные ИС, применяемые для автоматизации управления на предприятиях. В чем состоят недостатки таких ИС?

4. Что означает полнофункциональность ИС?

5. Какие уровни управления существуют на современном предприятии?

6. В чем особенность систем поддержки принятия решений?

7. Какие математические методы применяют при создании СППР?

8. Охарактеризуйте возможности современных интегрированных корпоративных информационных систем.

9. Охарактеризуйте возможности ИС класса MRP.

10. Охарактеризуйте возможности ИС класса CRP.

11. Назовите преимущества систем MRP II.

12. В чем заключается концепция ERP?

13. Какие вопросы содержит тест для определения степени соответствия систем стандарту ERP?

14. Охарактеризуйте основные технологии, используемые для решения задач поддержки принятия управленческих решений.

15. Опишите модель многомерных кубов, лежащую в основе OLAP-технологии.

16. Какие основные типы закономерностей можно выявить с помощью систем интеллектуального анализа данных?

17. Назовите преимущества и недостатки процессного и функционального внедрения ИС.

18. Назовите проблемы, возникающие при внедрении ИСУ на предприятии, и пути их решения.

19. Назовите виды информационных рисков и способы их минимизации.

20. На чем основана оценка экономической эффективности?

21. Что понимают под TCO?

22. Что представляют собой фиксированные затраты?

23. Что относится к текущим затратам?

24. По какой формуле подсчитывают Совокупную Стоимость Владения?

25. Что представляют собой прямые затраты?

26. Из чего состоят косвенные затраты?

27. Какие работы входят в состав типового проекта по оценке TCO?

28. Как рассчитывается ROI?

## СЛОВАРЬ ТЕРМИНОВ

**Абстрагирование (абстракция)** — включение в модель только тех аспектов проектируемой системы, которые имеют непосредственное отношение к выполнению системой своих функций или своего целевого предназначения.

**Администратор БД** — специалист, имеющий представление об информационных потребностях конечных пользователей и отвечающий за определение, загрузку, защиту и эффективность БД.

**Аспект** — точка зрения на объект классификации, который характеризуется одним или несколькими признаками.

**База данных (БД)** — именованная совокупность структурированных, организованных данных, отображающая состояние объектов и их отношений в определенной предметной области.

**База знаний (БЗ)** — именованная совокупность организованных данных и знаний в определенной предметной области и логические правила манипулирования ими для получения необходимых, в том числе новых, знаний.

**Базовое программное обеспечение** — минимальный набор программных средств, обеспечивающих работу компьютера (операционные системы).

**Банк данных (БнД)** — система специально организованных данных, программных, языковых, организационных и технических средств, предназначенных для централизованного накопления и коллективного многоцелевого использования данных.

**Безопасность ИС** — свойство, заключающееся в способности системы обеспечить конфиденциальность и целостность информации, т.е. защиту информации от несанкционированного доступа.

**Бизнес-процесс** — это совокупность взаимосвязанных действий или операций, которые используют ресурсы предприятия для создания или получения продукции (работ, услуг), необходимой потребителю.

**Возврат инвестиций (ROI — Return of Investment)** — показатель, характеризующий выгоду программного проекта, рассчитывается как отношение дисконтированных поступлений, ожидаемых от внедрения данной информационной системы к начальной стоимости инвестиций.

**Гипертекст** — обычный текст, который содержит ссылки на связанные по смыслу фрагменты текста того же или другого документа.

**Данные** — информация, представленная в виде, позволяющем передавать или обрабатывать ее с помощью технических средств.

**Дескриптор** — ключевое слово, определяющее некоторое понятие, которое формирует описание объекта и дает принадлежность этого объекта к классу, группе и т. д.

**Дескрипторная статья** — упорядоченный класс дескрипторов и недескрипторов (синонимов дескрипторов), связанных внеконтекстными отношениями с заглавным дескриптором.

**Диаграммы потоков данных (DFD, Data Flow Diagrams)** — модель описания взаимодействия источников и потребителей информации через реализуемые в системе процессы.

**Диаграммы «сущность-связь» (ERD — Entity-Relationship Diagrams)** — модель описания базы данных системы.

**Документооборот** — последовательность прохождения документов с момента их составления или получения до момента их обработки и использования.

**Достоверность ИС** — свойство системы, обуславливающее безошибочность производимых ею преобразований информации. Достоверность функционирования информационной системы полностью определяется и измеряется достоверностью ее резульатной информации.

**Жизненный цикл ПО (ЖЦ ПО)** — период времени с момента принятия решения о необходимости создания программного обеспечения (ПО) до момента его полного изъятия из эксплуатации. ЖЦ ПО — одно из базовых понятий программной инженерии.

**Запись БД** — совокупность полей, описывающих один объект.

**Знания** — система понятий и отношений между ними.

**Знания понятийные** — знания, выработанные в теоретических науках и используемые для решения определенной задачи.

**Знания конструктивные** — знания, полученные в технике и большей части прикладных наук, о наборах возможных структур объектов и взаимодействии между их частями.

**Знания процедурные** — знания, используемые в выбранной предметной области, методы, алгоритмы и программы, полезные для конкретного приложения, которые можно использовать, передавать и объединять в библиотеки.

**Знания фактографические** — количественные и качественные характеристики объектов и явлений.

**Иерархия** — это формирование из абстракций иерархической структуры, т. е. расположение их по уровням.

**Инструментарий разработки программного обеспечения** — это программные продукты поддержки технологии программирования.

**Информация** — это сведения об объектах, явления, процессах, событиях окружающего мира, уменьшающие неопределенность знаний о них.

**Информационная технология управления** — совокупность методов и средств сбора, регистрации, передачи, накопления и обработки информации на базе программно-аппаратного обеспечения для решения управленческих задач.



**Информационная система** — это взаимосвязанная совокупность средств и методов, обеспечивающих сбор, хранение, обработку, качественное преобразование и выдачу информации с использованием вычислительной техники, компьютерных сетей, средств и каналов связи в интересах достижения поставленной цели в задачах из любой области.

**Инвестиционные критерии** — показатели, характеризующие стоимость и доходность акций предприятия (прибыль на акцию, коэффициент покрытия дивидендов и др.).

**Инкапсуляция** — скрытие элементов объекта, определяющих его устройство и поведение.

**Информационные риски** — это опасность возникновения убытков или ущерба в результате применения компанией информационных технологий.

**Информационный поток** — информация, рассматриваемая в процессе ее движения в пространстве и времени в определенном направлении.

**Искусственный интеллект (ИИ — artificial intelligence, AI)** — свойство технических систем брать на себя отдельные функции интеллекта человека.

**Классификатор** — систематизированный свод (документ, словарь), отображающий закон разбиения множества объектов на классы, группы и т. д. Это также свод кодовых обозначений классификации, наименований этих объектов и фасетов классификационных признаков объектов.

**Кодификатор** — словарь, в котором присвоены коды (символы) обозначаемым понятиям, объектам, сущностям и т. д.

**Контроль управления проектом** — деятельность, связанная с контролем развития проектом, направлением его развития, качеством и использованием ресурсов, а также сравнение этих показателей с плановыми.

**Массив** — множество данных, содержащих достаточно полное описание информационной совокупности, состоящей из однотипных объектов. Смысловое содержание массива обычно указывается в его названии.

**Математическое моделирование** — важнейший трудоемкий и наукоемкий процесс при создании и сопровождении сложных автоматизированных информационных систем, который позволяет в должной степени оценить вероятность успеха, связанные с этим риски, прибыли и ущербы.

**Машины баз данных** — аппаратно-программный мультимикропроцессорный комплекс для работы с базами данных и/или базами знаний, осуществляющий функции СУБД.

**Метазнания** — знания о порядке и правилах применения знаний.

**Многоаспектная система** — это система классификации, которая использует параллельно несколько независимых признаков (аспектов) в качестве основания классификации.

**Модульность** — это свойство системы, связанное с возможностью ее декомпозиции на ряд модулей.

**Модель** — это информационный образ реального объекта, воспроизводящий данный объект (систему) с определенной степенью точности и в форме, часто отличной от формы самого объекта.

**Моделирование** — способ системного анализа проектирования, при котором используют математические или физические модели функционирования всей системы или ее части.

**Монитор** — совокупность программ, которые обеспечивают управление решением задач на ЭВМ в различных режимах. Монитор получает от работающих программ или от оператора команды-директивы управления и организует их выполнение.

**Надежность ИС** — свойство системы сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных условиях применения. Надежность информационных систем является средством обеспечения актуальной и достоверной информации на выходе системы.

**Неструктурированная (неформализуемая) задача** — задача, в которой невозможно выделить элементы и установить между ними связи.

**Обеспечивающая часть ИС** — это совокупность средств, на базе которых (с использованием которых) решаются задачи функциональных подсистем.

**Подтверждение соответствия** — процедура, результатом которой является документальное удостоверение (декларация о соответствии или сертификат соответствия) того, что продукция, процессы (методы) производства, эксплуатации и утилизации работы или услуги соответствуют установленным требованиям технических регламентов или положениям стандартов, условиям гражданско-правовых договоров.

**Поиск** — процедура выделения из некоторого множества объектов подмножества, содержащего только те объекты, которые удовлетворяют некоторому условию.

**Пакеты прикладных программ (ППП)** — совокупность программ, совместимых между собой и обеспечивающих решение задач из некоторой области знаний, называемой предметной областью пакета.

**Показатели деловой активности** — набор показателей для анализа эффективности использования предприятием своих средств, включая оборачиваемость различных типов средств, обязательств и т. п.

**Показатели ликвидности (текущей)** — чистый рабочий капитал и другие показатели, отражающие способность компании удовлетворять претензии держателей краткосрочных долговых обязательств.

**Показатели рентабельности** — набор показателей (коэффициентов), отражающих прибыльность компании и эффективность использования различных ресурсов.

**Показатели устойчивости** — набор показателей, характеризующих финансовую независимость и платежеспособность компании, включают различные соотношения между источниками финансирования (ко-

эefficient финансово́й независимости, коэффициент покрытия процентов и др.).

**Показатели эффективности инвестиций** (дисконтированные Cash-Flow-критерии) — чистый приведенный доход, период окупаемости проекта, внутренняя норма рентабельности и другие показатели, используемые для комплексного анализа проекта.

**Поле** — именованный наименьший элемент записи БД.

**Поток информации** — группа данных, рассматриваемых в процессе движения информации в пространстве и времени в одном направлении. У этих данных есть общий источник и общий приемник.

**Программный модуль** — программный блок, реализующий определенную функциональную возможность и рассчитанный на стандартные формы связи.

**Прикладное программное обеспечение** — программный инструмент решения функциональных задач.

**Протокол** — совокупность правил, определяющих передачу данных между компонентами компьютерной сети. В Интернете действуют два главных протокола: IP (Internet Protocol) — межсетевой протокол и TCP (Transmission Control Protocol) — протокол управления передачей информации.

**Продукция** — результат деятельности, представленный в вещественной форме и предназначенный для дальнейшего использования в хозяйственных и иных целях.

**Реинжиниринг** — процесс, направленный на изучение и изменение унаследованной системы для реконструкции ее проекта и ее повторной реализации в новом виде.

**CASE-технология (CASE — Computer Aided Software Engineering)** — технология создания и сопровождения ПО различных систем.

**Сервисное программное обеспечение** — программы и программные комплексы, которые расширяют возможности базового программного обеспечения и организуют более удобную среду работы пользователя (утилиты для обслуживания компьютера, архиваторы и т. п.).

**Сертификация** — форма подтверждения соответствия, в ходе которого орган по сертификации документально удостоверяет, что продукция, процессы (методы) производства, эксплуатации и утилизации работы или услуги соответствуют установленным требованиям, техническим регламентам или положениям стандартов.

**Сертификат соответствия** — документ, удостоверяющий, что сертифицированная продукция, процессы (методы) производства, эксплуатации и утилизации работы или услуги соответствуют установленным требованиям технических регламентов, положениям стандартов или условиям договора.

**Система** — совокупность взаимодействующих элементов, упорядоченная для достижения одной или нескольких поставленных целей.

**Система управления базами данных (СУБД)** — совокупность методов, языковых и программных средств, предназначенных для создания, ведения и использования БД многими пользователями. СУБД по-

зволяют создавать и хранить большие массивы данных и манипулировать ими.

**Системное программное обеспечение** — совокупность программ и программных комплексов для обеспечения работы компьютера и вычислительных устройств.

**Совокупная стоимость владения ИТ (Total Cost of Ownership — TCO)** — сумма всех затрат на внедрение и обеспечение функционирования информационных систем вплоть до момента вывода из эксплуатации.

**Состояние системы** — фиксация совокупности доступных системе ресурсов (материальных, энергетических, информационных, пространственных, временных, людских, организационных), определяющих ее отношение к ожидаемому результату или его образу, «фотография» механизма преобразования входных данных системы в выходные данные.

**Сообщение** — поток информации, состоящий из смысловых структурных элементов.

**Сканирование** — это технологический процесс, в результате которого создается графический образ бумажного документа.

**Структурированная (формализуемая) задача** — задача, где известны все ее элементы и взаимосвязи между ними.

**Структуризация информации** — процесс представления неформализованной документированной информации в конкретной АИС.

**Тезаурус** — толковый словарь дескрипторов (слов или словосочетаний), значение которых объясняется через связи с другими лексическими единицами (дескрипторами).

**Технологический процесс обработки информации** — комплекс взаимосвязанных операций по преобразованию информации в соответствии с поставленной целью с момента ее возникновения (входа в систему) до момента потребления пользователем.

**Технология Интранет** — простой способ представления текстовой и графической информации в виде гипертекстовых страниц. Используется в информационной службе Интернета. Интранет (intra — внутренний) применяют службы глобальных сетей в локальных (внутренних) сетях.

**Технология программирования** — это совокупность методов и средств для разработки программного обеспечения.

**Управление качеством** — скоординированная деятельность по руководству и управлению организацией применительно к качеству.

**Управление рисками** — организованный процесс распознавания и оценки рисков и реализации средств для поддержки рисков на приемлемом уровне.

**Управляемость** — способность программного продукта предоставлять пользователю возможность управлять этим продуктом и контролировать его.

**Файл БД** — именованная совокупность записей, связанных по каким-либо признакам.

**Фасет** — это аспект классификации, который используется для образования независимых классификационных группировок.

**Фреймы** — специфические объекты, соответствующие понятиям предметной области, имеющие внутреннюю структуру в виде слотов: данных, правил и др.

**Функциональная часть ИС** — это комплекс подсистем, каждая из которых решает некоторое подмножество информационных задач предметной области.

**Эвристики** — неформальные правила рассуждений эксперта.

**Экспертные системы** — это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и распространяющие этот опыт для консультаций менее квалифицированных пользователей.

**Эффективность** — относительный эффект, результативность процесса, операции, проекта, определяемые как отношение эффекта, результата к затратам, расходам, обеспечившим его получение.

**Эффективность ИС** — степень достижения целей, поставленных при создании системы.

## СПИСОК ЛИТЕРАТУРЫ

1. *Анфилатов В. С.* Системный анализ в управлении / В. С. Анфилатов, А. А. Емельянов, А. А. Кукушкин; под ред. А. А. Емельянова. — М.: Финансы и статистика, 2007. — 368 с.
2. *Бочаров Е. П.* Интегрированные корпоративные информационные системы: Принципы построения. Лабораторный практикум на базе системы «Галактика» / Е. П. Бочаров, А. И. Колдина. — М.: Финансы и статистика, 2005. — 288 с.
3. *Вендров А. М.* Практикум по проектированию программного обеспечения экономических информационных систем / А. М. Вендров. — М.: Финансы и статистика, 2002. — 192 с.
4. *Вендров А. М.* Проектирование программного обеспечения экономических информационных систем / А. М. Вендров. — М.: Финансы и статистика, 2000. — 352 с.
5. *Гагарина Л. Г.* Основы технологии разработки программных продуктов / Л. Г. Гагарина, Б. Д. Виснадул, А. В. Игошин. — М.: ФОРУМ: ИНФРА-М, 2006. — 192 с.
6. *Гвоздева В. А.* Основы построения автоматизированных информационных систем / В. А. Гвоздева, И. Ю. Лаврентьева. — М.: ФОРУМ: ИНФРА-М, 2007. — 320 с.
7. *Голицына О. Л.* Базы данных / О. Л. Голицына, Н. В. Максимов, И. И. Попов. — М.: ФОРУМ: ИНФРА-М, 2003. — 352 с.
8. *Емельянова Н. З.* Основы построения автоматизированных информационных систем / Н. З. Емельянова, Т. Л. Партыка, И. И. Попов. — М.: ФОРУМ: ИНФРА-М, 2005. — 416 с.
9. *Иванова Г. С.* Технология программирования / Г. С. Иванова. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2003.
10. *Избачков Ю. С.* Информационные системы / Ю. С. Избачков, В. Н. Петров. — СПб.: Питер, 2005. — 656 с.
11. Информатика / под ред. Н. В. Макаровой. — М.: Финансы и статистика, 2004. — 768 с.
12. *Леоненков А. В.* Самоучитель UML / А. В. Леоненков — СПб.: БХВ-Петербург, 2004. — 432 с.
13. *Мозилев А. В.* Информатика / А. В. Мозилев, Н. И. Пак, Е. К. Хеннер — М.: Изд. центр «Академия», 2000. — 816 с.
14. *Попов И. И.* Автоматизированные информационные системы (по областям применения) / И. И. Попов; под ред. К. И. Курбакова. — М.: Изд-во Рос. экон. акад., 1999. — 103 с.

# ОГЛАВЛЕНИЕ

Предисловие.....	3
------------------	---

## ГЛАВА 1

### ОБЩАЯ ХАРАКТЕРИСТИКА АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

1.1. Состав и структура информационных систем .....	5
1.1.1. Основные понятия информационных систем .....	5
1.1.2. История развития ИС.....	8
1.1.3. Функциональная часть ИС.....	11
1.1.4. Обеспечивающая часть.....	15
1.2. Классификация информационных систем .....	22
1.3. Архитектуры ИС.....	29
1.3.1. Архитектура файл-сервер .....	30
1.3.2. Архитектура клиент-сервер.....	31
1.3.3. Многоуровневая архитектура.....	33
1.3.4. Интернет(Интранет)-технологии .....	35
1.4. Процессы в информационной системе. Эксплуатация информационных систем .....	36
1.4.1. Этапы и виды технологических процессов обработки информации .....	36
1.4.2. Организация сбора, размещения, хранения, накопления, преобразования и передачи данных в информационную систему.....	39
1.4.3. Системы классификации информации.....	43
1.4.4. Экспортирование структур баз данных.....	46
1.5. Показатели эффективности информационных систем .....	47
1.6. Безопасность информационных систем .....	48
1.7. Мировые информационные ресурсы .....	52

## ГЛАВА 2

### ЭКСПЕРТНЫЕ СИСТЕМЫ

2.1. Понятие искусственного интеллекта. Информационная технология экспертных систем .....	55
2.2. Смысл экспертного анализа .....	61

2.3. Характеристики экспертных систем .....	63
2.4. Функции экспертных систем .....	65
2.4.1. Приобретение знаний .....	65
2.4.2. Представление знаний .....	66
2.4.3. Решение задач с использованием знаний .....	68

### ГЛАВА 3

## ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

3.1. Стадии жизненного цикла информационных систем .....	72
3.2. Модели жизненного цикла информационной системы .....	75
3.2.1. Каскадная модель жизненного цикла информационной системы .....	75
3.2.2. Спиральная модель жизненного цикла .....	82
3.3. Процессы жизненного цикла информационной системы .....	85
3.3.1. Основные процессы жизненного цикла .....	86
3.3.2. Вспомогательные процессы жизненного цикла .....	88
3.3.3. Организационные процессы .....	89

### ГЛАВА 4

## МОДЕЛИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

4.1. Понятие модели предметной области .....	92
4.2. Структурный подход в моделировании предметной области .....	94
4.2.1. Сущность структурного подхода .....	94
4.2.2. Методология функционального моделирования SADT .....	95
4.2.3. Диаграммы потоков данных DFD .....	100
4.2.4. Диаграмма «сущность-связь» .....	106
4.3. Объектно-ориентированный подход в моделировании систем .....	111
4.3.1. Сущность объектно-ориентированного подхода .....	111
4.3.2. UML — язык универсального моделирования .....	113
4.3.3. Диаграммы вариантов использования .....	115
4.3.4. Диаграммы деятельности .....	123
4.3.5. Диаграммы последовательности .....	128
4.4. Сравнение структурного и объектно-ориентированного методов моделирования .....	131

### ГЛАВА 5

## ИНФОРМАЦИОННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ

5.1. Понятие информационной технологии управления .....	135
5.2. Локальные и корпоративные информационные системы .....	141
5.3. Классификация информационных систем по уровню управления .....	143



5.3.1. Системы оперативного уровня .....	144
5.3.2. Системы тактического (функционального) уровня .....	147
5.3.3. Системы стратегического уровня .....	149
5.4. Системы поддержки принятия решений .....	151
5.5. Основные методологии создания экономических информационных управляющих систем .....	154
5.5.1. Планирование потребности в материалах (Material Requirements Planning, MRP) и мощностях (Capacity Requirements Planning, CRP) .....	154
5.5.2. Система MRP II .....	158
5.5.3. ERP-системы .....	163
5.5.4. BPM (Business Performance Management) — автоматизация процессов управленческого планирования и контроля .....	166
5.6. Аналитическая обработка данных для поддержки принятия решений .....	167
5.6.1. OLAP (On-line Analytical Processing) — оперативный анализ данных .....	168
5.6.2. Технологии интеллектуальной обработки данных .....	174
5.7. Внедрение корпоративных информационных систем .....	177
5.8. Управление информационными рисками .....	181
5.9. Экономическая эффективность информационных систем .....	185
Словарь терминов .....	192
Список литературы .....	199



*Учебное издание*

**Федорова Галина Николаевна**

**Информационные системы**

**Учебник**

3-е издание, стереотипное

Редактор *Е. Н. Соколова*

Технический редактор *Н. И. Горбачева*

Компьютерная верстка: *Л. А. Смирнова*

Корректоры *Н. В. Савельева, Н. В. Козлова*

Изд. № 103114017. Подписано в печать 15.10.2012. Формат 60 × 90/16.

Гарнитура «Таймс». Бумага офс. № 1. Печать офсетная. Усл. печ. л. 13,0.

Тираж 1 000 экз. Заказ № 3259.

ООО «Издательский центр «Академия». [www.academia-moscow.ru](http://www.academia-moscow.ru)

129085, Москва, пр-т Мира, 101В, стр. 1.

Тел./факс: (495) 648-0507, 616-00-29.

Санитарно-эпидемиологическое заключение № РОСС RU. АЕ51. Н 16067 от 06.03.2012.

Отпечатано с электронных носителей издательства.

ОАО «Тверской полиграфический комбинат», 170024, г. Тверь, пр-т Ленина, 5.

Телефон: (4822) 44-52-03, 44-50-34. Телефон/факс: (4822) 44-42-15.

Home page — [www.tverpk.ru](http://www.tverpk.ru) Электронная почта (E-mail) — [sales@tverpk.ru](mailto:sales@tverpk.ru)