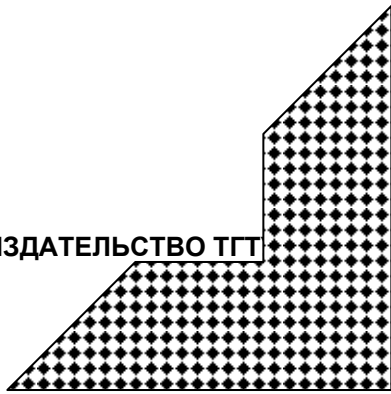


В.А. Погонин, А.Г. Схиртладзе

**ИНТЕГРИРОВАННЫЕ  
СИСТЕМЫ ПРОЕКТИРОВАНИЯ  
И УПРАВЛЕНИЯ  
КОРПОРАТИВНЫЕ  
ИНФОРМАЦИОННЫЕ СИСТЕМЫ**

◆ ИЗДАТЕЛЬСТВО ТГТ



Министерство образования и науки Российской Федерации  
Государственное образовательное учреждение  
высшего профессионального образования  
«Тамбовский государственный технический университет»

**В.А. Погонин, А.Г. Схиртладзе**

# **ИНТЕГРИРОВАННЫЕ СИСТЕМЫ ПРОЕКТИРОВАНИЯ И УПРАВЛЕНИЯ КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ**

*Допущено Учебно-методическим объединением вузов по образованию в области автоматизированного машиностроения (УМО АМ) в качестве учебного пособия для студентов высших учебных заведений, обучающихся по направлению подготовки дипломированных специалистов «Автоматизированные технологии и производства»*



---

Тамбов  
Издательство ТГТУ  
2006

УДК 681.324(03)  
ББК 32.988я22  
П43

Рецензенты:

Доктор физико-математических наук, профессор,  
заведующий кафедрой математической экономики и информатики  
Тамбовского государственного университета им. Г.Р. Державина  
*С.М. Дзюба*

Доктор экономических наук, профессор  
кафедры экономической теории  
Липецкого государственного технического университета  
*В.А. Шайтанов*

**Погонин В.А., Схиртладзе А.Г.**

П43 Интегрированные системы проектирования и управления. Корпоративные информационные системы: Учеб. пособие. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2006. 144 с.

В учебном пособии представлены основные сведения о современных информационных системах. Приведены характеристики основных стандартов IDEF – описания и анализа бизнес-процессов. Изложена методология и техноло-

гии проектирования информационных систем с применением современных CASE-средств. Рассмотрены примеры корпоративных информационных систем, таких как MRP, MRP II, ERP, ERP II.

Рекомендуется в качестве учебного пособия студентам, обучающимся по специальностям «Автоматизация технологических процессов и производств», «Прикладная информатика в экономике».

УДК 681.324(03)

ББК 32.988я22

ISBN 5-8265-0493-5

© Погонин В.А., Схиртладзе А.Г.,  
2006

© Тамбовский государственный  
технический университет  
(ТГТУ), 2006

Учебное издание

ПОГОНИН Василий Александрович,  
СХИРТЛАДЗЕ Александр Георгиевич

# ИНТЕГРИРОВАННЫЕ СИСТЕМЫ ПРОЕКТИРОВАНИЯ И УПРАВЛЕНИЯ КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Учебное пособие

Редактор Т.М. Глинкина  
Инженер по компьютерному макетированию Т.А. Сынкova

Подписано к печати 18.04.2006.

Формат 60 × 84/16. Гарнитура Times. Бумага офсетная. Печать офсетная.

Объем: 8,37 усл. печ. л.; 8,20 уч.-изд. л.

Тираж 100 экз. С. 207

Издательско-полиграфический центр  
Тамбовского государственного технического университета  
392000, Тамбов, ул. Советская, 106, к. 14

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	4
1 ПРЕДПРИЯТИЕ КАК ОБЪЕКТ АВТОМАТИЗАЦИИ .....	6
1.1 История автоматизированных систем управления пред- приятием .....	6
1.1.1 Информационная система .....	9
1.2 Информационное обследование предприятия .....	15
1.3 Реинжиниринг и бизнес-процессы .....	19
1.4 Стандарты описания, анализа и реорганизации бизнес- процессов .....	24
1.4.1 Методология функционального моделирования SADT .....	25
1.4.2 Стандарт IDEF0 .....	34
1.4.3 Стандарт IDEF1 .....	41
1.4.4 Стандарт IDEF1X .....	44
1.4.5 Стандарт IDEF3 .....	52
1.4.6 Методология функционально-стоимостного ана- лиза .....	56
2 ОСНОВЫ МЕТОДОЛОГИИ ПРОЕКТИРОВАНИЯ ИС .....	63
2.1 Жизненный цикл по ИС .....	63
2.2 Модели жизненного цикла ПО .....	65
2.3 Методологии и технологии проектирования ИС (CASE-средства) .....	68
2.4 Структурный подход к проектированию ИС .....	70
2.4.1 Моделирование потоков данных (процессов) DFD	71
2.4.2 CASE-метод Баркера (ERD) .....	76
2.4.3 CASE-средство ERwin .....	81
3 КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ (КИС)	89
3.1 Описание базовых принципов MRP .....	89
3.1.1 MRP-системы .....	90
3.1.2 Планирование производственных мощностей с помощью CRP-системы .....	96
3.1.3 Замкнутый цикл MRP (Closed loop MRP) .....	100
3.1.4 MRP II-система .....	102
3.2 ERP-система .....	118
3.2.1 Логистика в ERP-системах .....	125
3.2.2 Центры обработки данных для ERP-систем .....	128
3.2.3 ERP II-система .....	134
3.3. Другие методики .....	139
ЗАКЛЮЧЕНИЕ .....	141
СПИСОК ЛИТЕРАТУРЫ .....	142

## ВВЕДЕНИЕ

С точки зрения системного подхода, все организации весьма похожи друг на друга. В структуру каждой из них, независимо от рода деятельности, входят многочисленные подразделения, непосредственно осуществляющие тот или иной вид деятельности организации, а также дирекция, бухгалтерия, канцелярия и т.д. Подразделения организации пронизаны вертикальными и горизонтальными связями, они обмениваются между собой информацией, а также выполняют отдельные части одной «большой работы». При этом некоторые из подразделений, например дирекция, финансовые и снабженческие службы, взаимодействуют с внешними партнерами (банк, налоговая инспекция, поставщики сырья, комплектующих, энергии и т.д.), а также филиалами самой организации. Таким образом, любая организация – это совокупность взаимодействующих элементов (подразделений), каждый из которых может иметь свою структуру. Элементы связаны между собой функционально, т.е. они выполняют отдельные виды работ в рамках единого бизнес-процесса, а также информационно, обмениваясь документами, факсами, письменными и устными распоряжениями и т.д. Кроме того, эти элементы взаимодействуют с внешними системами, причем их взаимодействие также может быть как информационным, так и функциональным. И эта ситуация справедлива практически для всех организаций, каким бы видом деятельности они ни занимались – для правительственного учреждения, администрации территориального органа, банка, промышленного предприятия, коммерческой фирмы и т.д.

Такой взгляд на организацию позволяет сформулировать некоторые общие принципы построения корпоративных информационных систем (КИС), т.е. информационных систем в масштабе всей организации (предприятия).

В течение последних лет значительная часть дискуссий, касающихся развития КИС, протекает в ракурсе практического применения современных информационных технологий. Проблема построения комплексных управленческих систем выросла в отдельную ветвь науки об управлении и стала причиной развития целой отрасли высоких технологий. При этом иногда складывается впечатление, что все нынешнее информационное изобилие развивалось и продолжает развиваться таким образом, что становится способным ввести в заблуждение не только неподготовленных читателей, но и самих специалистов в области управления. С одной стороны, это не удивительно, если подходить к данной проблеме с научной точки зрения, так как для любых дисциплин на первоначальном этапе развития характерны отсутствие единой системы терминов и понятий, наличие принципиально разных научных школ и подходов, а также масса других недостатков, вызванных отсутствием достаточного практического опыта. Однако если принимать во внимание интересы всех существующих и потенциальных заказчиков информационных систем, на предприятиях которых производятся дорогостоящие и зачастую плачевные опыты по обкатке и адаптации существующих решений, то фокус необходимо сместить из научно-методической области в область бизнеса. Постараемся в рамках этого пособия хотя бы частично разобраться в путанице характерных определений терминов, понятий и аббревиатур.

## 1 ПРЕДПРИЯТИЕ КАК ОБЪЕКТ АВТОМАТИЗАЦИИ

### 1.1 ИСТОРИЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ ПРЕДПРИЯТИЕМ

Задача планирования и эффективного управления предприятиями – одна из основных областей применения информационных технологий, являющихся базой автоматизированных систем управления (АСУ).

АСУ может быть представлено в виде совокупности автоматизированных систем взаимодействующих уровней (рис. 1.1), условно называемых «управление предприятием» (уровень ERP, MRP), «управление производством» (уровень MES), и «управление технологическими процессами и оборудованием» (уровень DCS).

Уровень ERP реализуется автоматизированными системами управления предприятием (АСУП), уровень DCS – автоматизированными системами управления технологическими процессами (АСУ ТП), а важнейшей функцией уровня MES является сопряжение между АСУП и АСУ ТП.

В настоящем пособии рассматривается только верхний уровень.

Прежде чем восстановить историю создания и внедрения автоматизированных систем управления предприятием (АСУП) в России (бывшем СССР), необходимо уточнить, о чем идет речь с точки зрения современных понятий об информационных технологиях. Когда-то тиражируемый термин АСУП сейчас почти не употребляется. Дело здесь в том, что смысл понятия «автоматизированная система управления», предполагавшего непереносимое присутствие человека как звена системы



Рис. 1.1 Блок-схема автоматизированной системы управления

управления был потерян. Многократное упоминание понятия АСУП привело к тому, что перестали замечать содержащееся в нем противоречие. В действительности именно функции управления предприятием никогда не были реализованы в АСУП, которые лишь предоставляли информацию лицам, принимавшим решения по управлению предприятием, но находившимся вне системы.

Работы по созданию АСУП в рамках направления, называвшегося «экономической кибернетикой», были начаты по инициативе академика В.М. Глушкова в Институте кибернетики АН СССР в 1963 – 1964 гг.

Первой в СССР системой для предприятий с крупносерийным характером производства была АСУП «Львов», внедренная на Львовском телевизионном заводе «Электрон».

Решение задачи, поставленной В.М. Глушковым, – создать не индивидуальную для данного предприятия, а типовую систему, привело к методам построения прикладных программ, использующих параметрическую настройку на особенности конкретного предприятия при привязке, наладке и внедрении типовой системы. Эти методы максимального использования параметров, а не числовых значений при построении прикладных программ, стали со временем широко распространенными и используются до сих пор в корпоративных информационных системах планирования ресурсов предприятия (ERP).

Глушковым В.М. было выдвинуто понятие специализированной операционной системы, предназначенной для систем с регулярным потоком задач. Универсальные операционные системы для решения случайных потоков задач в пакетном режиме в вычислительных центрах, например OS/360 фирмы IBM для семейства микропроцессоров 360, не позволяли использовать преимущества, которые могло представить знание регулярного потока задач в АСУП. Поэтому для программного обеспечения АСУП на базе отечественных ЭВМ второго поколения серий «Минск» и «Урал» были разработаны программные средства управления расписанием задач, предварительной подготовкой информации и мультипрограммными режимами исполнения прикладных программ. Хотя до штатных операционных систем ЭВМ «Минск» и «Урал» такие решения не были доведены.

В конце 1960-х – начале 1970-х гг. после завершения работ по АСУП «Львов» под руководством В.М. Глушкова была создана типовая АСУП «Кунцево», внедренная на Кунцевском радиозаводе.

Создание крупных АСУП потребовало использования и развития методов оптимизации. Работы в этой области проводились в Институте кибернетики АН УССР под руководством В.С. Михалевича. В 1960 – 1962 гг. была предложена общая алгоритмическая схема последовательного анализа вариантов, включавшая в себя, как частный случай, вычислительные методы динамического программирования. Эта схема была развита вместе с методами математического моделирования для решения задач упорядочения, в частности в теории расписаний и календарного планирования. Эти результаты послужили математической основой АСУП «Львов» и «Кунцево».

Новый этап в развитии АСУП пришелся на вторую половину 1970-х гг. и 1980-е гг. Это были комплексные системы, в которых интегрировались задачи автоматизированного проектирования новых изделий (САПР), технологической подготовки производства, автоматизации испытаний готовых изделий и автоматизации организационного управления предприятием. Техническую базу нового поколения АСУП составляли, как правило, модели ЕС ЭВМ, СМ ЭВМ.

До настоящего времени автоматизация предприятия велась по трем обособленным и независимым друг от друга направлениям: системы автоматизации управленческой и финансово-хозяйственной деятельности (АСУП), системы автоматизированного проектирования (САПР) и системы автоматизации технологических и производственных процессов (АСУ ТП). Данные системы проектировались и создавались исходя из требований разных подразделений предприятий, они не подчинялись единым целям и задачам, были плохо связаны физически и информационно, каждая система строилась по своим внутренним законам. Большим недостатком было еще то, что системы базировались на различных аппаратных, программных и информационных стандартах.

Все перечисленное выше не позволяло руководителям предприятий построить единую автоматизированную систему управления предприятием. Руководители предприятий оказались перед трудным выбором: с чего начинать автоматизацию – с АСУП, САПР или АСУ ТП, на какие стандарты ориентироваться? Определяющей тенденцией в развитии единой системы управления предприятием стало логическое и информационное взаимопроникновение различных уровней: бизнес-уровня (АСУП), уровня проектирования (САПР) и производственно-технологического уровня (АСУ ТП). Благодаря взаимопроникновению этих систем, предприятие становится единым организмом, который функционирует в едином информационном пространстве.

Только тогда, когда предприятие становится единым целым, возможно эффективное управление финансово-хозяйственной и производственной деятельностью предприятия. Кроме того, идет активное сближение стандартов и упрощается задача совместимости различных аппаратных и программных средств. Это позволяет избежать дополнительных затрат при объединении в одну систему оборудования различных производителей. В настоящее время стремительно развиваются Интернет-технологии, которые становятся неотъемлемой частью всей автоматизированной системы предприятия.

Все вышеперечисленное позволяет каждому участнику производственного процесса самостоятельно, без специалистов-посредников, с помощью специально разработанного интерфейса запрашивать и принимать необходимую информацию, осуществлять настройки различных режимов своего информационного обслуживания, что способствует повышению эффективности производства, но при этом возникает проблема информационной безопасности работы вычислительной системы, целостности хранимых данных. Особенно актуально решение данных вопросов в корпоративных системах, где взаимодействуют между собой огромное количество пользователей. Бизнес, который становится глобальным и распределенным, требует общего пространства для работы с информацией. Через Интернет обеспечивается доступ к корпоративным данным, в том числе и к жизненно важной информации предприятий. Если государственные организации, военные или дипломатические ведомства работают с закрытыми сетями, то у коммерческих организаций такой возможности нет: это невыгодно, неудобно и нецелесообразно.

В настоящее время вместо понятия АСУП используется более точное понятие – корпоративные информационные системы (КИС). Под ними понимают системы, в которых функционально объединяются системы для решения задач автоматизации учета и управления производством, финансами, снабжением и сбытом, кадрами и информационными ресурсами. Техническую базу современных КИС, использующих преимущественно распределенную архитектуру клиент-сервер, составляют серверы и рабочие места пользователей, объединенные локальными сетями.

Прежде чем перейти к рассмотрению следующих разделов пособия, введем ключевое понятие «информационная система».

### 1.1.1 Информационная система

Информационная система (ИС) – это вся инфраструктура предприятия (организации), задействованная в процессе управления всеми информационно-документальными потоками, включающая в себя следующие основные элементы:

- информационная модель, представляющая собой совокупность правил и алгоритмов функционирования ИС. Информационная модель включает в себя все формы документов, структуру справочников и данных и т.д.;
- регламент развития информационной модели и правил внесения в нее изменений;
- программный комплекс, конфигурация которого соответствует требованиям информационной модели;
- аппаратно-техническая база (компьютеры, периферийные устройства, каналы связи, системное ПО, СУБД).

Информационную систему можно представить в виде модели, состоящей из несколько взаимодействующих уровней иерархии (рис. 1.2).

В основании модели лежит слой различных типов компьютеров, являющихся средствами хранения и обработки данных. Компьютеры определяют аппаратную платформу информационной системы.

Транспортная система состоит из активных и пассивных сетевых устройств, объединяющих компьютеры в локальные и глобальные сети и обеспечивающих обмен данными. Активными сетевыми устройствами являются сетевые адаптеры и модемы компьютеров, концентраторы, коммутаторы, маршрутизаторы и другие подобные устройства. Среда передачи данных и элементы кабельной сети составляют пассивную часть транспортной системы.

Слой сетевых операционных систем обеспечивает выполнение приложений пользователей и посредством транспортной системы организует доступ к ресурсам других компьютеров и предоставляет свои ресурсы в общее пользование. Операционные системы компьютеров определяют программную платформу информационной системы. Ряд активных сетевых устройств, таких как коммутаторы и маршрутизаторы, как правило, работают под управлением собственных операционных систем, называемых операционными системами межсетевое взаимодействия.

<b>Приложения предметной области</b>
<b>Системные сервисы</b> интернет, электронная почта, системы управления базами данных, средства групповой работы
<b>Сетевые операционные системы</b>
<b>Транспортная система</b> локальных и глобальных сетей
<b>Компьютеры:</b> персональные, рабочие станции, серверы, мейнфреймы, кластеры

**Рис. 1.2 Иерархическая модель информационной системы**

Над слоем операционных систем работают слои различных приложений. Системные сервисы служат для обработки и преобразования информации, полученной от систем управления базами данных (СУБД) и других ресурсов, в вид удобный для восприятия конечным пользователем или прикладной программой. СУБД иногда выделяются в отдельный слой. Этим подчеркивается их высокая значимость как средства хранения в упорядоченном виде данных и выполнения базовых операций поиска и извлечения нужной информации.

Верхний слой информационной системы составляют приложения предметной области, специфические для конкретного предприятия (организации) или определенного типа предприятия. Это могут быть программные системы автоматизации бухгалтерского учета, проектирования, управления производством, агрегатами, технологическими процессами и другие.

Информационная система предприятия создается для работы прикладных программ. Именно эти программы обеспечивают сотрудников необходимой информацией для принятия решений и автоматизируют деятельность различных служб. Поэтому при проектировании информационной системы сначала определяются требования к этим программам, а уже затем – какие системные сервисы, базы данных, операционные системы, сетевые средства, компьютеры и серверы необходимы для их эффективного функционирования.

С точки зрения программных технологий, информационная система – это не один, и даже не несколько программных комплексов. Можно построить структурную модель информационной системы, выделив ее основные компоненты, которые содержат программные модули определенного класса (рис. 1.3).



Рис. 1.3 Структурная модель информационной системы

Самым нижним уровнем информационной системы является хранилище, в котором содержится вся интеллектуальная собственность предприятия. Это могут быть документы, справочники, структурные таблицы, деловые правила, описание процессов. Прямого доступа к хранилищу быть не должно, как для пользователей, так и для различных систем предприятия. Прямой доступ имеет лишь система управления знаниями, которая служит своего рода шлюзом для остальных систем и формирует информационное окружение предприятия. Система управления знаниями объединяет идеи, знания, содержание документов и деловые правила, автоматизируя процессы, базирующиеся на знаниях, как внутри предприятия, так и между разными организациями. Для этого нужен шлюз, позволяющий производить обмен данными с внешними системами. Это необходимое условие, так как современные процессы направлены на объединение предприятий в корпорации и очевидно, что передача знаний очень важна. Например, системы планирования ресурсов предприятия (ERP – Enterprise Resource Planning) не могут работать независимо – процессы, связанные с управлением финансами, складами, человеческими ресурсами, используют уже накопленные знания и приносят новые.

Также важно выделить класс систем анализа и принятия решений (DSS-decision support system), без которого жизненный цикл информации не будет завершен. В современных предприятиях интеллектуальный анализ данных становится все более важной задачей. Связано это с необходимостью аналитической обработки больших объемов информации, накопившейся в хранилищах. Такие системы помогают найти новые знания, выявить недостатки и слабые места информационной системы, оценить эффективность тех или иных процессов, установить новые информационные взаимосвязи.

Часто говорят, что такой класс систем должен работать непосредственно с хранилищем, поскольку обработке подлежат содержащиеся в нем данные. Теоретически это верно, но на практике такое невозможно – любые изменения в содержимом хранилища, процессах, правилах и взаимосвязях могут и должны производиться системой управления знаниями. Тогда DSS-системам не придется задумываться над тем, в каком формате хранятся данные, и главное, что любое изменение информации будет немедленно влиять на взаимосвязи и процессы, в которых она принимает участие.

Сформулируем минимальный перечень требований к КИС.

Функциональная полнота системы. Учитывая, что методологические подходы всех разработчиков программного обеспечения к структуризации предметной области и названию формируемых приложений различаются, общей характеристикой функциональной полноты корпоративной информационной системы является количество однократно учитываемых параметров деятельности предприятия. Считается, что для КИС значение этих параметров должно быть примерно следующим:

- количество учитываемых параметров 2 – 10 тысяч;
- количество таблиц баз данных 800 – 3000.

КИС должна обеспечивать не только формирование отчетов, но и ведение учета одновременно по российским и международным стандартам (ISA и GAAP).

Обязательным условием является локализация информационной системы: учет национального законодательства и системы расчетов; интерфейс и система помощи на национальном языке.

Система должна обеспечивать разграничение доступа к данным и функциям, предупреждать попытки несанкционированного доступа к информации.

КИС – система постоянно развивающаяся, как в силу влияния внешних факторов (например, постоянных изменений в законодательстве), так и из-за изменения бизнес-функций предприятия, поэтому необходимо наличие инструментальных средств адаптации и сопровождения системы:

- управление структурой и функциями бизнес-процессов;
- изменение информационного пространства (редактирование БД, модификация структуры, полей таблиц, связей, индексов и т.п.);
- модификация интерфейсов ввода, просмотра и корректировки информации;
- изменение организационного и функционального наполнения рабочего места пользователя;
- генерация произвольных отчетов, сложных хозяйственных операций и форм.

Учитывая важность хранимых в системе данных, следует обеспечить: авторизацию информации, регистрацию времени ввода и модификации данных, ведение протокола удалений данных,

Как правило, большинство предприятий, для которых разрабатываются КИС, уже имеют установленные автоматизированные системы: АСУ ТП, САПР и т.п. Важно обеспечить обмен данными между КИС и другими программными продуктами, функционирующими на предприятии.



Для пользователей КИС большое значение имеет возможность консолидации информации: на уровне предприятий – для объединения информации филиалов, дочерних компаний, предприятий, входящих в холдинг и т.п.; на уровне отдельных задач, на уровне временных периодов – для выполнения анализа изменения тех или иных показателей за период, превышающий отчетный.

Очевидно, что КИС – это сложная система и для обеспечения ее надежности требуются специальные средства анализа состояния системы в процессе эксплуатации:



**Рис. 1.4 Обобщенная структура ИТ предприятия:**

САПР (CAD/CAM) – система автоматизированного проектирования-изготовления; АС ТПП (CAE) – автоматизированная система технологической подготовки производства; АСУ ТП (SCADA) – автоматизированные системы управления технологическими процессами (от эффективности которых зависит эффективность производства); КИС (MRP, ERP) – корпоративные информационные системы; ERP II – расширение ERP-системы за контуры производства (т.е. ERP +: CRM, B2B, DSS, SCM, PLM и др.); CRM – управление отношениями с клиентами (WF – частный случай CRM); B2B – электронная торговая площадка («онлайн-бизнес»); DSS – поддержка принятия управленческих решений; SPSS – статистический анализ данных; OLAP – анализ многомерных данных; MIS – автоматизированное рабочее место (АРМ) руководителя; SCM – управление цепочками поставок; PLM – управление жизненным циклом продукции; HR – управление персоналом, можно рассматривать как самостоятельную задачу, так и входящую в состав ERP; WF – электронный документооборот (электронная почта); УП – управление производством; ПУ – первичный учет; ПФ – планирование и бюджетирование; КУ – коммерческий учет; СУ – складской учет; БУ – бухгалтерский учет; СТ – системы телекоммуникаций

- анализ архитектуры баз данных;
- анализ алгоритмов;
- анализ статистики: количество записей, документов, проводов, объем дисковой памяти;
- журнал выполненных операций;
- список работающих станций, внутрисистемная почта.

Некоторые специалисты рассматривают вложение средств в создание КИС как долговременные инвестиции, при этом большое значение приобретают уровень и качество обслуживания, предоставляемого разработчиком. Для заказчика оптимальной является ситуация, когда он, обратившись к одному поставщику, получает весь спектр услуг:

- постановка системы управления предприятием;
- консалтинг;
- решение вопросов постановки учета и документооборота;
- обучение персонала заказчика;
- внедрение КИС в опытную и промышленную эксплуатацию;
- сопровождение системы на протяжении всего ее жизненного цикла;
- проведение тематических семинаров как по проблемам методологии и организации учета, так и по вопросам использования КИС.

Из приведенного перечня требований видно, что создание корпоративной информационной системы – задача очень сложная, требующая немалых затрат.

В заключении данного раздела приведем обобщенную структуру информационных технологий предприятия, которая представлена на рис. 1.4.

## 1.2 ИНФОРМАЦИОННОЕ ОБСЛЕДОВАНИЕ ПРЕДПРИЯТИЯ

Прежде чем попытаться выбрать существующие или создать собственную ИС, а затем ее внедрить ее, необходимо проанализировать, как работает предприятие (организация) в настоящее время. Для анализа необходимо знать не только как работает предприятие в целом и как оно взаимодействует с внешними организациями, заказчиками и поставщиками, но и как организована деятельность на каждом рабочем месте.

Отметим, что в современных ИС наблюдается сильная зависимость возможности приложения тех или иных методов управления предприятием от наличия адекватных средств их информационной поддержки. Освоив средства накопления и обработки информации в определенной области (Бухгалтерия 1С, SCADA-системы и т.д.), предприятие получает в свое распоряжение новый вид ресурсов – «информационных». Качество (ценность) этих ресурсов напрямую зависит от качества «организации» информации, позволяющей эффективно извлекать и представлять нужные сведения (данные), а также осуществлять различные преобразования над ними. Важно отметить, что в существующих ИС, поддерживающих управление предприятием, в основном сосредоточена оперативная количественная информация о материальных, финансовых и человеческих ресурсах (учет кадров). Гораздо меньше внимания уделяется информации о самой организации, деятельности предприятия и знаниях, лежащих в ее основе (целях, стратегии, процессах, функциях, бизнес-правилах и т.п.).

Исходя из вышесказанного, в основе построения современной ИС управления предприятием должна лежать модель предприятия, которая на различных уровнях структурирует и формализует знание о предприятии и его окружении.

Модель предприятия можно представить в виде схемы на рис. 1.5.

В самом верху диаграммы располагается объект «Миссия», описывающий такие вещи, как создание самого предприятия или выпуск нового типа продукции или модернизацию производственного цикла или еще что-то новое, привносимое в деятельность предприятия. Миссия порождает «Анализ» ситуации.

Анализ подразумевает проведение анализа рынка, спроса на товары и услуги, деятельности конкурентов и пр. Результатом анализа является постановка новых целей предприятия – объект «Цели». Для выполнения целей предприятия намечаются «Задачи», которые необходимо решить для достижения целей. Задачи, решаемые предприятием, определяют «Структуру организации». Новые же задачи могут привести к появлению на предприятии новых «Проектов» по разработке новых моделей изделий, новых типов услуг и т.д. Структура организации предполагает создание подразделений и филиалов предприятия. В каждом подразделении принимаются на работу сотрудники – «Исполнители». Результатами выполнения проектов являются также «Стандарты» предприятия и «Технические условия» – описания «Процессов» производства новых изделий на предприятии. Сами процессы состоят из последовательности «Операций». При выполнении каждой операции могут порождаться документы о результатах выполнения операций. Задачами на предприятии являются также «Планы» по выпуску продукции. Эти планы доводятся до исполнителей в виде «Заданий». На основании заданий исполнители выполняют производственные «Процессы».

Для обеспечения выполняемых производственных процессов необходимы материальные ресурсы – сырье, электроэнергия, комплектующие изделия и прочая «Продукция», получаемая от «Поставщиков». Но до ее использования в процессах нужно выполнить входной «Контроль качества» полученной при «Поставке» продукции, а для

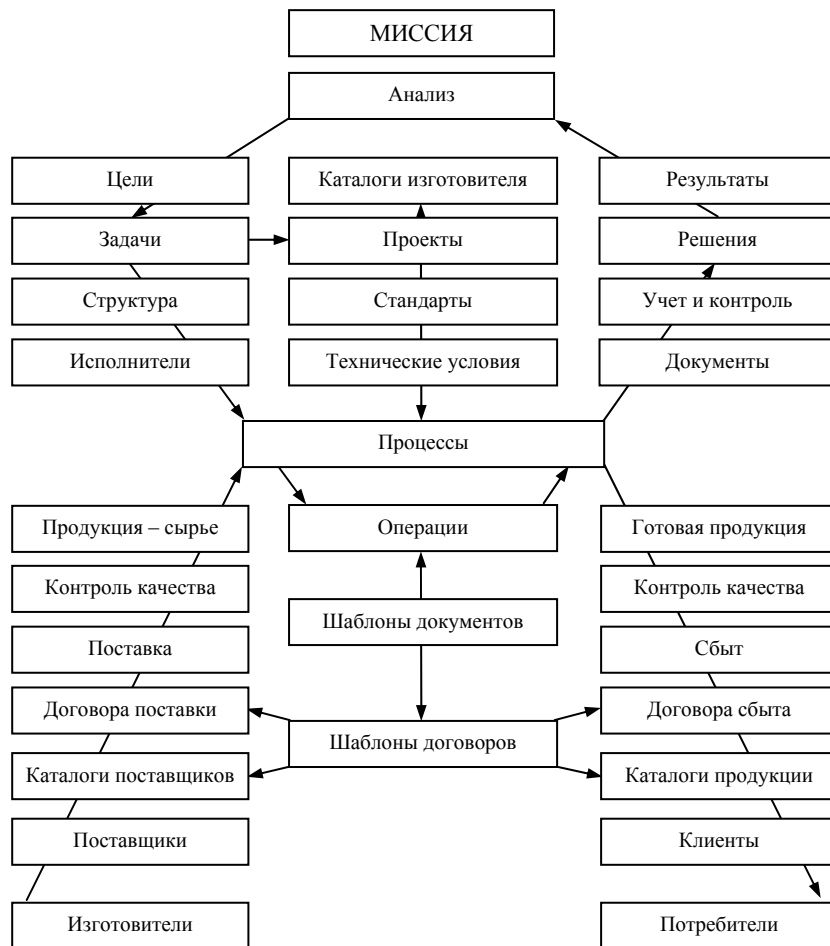


Рис. 1.5 Модель предприятия

выполнения поставки необходимо заключить «Договор с поставщиком». Перед заключением договора следует посмотреть характеристики и цены предлагаемых к поставке комплектующих в «Каталогах поставщика».

Результатами производственных процессов будут готовая продукция и документы, описывающие процесс производства (например, документы на каждое изделие, таблицы учета рабочего времени, акты о списании комплектующих изделий и пр.). Этот поток документов, порождаемый процессами, необходимо учитывать и контролировать – контролировать ход выполнения заданий исполнителей и планов структурных подразделений. По результатам контроля определяются достигнутые «Решения» в ходе выполнения поставленных перед предприятием задач. По этим решениям определяются «Результаты» достижения целей предприятия. На основе полученных результатов необходимо снова провести анализ деятельности предприятия, инициировав, возможно, при этом новые миссии. Далее вновь по кругу – намечаем новые цели.

Произведенная «Готовая продукция» должна пройти выходной «Контроль качества» и далее после хранения на «Складе готовой продукции», если это необходимо, должен быть «Сбыт» потребителям. Для сбыта продукции выпускаются «Каталоги продукции» (это такие же каталоги поставщика, которые предприятие запрашивало от своих поставщиков, но на этот раз само предприятие выступает в роли поставщика и предлагает свои каталоги для потенциальных клиентов), в которых указываются модели изделий и типы услуг, количество сбываемых изделий и их цены. Для каждого раздела каталога предполагаются стандартные «Договора сбыта», которые заключаются с «Клиентами» на основе «Шаблонов договоров». Шаблоны договоров являются частным случаем «Шаблоном документов», используемых в различных производственных операциях, а сами процессы приобретения комплектующих и сбыта готовой продукции – частными случаями производственных процессов. Производственным процессом является также процесс разработки нового проекта, т.е. имеет смысл говорить об основной информационной петле на предприятии (Анализ – Цели – Задачи – Структура – Исполнители – Процессы – Документы – Учет и контроль – Решения – Результаты – Анализ) и о библиотеке процессов: закупка комплектующих, сбыт готовой продукции, разработка новых проектов и пр.

Помимо вышеназванных объектов, вне организации имеются также «Изготовители» комплектующих изделий и услуг, используемых при производстве, и «Потребители», которые потенциально могут стать клиентами предприятия.

Образно приведенную выше диаграмму можно рассматривать как кольцо, по которому непрерывно крутится информация. На стадии производственного процесса она заглатывает сырье, комплектующие изделия, энергию, в результате чего появляются новые готовые изделия. Поставщики и клиенты являются «открытыми интерфейсами», связывающими предприятие с другими такими же предприятиями. Результатом таких связей

является обмен продукцией (в общем случае под продукцией будем понимать – товар, услуги, документы). Это своего рода живые организмы, взаимодействующие между собой.

### 1.3 РЕИНЖИНИРИНГ И БИЗНЕС-ПРОЦЕССЫ

В связи с развитием и широким внедрением коммуникационных и сетевых компьютерных технологий изменились и принципы построения информационных систем. Результатом такого развития стало появление корпоративных информационных систем (КИС). Такие системы представляют целый комплекс программно-аппаратных средств, который позволяет автоматизировать бизнес-процессы и информационные потоки на предприятии или организации с целью адекватного информационного обеспечения для повышения эффективности процесса управления.

Первые классические модели создания КИС опирались, как правило, на комплексную автоматизацию ключевых производственных, административно-управленческих и организационных функций предприятий или организаций путем использования соответствующей компьютерной инфраструктуры и программного обеспечения. Со временем стало ясно, что такое сужение функций КИС не соответствует возможностям системы. Особенность создания и внедрения КИС состоит не столько в автоматизации функциональных подсистем предприятия (организации), сколько в решении задач оптимизации бизнес-процессов и совершенствования процессной организации управления.

С начала 1990-х гг. методическим направлением, изучающим вопросы процессной организации систем управления предприятиями и дающим решения по их построению, является реинжиниринг бизнес-процессов РБП (BPR – Business Process Reengineering).

Автоматизация управления предприятием базируется на реинжиниринге бизнес-процессов. Те процессы, которые на «западе» мыслятся под термином реинжиниринг, в России традиционно подразумеваются под тремя понятиями:

- организация производства;
- организация труда;
- организация управления.

Реинжиниринг бизнес-процессов является направлением, возникшим на стыке двух различных сфер деятельности – управления (менеджмента) и информатизации. Именно поэтому реинжиниринг требует новых специфических средств представления и обработки проблемной информации, понятных как менеджерам, так и разработчикам информационных систем управления. Подобные средства требуют интеграции ключевых достижений информационных технологий и создания соответствующих инструментальных средств поддержки реинжиниринга.

Впервые термин «реинжиниринг бизнес-процессов» был введен М. Хаммером в 1990 г. в статье «Реинжиниринг: не автоматизируйте – уничтожайте», который определяет этот вид деятельности как «фундаментальное перепроектирование бизнес-процессов предприятий для достижения коренных улучшений в основных актуальных показателях их деятельности: стоимость, качество, услуги и темпы». За несколько лет реинжиниринг превратился в одну из ведущих и активно развивающихся отраслей информатики.

Сегодня начинается продвижение консалтинговых услуг и инструментариев по реинжинирингу и на российский рынок. Отечественная практика применения информационных технологий управления показала, что этот метод необходим, особенно в условиях проведения глобальной экономической реформы и активного внедрения России в мировую экономическую систему.

Хаммер М. рассматривает BPR как революцию в бизнесе, которая знаменует отход от базовых принципов построения предприятий и превращает конструирование бизнеса в инженерную деятельность. Возможность такой революции обусловлена, в первую очередь, новейшими достижениями в области информационных технологий, специалисты которой начинают играть ведущую роль в конструировании бизнеса.

Именно информации, информационным потокам, отражающим все производственные и управленческие процессы на предприятии, способам ее обработки и анализа уделяется особое внимание.

Одной из основных особенностей BPR является ориентация реинжиниринга не на функции, а на процессы. Причем из всех концепций менеджмента, основанных на процессах, BPR рассматривается как наиболее эффективная, революционность которой обусловлена современным состоянием информационных технологий.

Существуют и другие методы, которые можно рассматривать либо как частные случаи BPR (если в основе этих методов лежит управление процессами), либо как автономные концепции (если они базируются на иных принципах):

- автоматизация бизнес-процессов (business process automation – BPA). Автоматизация приводит лишь к ускорению существующих бизнес-процессов. Используя информационные технологии, BPA автоматизирует существующий процесс со всеми его недостатками и не ставит перед собой задачу проектирования нового процесса для кардинального повышения эффективности;
- реинжиниринг программного обеспечения. На основе современных технологий производит переписывание устаревших информационных систем без изменения самих автоматизируемых процессов;
- реорганизация (reorganizing) предприятия. Данная концепция имеет дело только с организационными структурами, а не с процессами;
- улучшение качества (quality improvement – QI), глобальное управление качеством (total quality management – TQM). Хотя управление качеством отводит центральную роль бизнес-процессам, данный метод принимает имеющиеся процессы и старается их улучшить, не меняя их на новые.

Реинжиниринг основан на концепции прерывистого мышления в отыскании устаревших правил и фундаментальных допущений, на которых строилась работа, и решительном разрыве с ними. Нельзя достичь кардинального повышения производительности просто автоматизацией существующего процесса. Скорее, следует проверить обоснованность существующих допущений и отказаться от старых правил, которые, собственно, и вызывают недостаточную производительность.

Базовыми понятиями BPR являются бизнес-процесс, бизнес-система, деловая процедура:

*Бизнес-система* – это связанное множество бизнес-процессов, конечной целью которого является выпуск продукции. Под продукцией понимают товары, услуги и документы.

*Бизнес-процесс* – это горизонтальная иерархия внутренних и зависимых между собой функциональных действий, конечной целью которых является выпуск продукции или отдельных ее компонентов.

*Деловая процедура* – это функция, задача, цепь событий, происходящих в течение определенного промежутка времени и обладающих познаваемым результатом.

Существуют следующие категории бизнес-процессов:

- процессы, непосредственно обеспечивающие выпуск продукции;
- процессы планирования и управления;
- ресурсные процессы;
- процессы преобразования.

Бизнес-процесс характеризуется:

- существующей технологией реализации бизнес-процесса;
- существующей структурой бизнес-системы;
- средствами автоматизации, оборудованием, механизмами и т.п., обеспечивающими реализацию процесса.

Основными показателями оценки эффективности бизнес-процессов являются:

- количество производимой продукции заданного качества, оплаченное за определенный интервал времени;
- количество потребителей продукции;
- количество типовых операций, которые необходимо выполнить при производстве продукции за определенный интервал времени;
- стоимость издержек производства продукции;
- длительность выполнения типовых операций;
- капиталовложения в производство продукции.

Реинжиниринг бизнес-процессов базируется на следующих основных принципах:

- несколько рабочих процедур объединяются в одну, т.е. происходит горизонтальное сжатие процесса (по имеющимся оценкам, горизонтальное сжатие ускоряет выполнение процесса примерно в 10 раз);
- исполнители принимают самостоятельные решения, т.е. осуществляется не только горизонтальное, но и вертикальное сжатие процессов (наделение сотрудников большими полномочиями и увеличение роли каждого из них приводит к значительному повышению их отдачи);
- процессы имеют различные варианты исполнения (тот или иной вариант выбирается в зависимости от конкретной ситуации, состояния и т.д.);
- работа выполняется в том месте (подразделении, отделе), где это целесообразно (устраняется излишняя интеграция, что приводит к повышению эффективности процесса в целом);
- уменьшается количество проверок и управляющих воздействий;
- минимизируется количество согласований путем сокращения внешних точек контакта;
- единая точка контакта обеспечивается уполномоченным менеджером (в тех случаях, когда шаги процесса либо сложны, либо распределены таким образом, что их не удастся объединить силами небольшой команды).

Успешное использование принципа «непрерывного улучшения» бизнес-процессов (BPI) основывается на пересечении трех областей знаний (рис. 1.6).

*Область А* – развитие информационных технологий:

- использование профессиональных операционных систем (для серверов баз данных) и персональных компьютеров;
- использование профессиональных систем управления базами данных (СУБД);



**Рис. 1.6 Области знаний**

- использование ERP-систем как ядра интегрированной ИС предприятия;
- использование кооперативных технологий, обеспечивающих компьютерную поддержку параллельной согласованной работы группы («команды») сотрудников над одним проектом, документом и т.п.;
- использование телекоммуникации, позволяющей исключить передачу бумажных документов и личных встреч, свести к минимуму необходимость переездов для проведения совещаний;
- использование систем управления знаниями для организации хранилища и поиска неструктурированных документов.

*Область В* – развитие бизнес-платформ, включающих:

- методики управления качеством (т.е. целостную идеологию управления предприятием) на базе стандартов ИСО серии 9000;
- методики организации операционного менеджмента (ERP-стандарты);
- методики управления требованиями и конструкторскими разработками (CALS-стандарты);
- методики моделирования бизнес-процессов (SADT, IDEF0, DFD, UML).

*Область С* определяет «психологию труда» и направлена на решение следующих задач:

- внедрение принципа «лидерства» (устранение недостатков производственной системы, а не отдельных работников);
- внедрение принципа «вовлеченности работников» (повышение значимости и инициативности каждого работника);
- снятие барьеров между производственными подразделениями, организация групповой «артериальной работы»; образование так называемых «плоских» рабочих групп, использующих эдхократические («эдхократия» – компетентная бюрократия) способы управления, опирающиеся на информационные технологии и организуемые динамическое и неформальное распределение прав и обязанностей сотрудников группы;
- формирование корпоративной культуры и повышение эдхократии в организации;
- внедрение философии тотального управления качеством на всех рабочих местах (TQM);
- внедрение философии организации производственных процессов «точно вовремя» на всех рабочих местах.

Отметим, что реинжиниринг не является модной тенденцией, это следствие жестокой конкурентной борьбы, выдержать которую можно только путем внедрения новых, наукоемких инновационных технологий. Большинство предприятий (организаций), проводивших реинжиниринг своего бизнеса, были просто вынуждены это сделать, оказавшись перед лицом кризиса.

#### **1.4 СТАНДАРТЫ ОПИСАНИЯ, АНАЛИЗА И РЕОРГАНИЗАЦИИ БИЗНЕС-ПРОЦЕССОВ**

Мировой опыт свидетельствует, что внедрению на предприятии ИС должно предшествовать серьезное функционально-информационное обследование предприятия, с целью определения оптимальности бизнес-процессов, распределения ресурсов между функциями и т.д.

Понятие «моделирование бизнес-процессов» пришло в быт большинства аналитиков одновременно с появлением на рынке сложных программных продуктов, предназначенных для систем комплексной автоматизации управления предприятием. Подобные системы всегда подразумевают проведение глубокого предпроектного обследования деятельности. Результатом этого обследования является экспертное заключение, в котором отдельными пунктами выносятся рекомендации по устранению «узких мест» в управлении деятельностью. На основании данного заключения, непосредственно перед проектом внедрения системы автоматизации, проводится так называемая реорганизация бизнес-процессов, иногда достаточно серьезная и болезненная для компании. Это и естественно, сложившийся годами коллектив всегда сложно заставить «думать по-новому». Подобные комплексные обследования предприятий являются сложными и существенно отличающимися от случая к случаю задачами. Для решения подобных задач моделирования сложных систем существуют хорошо обкатанные методологии и стандарты. К таким стандартам относятся методологии семейства IDEF. С их помощью можно эффективно отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах. При этом широта и глубина обследования процессов в системе определяется самим разработчиком, что позволяет не перегружать создаваемую модель излишними данными. В настоящий момент к семейству IDEF можно отнести следующие стандарты:

1) IDEF0 – методология функционального моделирования (Руководящий документ Госстандарта РФ «Методология функционального моделирования IDEF0»). Метод IDEF0 предназначен для функционального моделирования, т.е. моделирования выполнения функций объекта путем создания описательной графической модели, показывающей что, как и кем делается в рамках функционирования предприятия. Функциональная модель представляет собой структурированное изображение функций производственной системы или среды, информации и объектов, связывающих эти функции;

2) IDEF1 – методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи;

3) IDEF1X (IDEF1 Extended) – методология построения реляционных структур. IDEF1X относится к типу методологий «Сущность-взаимосвязь» (ER – Entity-Relationship) и, как правило, используется для моделирования реляционных баз данных, имеющих отношение к рассматриваемой системе;

4) IDEF2 – методология динамического моделирования развития систем. В связи с весьма серьезными сложностями анализа динамических систем от этого стандарта практически отказались, и его развитие приостановилось на самом начальном этапе. Однако в настоящее время применяются алгоритмы и их компьютерные

реализации, позволяющие превращать набор статических диаграмм IDEF0 в динамические модели, построенные на базе «раскрашенных сетей Петри» (CPN – Color Petri Nets);

5) IDEF3 – методология документирования процессов, происходящих в системе, которая используется, например, при исследовании технологических процессов на предприятиях. С помощью IDEF3 описываются сценарий и последовательность операций для каждого процесса. IDEF3 имеет прямую взаимосвязь с методологией IDEF0 – каждая функция (функциональный блок) может быть представлена в виде отдельного процесса средствами IDEF3;

6) IDEF4 – методология построения объектно-ориентированных систем. Средства IDEF4 наглядно отображают структуру объектов и заложенные принципы их взаимодействия, тем самым позволяя анализировать и оптимизировать сложные объектно-ориентированные системы;

7) IDEF5 – методология онтологического исследования сложных систем. С помощью методологии IDEF5 онтология системы может быть описана при помощи определенного словаря терминов и правил, на основании которых могут быть сформированы достоверные утверждения о состоянии рассматриваемой системы в некоторый момент времени. На основе этих утверждений формируются выводы о дальнейшем развитии системы и производится ее оптимизация.

Рассмотрим наиболее часто используемую методологию функционального моделирования IDEF0. Методологию IDEF0 можно считать следующим этапом развития хорошо известного графического языка описания функциональных систем SADT (Structured Analysis and Design Technique).

#### 1.4.1 Методология функционального моделирования SADT

Построение SADT-модели начинается с представления всей системы в виде простейшей компоненты – одного блока и дуг, изображающих интерфейсы с функциями вне системы. Поскольку единственный блок представляет 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.

Результатом применения методологии SADT является модель, которая состоит из диаграмм, фрагментов текстов и глоссария, имеющих ссылки друг на друга. Диаграммы – главные компоненты модели, все функции ИС и интерфейсы на них представлены как блоки и дуги. Место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, в то время как информация, которая подвергается обработке, показана с левой стороны блока, а результаты выхода показаны с правой стороны. Механизм (человек или автоматизированная система), который осуществляет операцию, представляется дугой, входящей в блок снизу (рис. 1.7).

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

На рис. 1.8, где приведены четыре диаграммы и их взаимосвязи, показана структура SADT-модели. Каждый компонент модели может быть декомпозирован на другой диаграмме. Каждая диаграмма иллюстрирует «внутреннее строение» блока на родительской диаграмме.

Затем блок, который представляет систему в качестве единого модуля, детализируется на другой диаграмме с помощью нескольких блоков, соединенных интерфейсными дугами. Эти блоки представляют основные подфункции исходной функции. Данная декомпозиция выявляет полный набор подфункций, каждая из которых представлена как блок, границы которого определены интерфейсными дугами. Каждая из этих подфункций может быть декомпозирована подобным образом для более детального представления.

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.

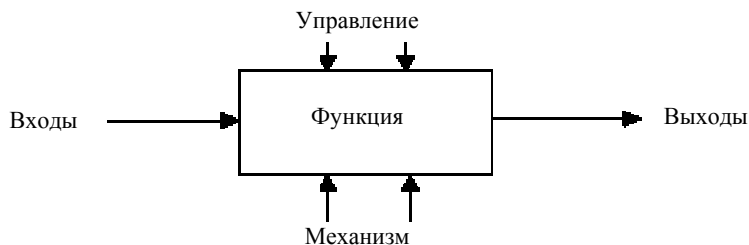
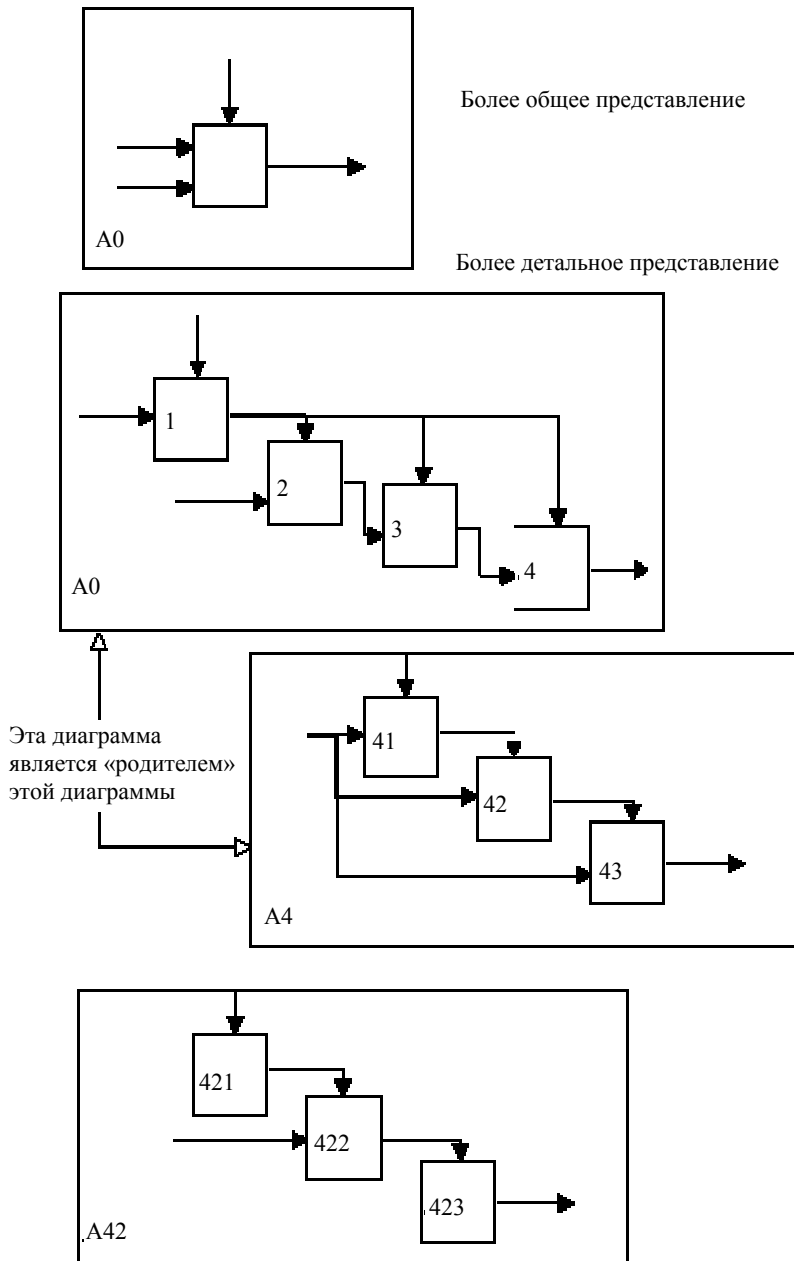


Рис. 1.7 Функциональный блок и интерфейсные дуги

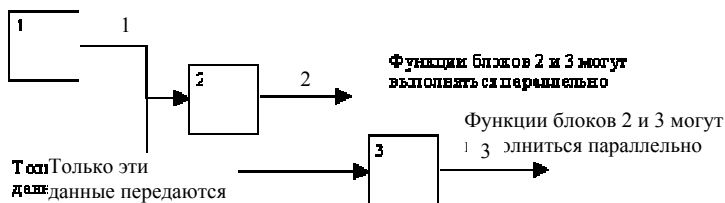


**Рис. 1.8 Структура SADT-модели. Декомпозиция диаграмм**

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

На рис. 1.9 – 1.11 представлены различные варианты выполнения функций и соединения дуг с блоками.



**Рис. 1.9 Одновременное выполнение**



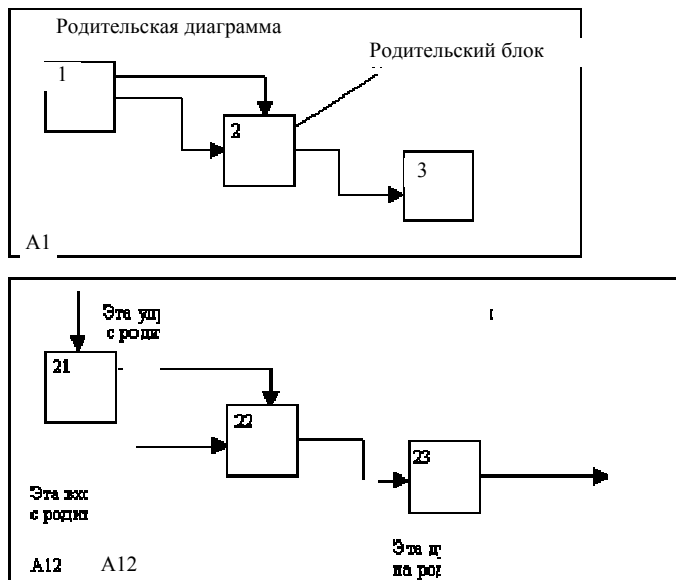


Рис. 1.10 Пример детализации

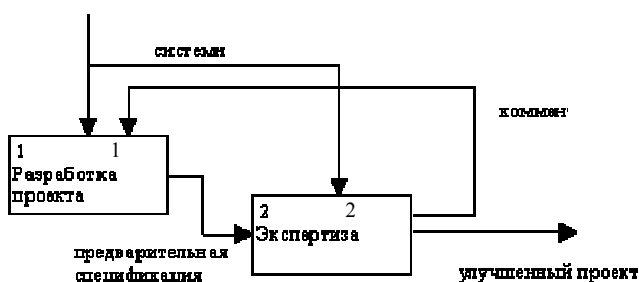


Рис. 1.11 Пример обратной связи

Некоторые дуги присоединены к блокам диаграммы обоими концами, у других же один конец остается не присоединенным. Не присоединенные дуги соответствуют входам, управлениям и выходам родительского блока. Источник или получатель этих пограничных дуг может быть обнаружен только на родительской диаграмме. Не присоединенные концы должны соответствовать дугам на исходной диаграмме. Все граничные дуги должны продолжаться на родительской диаграмме, чтобы она была полной и непротиворечивой.

На SADT-диаграммах не указаны явно ни последовательность, ни время. Обратные связи, итерации, продолжающиеся процессы и перекрывающиеся (по времени) функции могут быть изображены с помощью дуг. Обратные связи могут выступать в виде комментариев, замечаний, исправлений и т.д. (рис. 1.11).

Как было отмечено, механизмы (дуги с нижней стороны) показывают средства, с помощью которых осуществляется выполнение функций. Механизм может быть человеком, компьютером или любым другим устройством, которое помогает выполнять данную функцию (рис. 1.12).

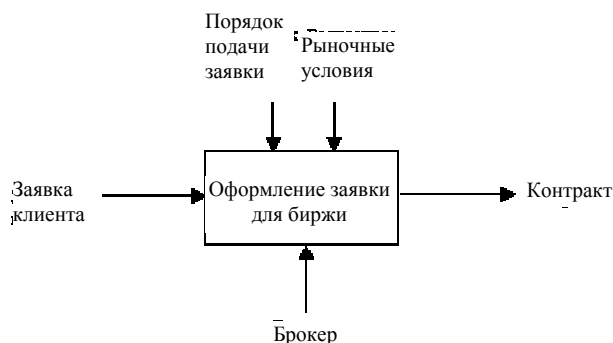


Рис. 1.12 Пример механизма

Каждый блок на диаграмме имеет свой номер. Блок любой диаграммы может быть далее описан диаграммой нижнего уровня, которая, в свою очередь, может быть далее детализирована с помощью необходимого числа диаграмм. Таким образом, формируется иерархия диаграмм.

Для того, чтобы указать положение любой диаграммы или блока в иерархии, используются номера диаграмм. Например, A21 является диаграммой, которая детализирует блок 1 на диаграмме A2. Аналогично, A2 детализирует блок 2 на диаграмме A0, которая является самой верхней диаграммой модели. На рис. 1.13 показано типичное дерево диаграмм.

Одним из важных моментов при проектировании ИС с помощью методологии SADT является точная согласованность типов связей между функциями. Различают по крайней мере шесть типов связывания.

Ниже каждый тип связи кратко определен и проиллюстрирован с помощью типичного примера из SADT.

(0) *Тип случайной связности*: наименее желательный.

Случайная связность возникает, когда конкретная связь между функциями мала или полностью отсутствует. Это относится к ситуации, когда имена данных на SADT-дугах в одной диаграмме имеют малую связь друг с другом. Крайний вариант этого случая показан на рис. 1.14.

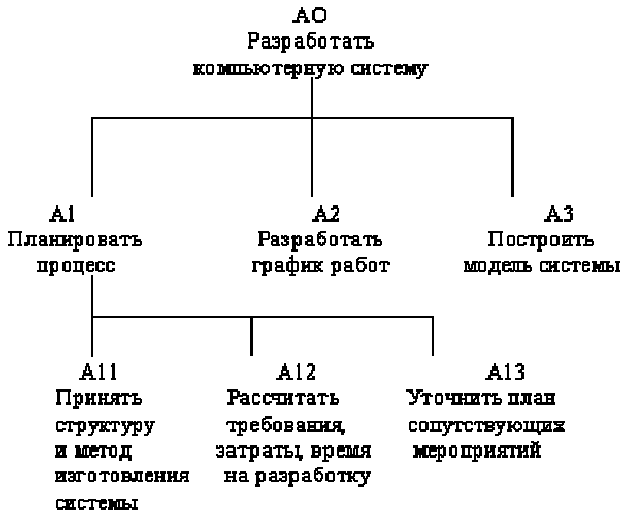


Рис. 1.13 Иерархия диаграмм

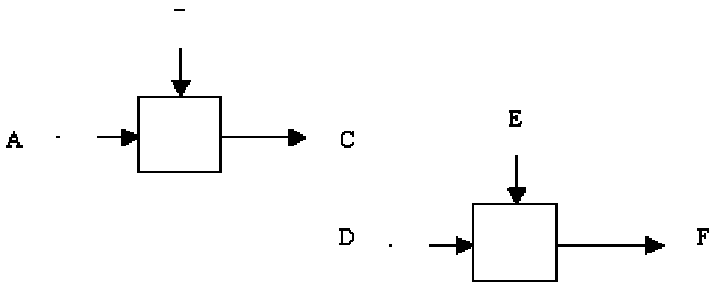


Рис. 1.14 Случайная связность

(1) *Тип логической связности*. Логическое связывание происходит тогда, когда данные и функции собираются вместе вследствие того, что они попадают в общий класс или набор элементов, но необходимых функциональных отношений между ними не обнаруживается.

(2) *Тип временной связности*. Связанные по времени элементы возникают вследствие того, что они представляют функции, связанные во времени, когда данные используются одновременно или функции включаются параллельно, а не последовательно.

(3) *Тип процедурной связности*. Процедурно-связанные элементы появляются сгруппированными вместе вследствие того, что они выполняются в течение одной и той же части цикла или процесса. Пример процедурно-связанной диаграммы приведен на рис. 1.15.

(4) *Тип коммуникационной связности*. Диаграммы демонстрируют коммуникационные связи, когда блоки группируются вследствие того, что они используют одни и те же входные данные и/или производят одни и те же выходные данные (рис. 1.16).

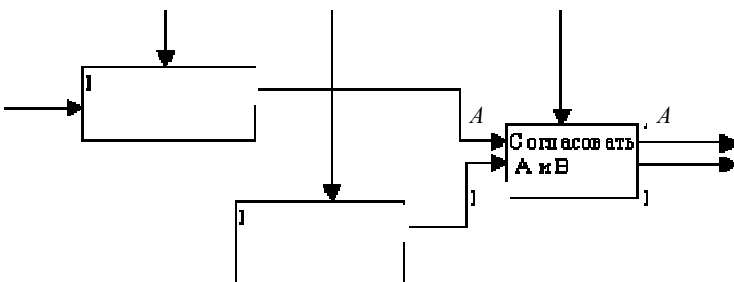


Рис. 1.15 Процедурная связность

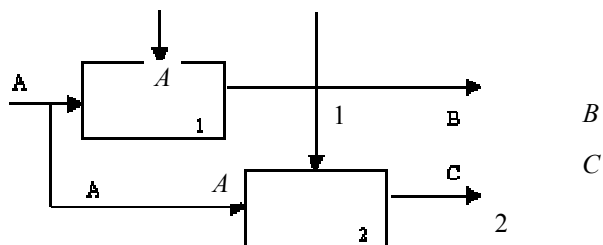


Рис. 1.16 Коммуникационная связность

(5) *Тип последовательной связности.* На диаграммах, имеющих последовательные связи, выход одной функции служит входными данными для следующей функции. Связь между элементами на диаграмме является более тесной, чем на рассмотренных выше уровнях связей, поскольку моделируются причинно-следственные зависимости (рис. 1.17).

(6) *Тип функциональной связности.* Диаграмма отражает полную функциональную связность, при наличии полной зависимости одной функции от другой. Диаграмма, которая является чисто функциональной, не содержит чужеродных элементов, относящихся к последовательному или более слабому типу связности. Одним из способов определения функционально-связанных диаграмм является рассмотрение двух блоков, связанных через управляющие дуги, как показано на рис. 1.18.

Ниже в табл. 1.1 представлены все типы связей, рассмотренные выше. Важно отметить, что уровни 4 – 6 устанавливают типы связностей, которые разработчики считают важнейшими для получения диаграмм хорошего качества.

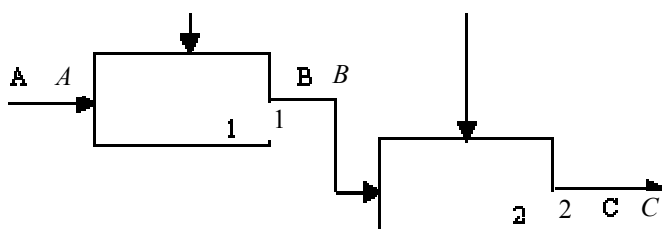


Рис. 1.17 Последовательная связность

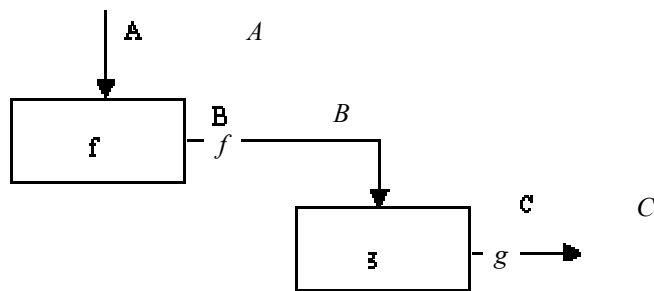


Рис. 1.18 Функциональная связность

### 1.1 Типы связей

Значимость	Тип связности	Для функций	Для данных
0	Случайная	Случайная	Случайная
1	Логическая	Функции одного и того же множества или типа (например, «редактировать все входы»)	Данные одного и того же множества или типа
2	Временная	Функции одного и того же периода времени (напри-	Данные, используемые в каком-либо временном

		мер, «операции инициализации»)	интервале
3	Процедурная	Функции, работающие в одной и той же фазе или итерации (например, «первый проход компилятора»)	Данные, используемые во время одной и той же фазы или итерации
4	Коммуникационная	Функции, использующие одни и те же данные	Данные, на которые воздействует одна и та же деятельность

Продолжение табл. 1.1

Значимость	Тип связности	Для функций	Для данных
5	Последовательная	Функции, выполняющие последовательные преобразования одних и тех же данных	Данные, преобразуемые последовательными функциями
6	Функциональная	Функции, объединяемые для выполнения одной функции	Данные, связанные с одной функцией

#### 1.4.2 Стандарт IDEF0

Исторически IDEF0 как стандарт был разработан в 1981 г. в рамках обширной программы автоматизации промышленных предприятий, которая носила обозначение ICAM (Integrated Computer Aided Manufacturing) и была предложена департаментом ВВС США. Собственно семейство стандартов IDEF унаследовало свое обозначение от названия этой программы (IDEF = ICAM DEFinition). В процессе практической реализации участницы программы ICAM столкнулись с необходимостью разработки новых методов анализа процессов взаимодействия в промышленных системах. При этом, кроме усовершенствованного набора функций для описания бизнес-процессов, одним из требований к новому стандарту было наличие эффективной методологии взаимодействия в рамках «аналитик-специалист». Другими словами, новый метод должен был обеспечить групповую работу над созданием модели с непосредственным участием всех аналитиков и специалистов, занятых в рамках проекта.

С 1981 г. стандарт IDEF0 претерпел несколько незначительных изменений, в основном ограничивающего характера, и последняя его редакция была выпущена в декабре 1993 г. Национальным Институтом по Стандартам и Технологичам США (NIST).

В основе методологии лежат четыре основных понятия. Первым из них является понятие *функционального блока* (Activity Box). Функциональный блок графически изображается в виде прямоугольника (рис. 1.19) и олицетворяет собой некоторую конкретную функцию в рамках рассматриваемой системы. По требованиям стандарта название каждого функционального блока должно быть сформулировано в глагольной форме (например, «производить услуги», а не «производство услуг»).

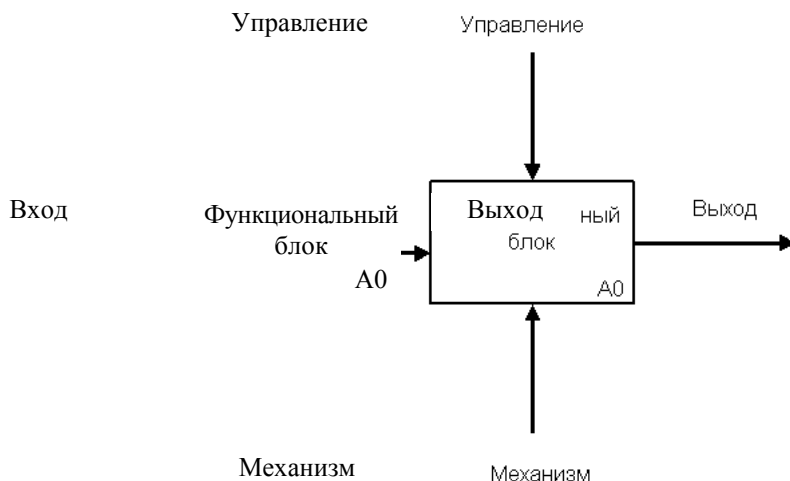


Рис. 1.19 Функциональный блок

Каждая из четырех сторон функционального блока имеет свое определенное значение (роль), при этом:

- верхняя сторона имеет значение «Управление» (Control);
- левая сторона имеет значение «Вход» (Input);
- правая сторона имеет значение «Выход» (Output);
- нижняя сторона имеет значение «Механизм» (Mechanism).

Каждый функциональный блок в рамках единой рассматриваемой системы должен иметь свой уникальный идентификационный номер.

Вторым понятием методологии IDEF0 является понятие *интерфейсной дуги* (Arrow). Интерфейсные дуги часто называют потоками или стрелками. Интерфейсная дуга отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, отображенную данным функциональным блоком.

Графическим отображением интерфейсной дуги является однонаправленная стрелка. Каждая интерфейсная дуга должна иметь свое уникальное наименование (Arrow Label). По требованию стандарта, наименование должно быть оборотом существительного.

С помощью интерфейсных дуг отображают различные объекты, в той или иной степени определяющие процессы, происходящие в системе. Такими объектами могут быть элементы реального мира (детали, вагоны, сотрудники и т.д.) или потоки данных и информации (документы, данные, инструкции и т.д.).

В зависимости от того, к какой из сторон подходит данная интерфейсная дуга, она носит название «входящей», «исходящей» или «управляющей». Кроме того, «источником» (началом) и «приемником» (концом) каждой функциональной дуги могут быть только функциональные блоки, при этом «источником» может быть только выходная сторона блока, а «приемником» – любая из трех оставшихся.

Необходимо отметить, что любой функциональный блок (рис. 1.19) по требованиям стандарта должен иметь, по крайней мере, одну управляющую интерфейсную дугу и одну исходящую. Это и понятно – каждый процесс должен происходить по каким-то правилам (отображаемым управляющей дугой) и должен выдавать некоторый результат (исходящая дуга).

При построении IDEF0-диаграмм важно правильно отделять входящие интерфейсные дуги от управляющих, что часто бывает непросто. К примеру, на рис. 1.20 изображен функциональный блок «Обработать заготовку». В реальном процессе рабочему, производящему обработку, выдают заготовку и технологические указания по обработке (или правила техники безопасности при работе со станком). Ошибочно может показаться, что и заготовка и документ с технологическими указаниями являются входящими объектами, однако это не так. На самом деле в этом процессе заготовка обрабатывается по правилам, отраженным в технологических указаниях, которые должны соответственно изображаться управляющей интерфейсной дугой.

Другое дело, когда технологические указания обрабатываются главным технологом и в них вносятся изменения (рис. 1.21). В этом случае они отображаются уже входящей интерфейсной дугой, а управляющим объектом являются, например, новые промышленные стандарты (стандарт предприятия), исходя из которых производятся данные изменения.

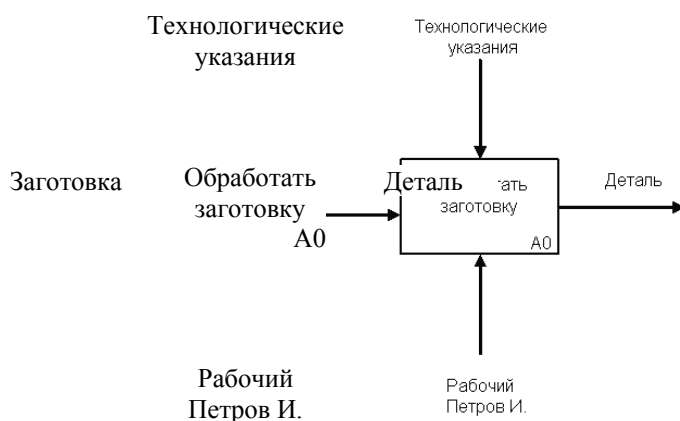


Рис. 1.20 Функциональный блок «Обработать заготовку»



Рис. 1.21 Функциональный блок

Приведенные выше примеры подчеркивают внешне схожую природу входящих и управляющих интерфейсных дуг, однако для систем одного класса всегда есть определенные разграничения. Например, в случае рассмотрения предприятий и организаций существуют пять основных видов объектов: материальные потоки (детали, товары, сырье и т.д.), финансовые потоки (наличные и безналичные, инвестиции и т.д.), потоки документов (коммерческие, финансовые и организационные документы), потоки информации (информация, данные о намерениях, устные распоряжения и т.д.) и ресурсы (сотрудники, станки, машины и т.д.). При этом в различных случаях входящими и исходящими интерфейсными дугами могут отображаться все виды объектов, управляющими – только относящиеся к потокам документов и информации, а дугами-механизмами – только ресурсы.

Обязательное наличие управляющих интерфейсных дуг является одним из главных отличий стандарта IDEF0 от других методологий классов DFD (Data Flow Diagram) и WFD (Work Flow Diagram).

Третьим основным понятием стандарта IDEF0 является *декомпозиция* (Decomposition). Принцип декомпозиции применяется при разбиении сложного процесса на составляющие его функции. При этом уровень детализации процесса определяется непосредственно разработчиком модели.

Декомпозиция позволяет постепенно и структурированно представлять модель системы в виде иерархической структуры отдельных диаграмм, что делает ее менее перегруженной и легко усваиваемой.

Модель IDEF0 всегда начинается с представления системы как единого целого – одного функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области. Такая диаграмма с одним функциональным блоком называется контекстной диаграммой и обозначается идентификатором «A-0».

В пояснительном тексте к контекстной диаграмме должна быть указана цель (Purpose) построения диаграммы в виде краткого описания и зафиксирована точка зрения (Viewpoint).

Определение и формализация цели разработки IDEF0-модели является крайне важным моментом. Фактически цель определяет соответствующие области в исследуемой системе, на которые необходимо фокусироваться в первую очередь. Например, если моделируется деятельность предприятия с целью построения в дальнейшем на базе этой модели информационной системы, то эта модель будет существенно отличаться от той, которую бы разрабатывали для того же самого предприятия, но уже с целью оптимизации логистических цепочек.

Точка зрения определяет основное направление развития модели и уровень необходимой детализации. Четкое фиксирование точки зрения позволяет разгрузить модель, отказавшись от детализации и исследования отдельных элементов, не являющихся необходимыми, исходя из выбранной точки зрения на систему. Например, функциональные модели одного и того же предприятия с точек зрения главного технолога и финансового директора будут существенно различаться по направленности их детализации. Это связано с тем, что в конечном итоге финансового директора не интересуют аспекты обработки сырья на производственном оборудовании, а главному технологу ни к чему прорисованные схемы финансовых потоков. Правильный выбор точки зрения существенно сокращает временные затраты на построение конечной модели предприятия.

В процессе декомпозиции функциональный блок, который в контекстной диаграмме отображает систему как единое целое, подвергается детализации на другой диаграмме. Получившаяся диаграмма второго уровня содержит функциональные блоки, отображающие главные подфункции функционального блока контекстной диаграммы, и называется дочерней (Child diagram) по отношению к нему (каждый из функциональных блоков, принадлежащих дочерней диаграмме, соответственно называется дочерним блоком – Child Box). В свою очередь, функциональный блок-предок называется родительским блоком по отношению к дочерней диаграмме (Parent Box), а диаграмма, к которой он принадлежит – родительской диаграммой (Parent Diagram). Каждая из подфункций дочерней диаграммы может быть далее детализирована путем аналогичной декомпозиции соответствующего ей функционального блока. Важно отметить, что в каждом случае декомпозиции функционального блока все интерфейсные дуги, входящие в данный блок или исходящие из него, фиксируются на дочерней диаграмме. Этим достигается структурная целостность IDEF0-модели. Следует обратить внимание на взаимосвязь нумерации функциональных блоков и диаграмм – каждый блок имеет свой уникальный порядковый номер на диаграмме (цифра в правом нижнем углу прямоугольника), а обозначение под правым углом указывает на но-

мер дочерней для этого блока диаграммы. Отсутствие этого обозначения говорит о том, что декомпозиции для данного блока не существует.

Часто бывают случаи, когда отдельные интерфейсные дуги не имеет смысла продолжать рассматривать в дочерних диаграммах ниже какого-то определенного уровня в иерархии или наоборот – отдельные дуги не имеют практического смысла выше какого-то уровня. Например, интерфейсную дугу, изображающую «деталь» на входе в функциональный блок «Обработать на токарном станке» не имеет смысла отражать на диаграммах более высоких уровней – это будет только перегружать диаграммы и делать их сложными для восприятия. С другой стороны, случается необходимость избавиться от отдельных «концептуальных» интерфейсных дуг и не детализировать их глубже некоторого уровня. Для решения подобных задач в стандарте IDEF0 предусмотрено понятие *туннелирования*. Обозначение «туннеля» (Arrow Tunnel) в виде двух круглых скобок вокруг начала интерфейсной дуги означает, что эта дуга не была унаследована от функционального родительского блока и появилась (из «туннеля») только на этой диаграмме. В свою очередь, такое же обозначение вокруг конца (стрелки) интерфейсной дуги в непосредственной близости от блока-приемника означает тот факт, что в дочерней по отношению к этому блоку диаграмме эта дуга отображаться и рассматриваться не будет. Чаще всего бывает, что отдельные объекты и соответствующие им интерфейсные дуги не рассматриваются на некоторых промежуточных уровнях иерархии – в таком случае они сначала «погружаются в туннель», а затем при необходимости «возвращаются из туннеля».

Последним из понятий IDEF0 является *гlossарий* (Glossary). Для каждого из элементов IDEF0: диаграмм, функциональных блоков, интерфейсных дуг существующий стандарт подразумевает создание и поддержание набора соответствующих определений, ключевых слов, повествовательных изложений и т.д., которые характеризуют объект, отображенный данным элементом. Этот набор называется глоссарием и является описанием сущности данного элемента. Например, для управляющей интерфейсной дуги «распоряжение об оплате» глоссарий может содержать перечень полей соответствующего дуге документа, необходимый набор виз и т.д. Глоссарий гармонично дополняет наглядный графический язык, снабжая диаграммы необходимой дополнительной информацией.

Обычно IDEF0-модели несут в себе сложную и концентрированную информацию, и для того, чтобы ограничить их перегруженность и сделать удобочитаемыми, в соответствующем стандарте приняты соответствующие ограничения сложности:

1) ограничение количества функциональных блоков на диаграмме тремя – шестью. Верхний предел (шесть) заставляет разработчика использовать иерархии при описании сложных предметов, а нижний предел (три) гарантирует, что на соответствующей диаграмме достаточно деталей, чтобы оправдать ее создание;

2) ограничение количества подходящих к одному функциональному блоку (выходящих из одного функционального блока) интерфейсных дуг четырьмя.

Разумеется, строго следовать этим ограничениям вовсе необязательно, однако, как показывает опыт, они являются весьма практичными в реальной работе.

Стандарт IDEF0 содержит набор процедур, позволяющих разрабатывать и согласовывать модель большой группой людей, принадлежащих к разным областям деятельности моделируемой системы. Обычно процесс разработки является итеративным и состоит из следующих условных этапов:

1 Создание модели группой специалистов, относящихся к различным сферам деятельности предприятия. Эта группа в терминах IDEF0 называется авторами (Authors). Построение первоначальной модели является динамическим процессом, в течение которого авторы опрашивают компетентных лиц о структуре различных процессов. На основе имеющихся положений, документов и результатов опросов создается черновик (Model Draft) модели.

2 Распространение черновика для рассмотрения, согласований и комментариев. На этой стадии происходит обсуждение черновика модели с широким спектром компетентных лиц (в терминах IDEF0 – читателей) на предприятии. При этом каждая из диаграмм черновой модели письменно критикуется и комментируется, а затем передается автору. Автор, в свою очередь, также письменно соглашается с критикой или отвергает ее с изложением логики принятия решения и вновь возвращает откорректированный черновик для дальнейшего рассмотрения. Этот цикл продолжается до тех пор, пока авторы и читатели не придут к единому мнению.

3 Официальное утверждение модели. Утверждение согласованной модели происходит руководителем рабочей группы в том случае, если у авторов модели и читателей отсутствуют разногласия по поводу ее адекватности. Окончательной моделью является согласованное представление о предприятии (системе) с заданной точки зрения и для заданной цели.

Наглядность графического языка IDEF0 делает модель вполне читаемой и для лиц, которые не принимали участия в проекте ее создания, а также эффективной для проведения показов и презентаций. В дальнейшем на базе построенной модели могут быть организованы новые проекты, нацеленные на производство изменений на предприятии (в системе).

В последние годы интерес в России к методологиям семейства IDEF неуклонно растет. При этом интерес к таким стандартам, как IDEF3-5 можно назвать теоретическим, а к IDEF0 вполне практически обоснованным. Собственно говоря, первые Case-средства, позволяющие строить DFD- и IDEF0-диаграммы, появились на российском рынке еще в 1996 г.

Не секрет, что практически все проекты обследования и анализа финансовой и хозяйственной деятельности предприятий сейчас в России, так или иначе, связаны с построением автоматизированных систем управления. Благодаря этому, стандарты IDEF в понимании большинства стали условно неотделимы от внедрения ин-

формационных технологий, хотя с их помощью порой можно эффективно решать даже небольшие локальные задачи, буквально при помощи карандаша и бумаги.

При осуществлении сложных проектов обследования предприятий разработка моделей в стандарте IDEF0 позволяет наглядно и эффективно отобразить весь механизм деятельности предприятия в нужном разрезе. Однако самое главное – это возможность коллективной работы, которую предоставляет IDEF0.

### 1.4.3 Стандарт IDEF1

Деятельность любого предприятия можно представить как непрерывное изменение состояния физических и интеллектуальных объектов, имеющих отношение к предприятию, таких как сотрудники, средства производства, производимые продукты, идеи, финансы и т.д. Для эффективного управления этим процессом каждое изменение того или иного объекта должно иметь свое документальное отображение. Этими отображениями служат личные дела сотрудников, отчеты, рекламная продукция, служебные записки и т.д. Их совокупность назовем информационной областью предприятия. Движение информации (например, документооборот) и изменение ее назовем информационными потоками. Очевидно, что любому бизнес-процессу, а также любому изменению физических объектов должен соответствовать определенный информационный поток. Более того, руководство, при построении стратегических планов развития и управлении деятельностью предприятия (издавая приказы, распоряжения и т.д.), фактически руководствуется информационными потоками и вносит в них изменения, таким образом, осуществляя информационный менеджмент.

Стандарт IDEF1 был разработан как инструмент для анализа и изучения взаимосвязей между информационными потоками в рамках коммерческой деятельности предприятия. Целью подобного исследования является дополнение и структуризация существующей информации и обеспечение качественного менеджмента информационными потоками. Необходимость в подобной реорганизации информационной области, как правило, возникает на начальном этапе построения корпоративной информационной системы, и методология IDEF1 позволяет достаточно наглядно обнаружить «черные дыры» и слабые места в существующей структуре информационных потоков. Применение методологии IDEF1 как инструмента построения наглядной модели информационной структуры предприятия по принципу «Как должно быть» позволяет решить следующие задачи:

- выявить структуру и содержание существующих потоков информации на предприятии;
- определить, какие проблемы, выявленные в результате функционального анализа и анализа потребностей, вызваны недостатком управления соответствующей информацией;
- выявить информационные потоки, требующие дополнительного управления для эффективной реализации модели.

С помощью IDEF1 происходит изучение существующей информации о различных объектах в области деятельности предприятия. Характерно то, что IDEF1-модель включает в рассмотрение не только автоматизированные компоненты, базы данных и соответствующую им информацию, но также и реальные объекты, такие как сами сотрудники, кабинеты, телефоны и т.д. Миссия методологии IDEF1 состоит в том, чтобы выявить и четко постулировать потребности в информационном менеджменте в рамках коммерческой деятельности предприятия. В отличие от методов разработки структур баз данных (например, IDEF1X), IDEF1 является аналитическим методом и используется преимущественно для выполнения следующих действий:

- определение самой информации и структуры ее потоков, имеющей отношение к деятельности предприятия;
- определение существующих правил и законов, по которым осуществляется движение информационных потоков, а также принципов управления ими;
- выяснение взаимосвязей между существующими информационными потоками в рамках предприятия;
- выявление проблем, возникающих вследствие недостатка качественного информационного менеджмента.

Результаты анализа информационных потоков могут быть использованы для стратегического и тактического планирования деятельности предприятия и улучшения информационного менеджмента.

Однако основной целью использования методологии IDEF1 все же остается исследование движения потоков информации и принципов управления ими на начальном этапе процесса проектирования корпоративной информационно-аналитической системы, которая будет способствовать более эффективному использованию информационного пространства. Наглядные модели IDEF1 обеспечивают базис для построения мощной и гибкой информационной системы.

Методология IDEF1 разделяет элементы структуры информационной области, их свойства и взаимосвязи на классы. Центральным понятием методологии IDEF1 является понятие *сущности*. Класс сущностей представляет собой совокупность информации, накопленной и хранящейся в рамках предприятия и соответствующей определенному объекту или группе объектов реального мира. Основными концептуальными свойствами сущностей в IDEF1 являются:

- 1) устойчивость (информация, имеющая отношение к той или иной сущности, постоянно накапливается);
- 2) уникальность (любая сущность может быть однозначно идентифицирована из другой сущности).

Каждая сущность имеет свое имя и атрибуты. Атрибуты представляют собой характерные свойства и признаки объектов реального мира, относящихся к определенной сущности. Класс атрибутов представляет собой набор пар, состоящих из имени атрибута и его значения для определенной сущности. Атрибуты, по которым можно однозначно отличить одну сущность от другой, называются ключевыми атрибутами. Каждая сущность



может характеризоваться несколькими ключевыми атрибутами. Класс взаимосвязей в IDEF1 представляет собой совокупность взаимосвязей между сущностями. Взаимосвязь между двумя отдельными сущностями считается существующей в том случае, если класс атрибутов одной сущности содержит ключевые атрибуты другой сущности. Каждый из вышеописанных классов имеет свое условное графическое отображение согласно методологии IDEF1.

На рис. 1.22 приведен пример IDEF1-диаграммы. На ней представлены две сущности с именами «Отдел» и «Сотрудник» и взаимосвязь между ними с именем «работает в». Имя взаимосвязи всегда выражается в глагольной форме. Если же между двумя или несколькими объектами реального мира не существует установленной зависимости, то, с точки зрения IDEF1, между соответствующими им сущностями взаимосвязь также отсутствует.

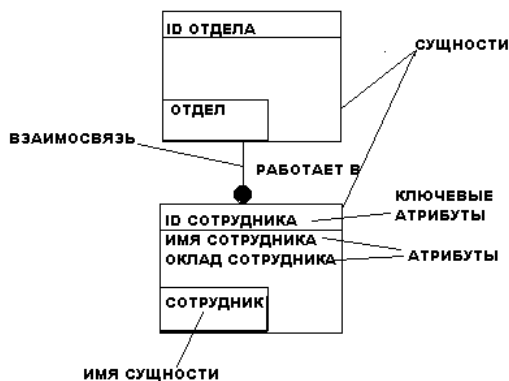


Рис. 1.22 IDEF1-диаграмма

В заключение стоит еще раз отметить, что стандарт IDEF1 является методом изучения и анализа в отличие от очень сходного по терминологии и семантике стандарта IDEF1X, предназначенного для разработки структуры реляционных баз данных и оперирующего с конкретными объектами физического мира.

#### 1.4.4 Стандарт IDEF1X

IDEF1X является методом для разработки реляционных баз данных и использует условный синтаксис, специально разработанный для удобного построения концептуальной схемы.

Известно, что основным компонентом реляционных баз данных является таблица. Таблицы используются для управления и хранения информации.

Таблицы в реляционной СУБД состоят из строк данных, однородных по своей природе. Другими словами, каждая строка таблицы описывает один экземпляр некоторой сущности, причем набор атрибутов каждого экземпляра постоянен.

Предположим, в базе данных хранится информация о покупателях. Таблица «покупатель» содержит 3 колонки и 4 строки:

Имя	Адрес	Идент. карты
Сидоров	1 улица	444444
Иванов	2 улица	222222
Петров	3 улица	333333
Павлов	4 улица	111111

Имя таблицы и имена ее колонок составляют структуру таблицы: customer (name, address, card\_id). В реляционной СУБД все значения данных являются атомарными, т.е. нельзя в клетке таблицы хранить список значений.

Таблицы в реляционной СУБД соответствуют (не обязательно совпадают по имени) сущностям, а колонки – атрибутам.

Концептуальной схемой назовем универсальное представление структуры данных в рамках коммерческого предприятия, независимое от конечной реализации базы данных и аппаратной платформы. Будучи статическим методом разработки, IDEF1X изначально не предназначен для динамического анализа, тем не менее, он иногда применяется в этом качестве как альтернатива методу IDEF1. Использование метода IDEF1X наиболее целесообразно для построения логической структуры базы данных после того, как все информационные ресурсы исследованы (скажем, с помощью метода IDEF1) и решение о внедрении реляционной базы данных, как части корпоративной информационной системы, было принято. Однако, не стоит забывать, что средства моделирования IDEF1X специально разработаны для построения реляционных информационных систем, и если существует необходимость проектирования другой системы, скажем объектно-ориентированной, то лучше избрать другие методы моделирования.

Существует несколько очевидных причин, по которым IDEF1X не следует применять в случае построения нереляционных баз данных. Во-первых, IDEF1X требует от проектировщика определить ключевые атрибуты,

чтобы отличить одну сущность от другой, в то время как объектно-ориентированные системы не требуют задания ключевых атрибутов в целях идентификации объектов. Во-вторых, в тех случаях, когда более чем один атрибут является однозначно идентифицирующей сущность, проектировщик должен определить один из этих атрибутов первичным ключом, а все остальные вторичными. И, таким образом, построенная проектировщиком IDEF1X-модель и переданная для окончательной реализации программисту является некорректной для применения методов объектно-ориентированной реализации и предназначена для построения реляционной системы.

Хотя терминология IDEF1X практически совпадает с терминологией IDEF1, существует ряд фундаментальных отличий в теоретических концепциях этих методологий. Сущность в IDEF1X описывает собой совокупность или набор экземпляров, похожих по свойствам, но однозначно отличаемых друг от друга по одному или нескольким признакам. Каждый экземпляр является реализацией сущности. Таким образом, сущность в IDEF1X описывает конкретный набор экземпляров реального мира, в отличие от сущности в IDEF1, которая представляет собой абстрактный набор информационных отображений реального мира. Примером сущности IDEF1X может быть сущность СОТРУДНИК, которая представляет собой всех сотрудников предприятия, а один из них, скажем, Иванов Иван Иванович, является конкретной реализацией этой сущности. В примере, приведенном на рис. 1.22, каждый экземпляр сущности СОТРУДНИК содержит следующую информацию: ID сотрудника, имя сотрудника, адрес сотрудника и т.п. В IDEF1X-модели эти свойства называются атрибутами сущности. Каждый атрибут содержит только часть информации о сущности.

Связи в IDEF1X представляют собой ссылки, соединения и ассоциации между сущностями. Связи это суть глаголы, которые показывают, как соотносятся сущности между собой. Ниже приведен ряд примеров связи между сущностями:

Отдел <состоит из> нескольких СОТРУДНИКОВ.

Автобус <перевозит> нескольких ПАССАЖИРОВ.

Сотрудник <пишет> разные ОТЧЕТЫ.

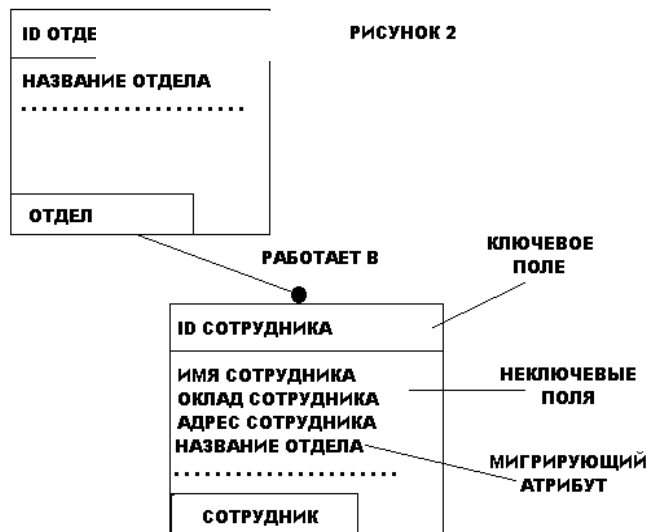
Во всех перечисленных примерах взаимосвязи между сущностями соответствуют схеме «один-многим». Это означает, что один экземпляр первой сущности связан с несколькими экземплярами второй сущности. Причем первая сущность называется родительской, а вторая – дочерней. В приведенных примерах глаголы заключены в угловые скобки. Связи отображаются в виде линии между двумя сущностями с точкой на одном конце и глагольной фразой, отображаемой над линией. На рис. 1.22 приводится диаграмма связи между СОТРУДНИКОМ и ОТДЕЛОМ.

Отношения «многие-ко-многим» обычно используются на начальной стадии разработки диаграммы, например, в диаграмме зависимости сущностей и отображаются в IDEF1X в виде сплошной линии с точками на обоих концах. Так как отношения «многие-ко-многим» могут скрыть другие бизнес-правила или ограничения, они должны быть полностью исследованы на одном из этапов моделирования. Например, иногда отношение «многие-ко-многим» на ранних стадиях моделирования идентифицируется неправильно, на самом деле представляя два или несколько случаев отношений «один-ко-многим» между связанными сущностями. Или, в случае необходимости хранения дополнительных сведений о связи «многие-ко-многим», например, даты или комментария, такая связь должна быть заменена дополнительной сущностью, содержащей эти сведения. При моделировании необходимо быть уверенным в том, что все отношения «многие-ко-многим» будут подробно обсуждены на более поздних стадиях моделирования для обеспечения правильного моделирования отношений.

Сущность описывается в диаграмме IDEF1X графическим объектом в виде прямоугольника. На рис. 1.23 приведен пример IDEF1X-диаграммы. Каждый прямоугольник, отображающий собой сущность, разделяется горизонтальной линией на часть, в которой расположены ключевые поля, и часть, где расположены неключевые поля. Верхняя часть называется ключевой областью, а нижняя часть – областью данных. Ключевая область объекта СОТРУДНИК содержит поле «Уникальный идентификатор сотрудника», в области данных находятся поля «Имя сотрудника», «Адрес сотрудника», «Телефон сотрудника» и т.д.

Ключевая область содержит первичный ключ для сущности. Первичный ключ – это набор атрибутов, выбранных для идентификации уникальных экземпляров сущности. Атрибуты первичного ключа располагаются над линией в ключевой области. Как следует из названия, неключевой атрибут – это атрибут, который не был выбран ключевым. Неключевые атрибуты располагаются под чертой, в области данных.

При создании сущности в IDEF1X-модели одним из главных вопросов, на который нужно ответить, является: как можно идентифицировать уникальную запись? Для этого требуется уникальная идентификация каждой записи в сущности для того, чтобы правильно создать логическую модель данных. Напомним, что сущности в IDEF1X всегда имеют ключевую область и поэтому в каждой сущности должны быть определены ключевые атрибуты.



**Рис. 1.23 IDEF1X-диаграмма**

Выбор первичного ключа для сущности является очень важным шагом и требует особого внимания. В качестве первичных ключей могут быть использованы несколько атрибутов или групп атрибутов. Атрибуты, которые могут быть выбраны первичными ключами, называются кандидатами в ключевые атрибуты (потенциальные атрибуты). Кандидаты в ключи должны уникально идентифицировать каждую запись сущности. В соответствии с этим ни одна из частей ключа не может быть NULL, не заполненной или отсутствующей.

Например, для того, чтобы корректно использовать сущность СОТРУДНИК в IDEF1X-модели данных (а позже в базе данных), необходимо иметь возможность уникально идентифицировать записи. Правила, по которым вы выбираете первичный ключ из списка предполагаемых ключей, очень строги, однако могут быть применены ко всем типам баз данных и информации. Правила устанавливают, что атрибуты и группы атрибутов должны:

- уникальным образом идентифицировать экземпляр сущности;
- не использовать NULL значений;
- не изменяться со временем;
- экземпляр идентифицируется при помощи ключа. При изменении ключа соответственно меняется экземпляр.
- быть как можно более короткими для использования индексирования и получения данных. Если вам нужно использовать ключ, являющийся комбинацией ключей из других сущностей, убедитесь в том, что каждая из частей ключа соответствует правилам.

Для наглядного представления о том, как целесообразно выбирать первичные ключи, приведем следующий пример – выберем первичный ключ для сущности СОТРУДНИК:

- Атрибут «ID сотрудника» является потенциальным ключом, так как он уникален для всех экземпляров сущности СОТРУДНИК.
- Атрибут «Имя сотрудника» не очень хорош для потенциального ключа, так как среди служащих на предприятии могут быть, к примеру, двое Иванов Петровых.
- Атрибут «Номер страхового полиса сотрудника» является уникальным, но проблема в том, что СОТРУДНИК может не иметь такового.

Комбинация атрибутов «имя сотрудника» и «дата рождения сотрудника» может оказаться удачной для наших целей и стать искомым потенциальным ключом.

После проведенного анализа можно назвать два потенциальных ключа – первый «Номер сотрудника» и комбинация, включающая поля «Имя сотрудника» и «Дата рождения сотрудника». Так как атрибут «Номер сотрудника» имеет самые короткие и уникальные значения, то он лучше других подходит для первичного ключа.

При выборе первичного ключа для сущности разработчики модели часто используют дополнительный (суррогатный) ключ, т.е. произвольный номер, который уникальным образом определяет запись в сущности. Атрибут «Номер сотрудника» является примером суррогатного ключа. Суррогатный ключ лучше всего подходит на роль первичного ключа потому, что является коротким и быстрее всего идентифицирует экземпляры в объекте. К тому же суррогатные ключи могут автоматически генерироваться системой так, чтобы нумерация была сплошной, т.е. без пропусков.

Потенциальные ключи, которые не выбраны первичными, могут быть использованы в качестве вторичных или альтернативных. С помощью альтернативных ключей часто отображают различные индексы доступа к данным в конечной реализации реляционной базы.

Если сущности в IDEF1X-диаграмме связаны, связь передает ключ (или набор ключевых атрибутов) дочерней сущности. Эти атрибуты называются внешними ключами. Внешние ключи определяются как атрибуты первичных ключей родительского объекта, переданные дочернему объекту через их связь. Передаваемые атрибуты называются мигрирующими.

При разработке модели зачастую приходится сталкиваться с сущностями, уникальность которых зависит от значений атрибута внешнего ключа. Для этих сущностей (для уникального определения каждой сущности) внешний ключ должен быть частью первичного ключа дочернего объекта.

Дочерняя сущность, уникальность которой зависит от атрибута внешнего ключа, называется зависимой. В примере на рис. 1.23 сущность СОТРУДНИК является зависимой потому, что ее идентификация зависит от сущности ОТДЕЛ.

Атрибуты изображаются в виде списка имен внутри блока сущности. Атрибуты, определяющие первичный ключ, размещаются наверху списка и отделяются от других атрибутов горизонтальной чертой (рис. 1.24).

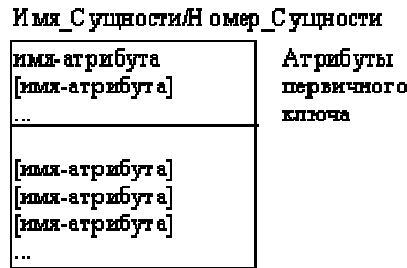


Рис. 1.24 Атрибуты и первичные ключи

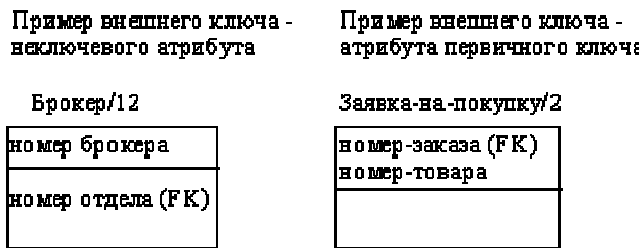


Рис. 1.25 Примеры внешних ключей

Внешний ключ (Foreign Key, FK) изображается с помощью помещения внутрь блока сущности имен атрибутов, после которых следуют буквы FK в скобках (рис. 1.25).

Зависимые сущности далее классифицируются на сущности, которые не могут существовать без родительской сущности, и сущности, которые не могут быть идентифицированы без использования ключа родителя (сущности, зависящие от идентификации). Сущность СОТРУДНИК принадлежит ко второму типу зависимых сущностей, так как сотрудники могут существовать и без отдела.

Напротив, есть ситуации, в которых сущность зависит от существования другой сущности. Рассмотрим две сущности: ЗАПРОС, используемый для отслеживания запросов покупателей, и ПОЗИЦИЯ ЗАПРОСА, которая отслеживает отдельные элементы в ЗАПРОСЕ. Связь между этими двумя сущностями может быть выражена в виде ЗАПРОС <содержит> один или несколько ПОЗИЦИЙ ЗАПРОСА. В этом случае, ПОЗИЦИЯ ЗАПРОСА зависит от существования ЗАКАЗА.

Сущности, не зависящие при идентификации от других объектов в модели, называются независимыми. В вышеописанном примере сущность ОТДЕЛ можно считать независимой. В IDEF1X независимые сущности представлены в виде прямоугольников, а зависимые сущности представлены в виде закругленных прямоугольников (рис. 1.26).

В IDEF1X концепция зависимых и независимых сущностей усиливается типом взаимосвязей между двумя сущностями. Если вы хотите, чтобы внешний ключ передавался в дочернюю сущность (и, в результате, создавал зависимую сущность), то можете создать идентифицирующую связь между родительской и дочерней сущностями.

Идентифицирующие взаимосвязи обозначаются сплошной линией между сущностями (рис. 1.27).

Неидентифицирующие связи (рис. 1.28), являющиеся уникальными для IDEF1X, также связывают родительскую сущность с дочерней и используются для отображения другого типа передачи атрибутов внешних ключей – передачи в область данных дочерней сущности (под линией).

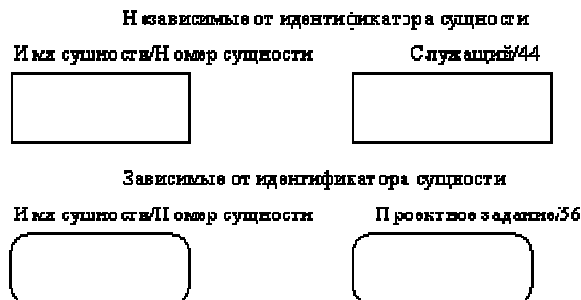


Рис. 1.26 Независимые и зависимые от идентификатора сущности

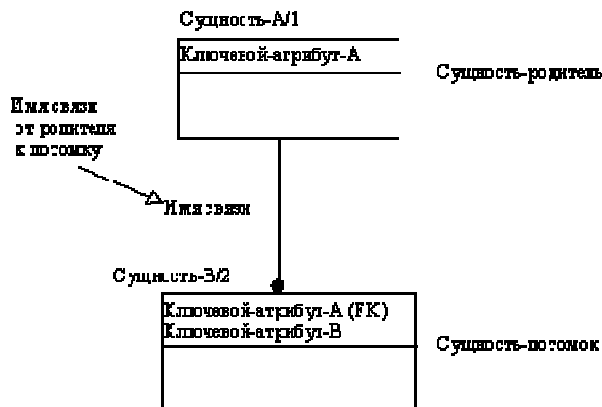


Рис. 1.27 Идентифицирующая связь

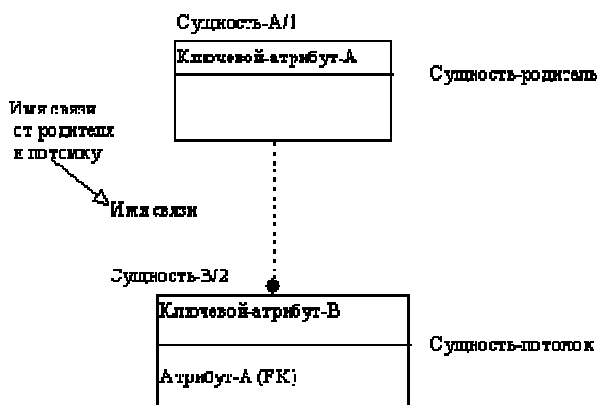


Рис. 1.28 Неидентифицирующая связь

Неидентифицирующие связи отображаются пунктирной линией между объектами. Так как переданные ключи в неидентифицирующей связи не являются составной частью первичного ключа дочерней сущности, то этот вид связи не проявляется ни в одной идентифицирующей зависимости. В этом случае и ОТДЕЛ, и СОТРУДНИК рассматриваются как независимые сущности.

Тем не менее, взаимосвязь может отражать зависимость существования, если бизнес-правило для взаимосвязи определяет то, что внешний ключ не может принимать значение NULL. Если внешний ключ должен существовать, то это означает, что запись в дочерней сущности может существовать только при наличии ассоциированной с ним родительской записи.

### 1.4.5 Стандарт IDEF3

IDEF3 является стандартом документирования технологических процессов, происходящих на предприятии, и предоставляет инструментарий для наглядного исследования и моделирования их сценариев. Сценарием (Scenario) назовем описание последовательности изменений свойств объекта в рамках рассматриваемого процесса (например, описание последовательности этапов обработки детали в цеху и изменение ее свойств после прохождения каждого этапа). Исполнение каждого сценария сопровождается соответствующим документооборотом, который состоит из двух основных потоков: документов, определяющих структуру и последовательность процесса (технологических указаний, описаний стандартов и т.д.), и документов, отображающих ход его выполнения (результатов тестов и экспертиз, отчетов о браке и т.д.). Для эффективного управления любым технологическим процессом необходимо иметь детальное представление об его сценарии и структуре сопутствующего документооборота. Средства документирования и моделирования IDEF3 позволяют выполнять следующие задачи:

- документировать имеющиеся данные о технологии процесса, выявленные, скажем, в процессе опроса компетентных сотрудников, ответственных за организацию рассматриваемого процесса;
- определять и анализировать точки влияния потоков сопутствующего документооборота на сценарий технологических процессов;
- определять ситуации, в которых требуется принятие решения, влияющего на жизненный цикл процесса, например изменение конструктивных, технологических или эксплуатационных свойств конечного продукта;
- содействовать принятию оптимальных решений при реорганизации технологических процессов;
- разрабатывать имитационные модели технологических процессов по принципу «КАК БУДЕТ, ЕСЛИ...».

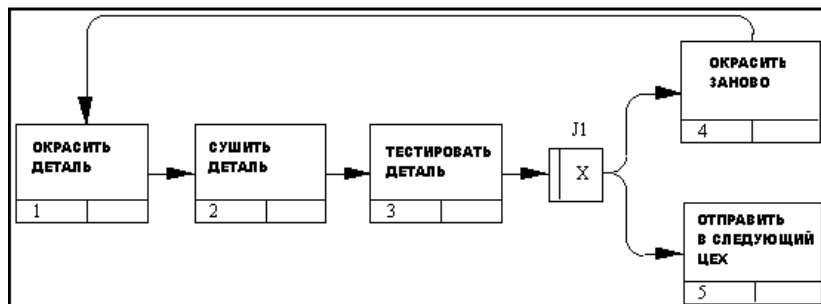
Существуют два типа диаграмм в стандарте IDEF3, представляющие описание одного и того же сценария технологического процесса в разных ракурсах. Диаграммы, относящиеся к первому типу, называются диаграммами «Описания последовательности этапов процесса» (Process Flow Description Diagrams, PFDD), а ко второму – диаграммами «Состояния объекта и его трансформаций в процессе» (Object State Transition Network, OSTN). Предположим, требуется описать процесс окраски детали в производственном цехе на предприятии. С помо-

шью диаграмм PFDD документируются последовательность и описание стадий обработки детали в рамках исследуемого технологического процесса. Диаграммы OSTN используются для иллюстрации трансформаций детали, которые происходят на каждой стадии обработки.

На следующем примере опишем, как графические средства IDEF3 позволяют документировать вышеуказанный производственный процесс окраски детали. В целом, этот процесс состоит непосредственно из самой окраски, производимой на специальном оборудовании, и этапа контроля ее качества, который определяет, нужно ли деталь окрасить заново (в случае несоответствия стандартам и выявления брака) или отправить ее в дальнейшую обработку.

На рис. 1.29 изображена диаграмма PFDD, являющаяся графическим отображением сценария обработки детали. Прямоугольники на диаграмме PFDD называются функциональными элементами или элементами поведения (Unit of Behavior, UOB) и обозначают событие, стадию процесса или принятие решения. Каждый UOB имеет свое имя, отображаемое в глагольной форме, и уникальный номер. Стрелки или линии являются отображением перемещения детали между UOB-блоками в ходе процесса. Линии бывают следующих видов:

- старшая (Precedence) – сплошная линия, связывающая UOB. Рисуется слева направо или сверху вниз;



**Рис. 1.29** Пример PFDD-диаграммы

- отношения (Relational Link) – пунктирная линия, используемая для изображения связей между UOB;
- потоки объектов (Object Flow) – стрелка с двумя наконечниками применяется для описания того факта, что объект (деталь) используется в двух или более единицах работы, например, когда объект порождается в одной работе и используется в другой.

ется в одной работе и используется в другой.

Объект, обозначенный J1 – называется перекрестком (Junction). Перекрестки используются для отображения логики взаимодействия стрелок (потоков) при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы. Различают перекрестки для слияния (Fan-in Junction) и разветвления (Fan-out Junction) стрелок.

Перекресток не может использоваться одновременно для слияния и для разветвления. При внесении перекрестка в диаграмму необходимо указать тип перекрестка. Классификация возможных типов перекрестков приведена в табл. 1.2.

Сценарий, отображаемый на диаграмме, можно описать в следующем виде. Деталь поступает в окрасочный цех, подготовленная к окраске. В процессе окраски наносится один слой эмали при высокой температуре. После этого производится сушка детали, затем начинается этап проверки качества нанесенного слоя. Если тест подтверждает недостаточное качество нанесенного слоя (недостаточную толщину, неоднородность и т.д.), то деталь заново пропускается через цех окраски. Если деталь успешно проходит контроль качества, то она отправляется в следующий цех для дальнейшей обработки.

Каждый функциональный блок UOB может иметь последовательность декомпозиций и, следовательно, может быть детализирован с любой необходимой точностью. Под декомпозицией понимается представление каждого UOB с помощью отдельной IDEF3-диаграммы. Например, можно декомпонировать UOB «Окрасить Деталь», представив его отдельным процессом и построив для него свою PFDD-диаграмму. При этом эта диаграмма будет называться дочерней по отношению к изображенной на рис. 1.29, а та, соответственно – родительской. Номера UOB дочерних диаграмм имеют сквозную нумерацию, т.е., если родительский UOB имеет номер «1», то блоки UOB на его декомпозиции будут соответственно иметь номера «1.1», «1.2» и т.д. Применение принципа декомпозиции в IDEF3 позволяет структурировано описывать процессы с любым требуемым уровнем детализации.

Если диаграммы PFDD – технологический процесс «С точки зрения наблюдателя», то другой класс диаграмм IDEF3 OSTN позволяет рассматривать тот же самый процесс «С точки зрения объекта». На рис. 1.30 представлено отображение процесса окраски с точки зрения OSTN-диаграммы. Состояния объекта (в нашем случае детали) и изменения состояния являются ключевыми понятиями OSTN-диаграммы.

## 1.2 Классификация возможных типов перекрестков

Обозначение	Наименование	Смысл в случае слияния стрелок (Fan-in Junction)	Смысл в случае разветвления стрелок (Fan-out Junction)
	Asynchronous AND	Все предшествующие процессы должны быть завершены	Все следующие процессы должны быть запущены
	Synchronous AND	Все предшествующие процессы завершены одновременно	Все следующие процессы запускаются одновременно

	Asynchronous OR	Один или несколько предшествующих процессов должны быть завершены	Один или несколько следующих процессов должны быть запущены
	Synchronous OR	Один или несколько предшествующих процессов завершаются одновременно	Один или несколько следующих процессов запускаются одновременно
	XOR (Exclusive OR)	Только один предшествующий процесс завершен	Только один следующий процесс запускается

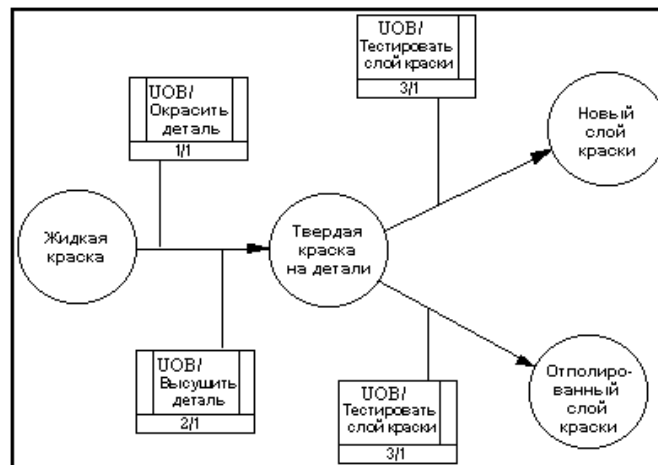


Рис. 1.30 Пример OSTN-диаграммы

Состояния объекта отображаются окружностями, а их изменения – направленными линиями. Каждая линия имеет ссылку на соответствующий функциональный блок UOB, в результате которого произошло отображаемое ей изменение состояния объекта.

#### 1.4.6 Методология функционально-стоимостного анализа

Считается, что функционально-стоимостной анализ ФСА (ABC, Activity Based Costing) является достаточно сложным для понимания. Возможно, это связано с тем, что существует слишком мало информации, объясняющей, что же он собственно из себя представляет. Целью данного параграфа является раскрытие сущности функционально-стоимостного анализа, простоты его применения, а также исключение элемента загадочности, связанного с ним.

Функционально-стоимостной анализ позволяет выполнить следующие виды работ:

- определение и проведение общего анализа себестоимости бизнес-процессов на предприятии (маркетинг, производство продукции и оказание услуг, сбыт, менеджмент качества, техническое и гарантийное обслуживание и др.);
- проведение функционального анализа, связанного с установлением и обоснованием выполняемых структурными подразделениями предприятий функций с целью обеспечения выпуска высокого качества продукции и оказания услуг;
- определение и анализ основных, дополнительных и ненужных функциональных затрат;
- сравнительный анализ альтернативных вариантов снижения затрат в производстве, сбыте и управлении за счет упорядочения функций структурных подразделений предприятия;
- анализ интегрированного улучшения результатов деятельности предприятия.

В настоящее время метод ФСА стал всеобъемлющим инструментом оценки систем, процессов и предприятий.

Функционально-стоимостной анализ – метод определения стоимости и других характеристик изделий, услуг и потребителей, использующих в качестве основы функции и ресурсы, задействованные в производстве, маркетинге, продаже, доставке, технической поддержке, оказании услуг, обслуживании клиентов, а также обеспечении качества.

Метод ФСА разработан как «операционно-ориентированная» альтернатива традиционным финансовым подходам. В частности, в отличие от традиционных финансовых подходов метод ФСА:

- предоставляет информацию в форме, понятной для персонала предприятия, непосредственно участвующего в бизнес-процессе;
- распределяет накладные расходы в соответствии с детальным просчетом использования ресурсов, подробным представлением о процессах и их влиянием на себестоимость, а не на основании прямых затрат или учета полного объема выпускаемой продукции.

ФСА – один из методов, позволяющий указать на возможные пути улучшения стоимостных показателей. Цель создания ФСА-модели для совершенствования деятельности предприятий – достичь улучшений в работе предприятий по показателям стоимости, трудоемкости и производительности. Проведение расчетов по ФСА-модели позволяет получить большой объем ФСА-информации для принятия решения.

В основе метода ФСА лежат данные, которые обеспечивают менеджеров информацией, необходимой для обоснования и принятия управленческих решений при применении таких методов, как:

- «точно в срок» (Just-in-time, JIT) и KANBAN;
- глобальное управление качеством (Total Quality Management, TQM);
- непрерывное улучшение (Kaizen);
- реинжиниринг бизнес-процессов (Business Process Reengineering, BPR).

Концепция ФСА позволяет представить управленческую информацию в виде финансовых показателей. Используя в качестве единиц измерения финансовых показателей просто доллар или рубль, ФСА-метод отображает финансовое состояние компании лучше, чем это делает традиционный бухгалтерский учет. Это происходит потому, что ФСА-метод физически отражает функции людей, машин и оборудования. ФСА-метод отображает уровень потребления ресурсов функциями, а также причины, по которым эти ресурсы используются.

ФСА-информацию можно использовать как для текущего (оперативного) управления, так и для принятия стратегических решений. На уровне тактического управления информацию из ФСА-модели можно использовать для формирования рекомендаций по увеличению прибыли и повышению эффективности деятельности организации. На стратегическом – помощь в принятии решений относительно реорганизации предприятия, изменения ассортимента продуктов и услуг, выхода на новые рынки, диверсификации и т.д. ФСА-информация показывает, как можно перераспределить ресурсы с максимальной стратегической выгодой, помогает выявить возможности тех факторов (качество, обслуживание, снижение стоимости, уменьшение трудоемкости), которые имеют наибольшее значение, а также определить наилучшие варианты капиталовложений.

Основные направления использования ФСА-модели для реорганизации бизнес-процессов – это повышение производительности, снижение стоимости, трудоемкости, времени и повышение качества.

Повышение производительности включает в себя три этапа. На первом этапе осуществляется анализ функций для определения возможностей повышения эффективности их выполнения. На втором – выявляются причины непропорциональных расходов и пути их устранения. И, наконец, на третьем этапе осуществляется мониторинг и ускорение нужных изменений с помощью измерения основных параметров производительности.

Что касается снижения стоимости, трудоемкости и времени, то с помощью ФСА-метода можно так реорганизовать деятельность, что будет достигнуто устойчивое их сокращение. Для этого необходимо сделать следующее:

- сократить время, необходимое для выполнения функций;
- устранить ненужные функции;
- сформировать ранжированный перечень функций по стоимости, трудоемкости или времени;
- выбрать функции с низкой стоимостью, трудоемкостью и временем;
- организовать совместное использование всех возможных функций;
- перераспределить ресурсы, высвободившиеся в результате усовершенствий.

Очевидно, что вышеперечисленные действия улучшают качество бизнес-процессов. Повышение качества бизнес-процессов осуществляется за счет проведения сравнительной оценки и выбора рациональных (по стоимостному или временному критерию) технологий выполнения операций или процедур.

В основе управления, основанного на функциях, лежат несколько аналитических методов, использующих ФСА-информацию. Это – стратегический анализ, стоимостной анализ, временной анализ, анализ трудоемкости, определение целевой стоимости и исчисление стоимости, исходя из жизненного цикла продукта или услуги.

Одним из направлений использования принципов, средств и методов ФСА является планирование бюджета, основанное на функциях. Планирование бюджета использует ФСА-модель для определения объема работ и потребности в ресурсах, при этом можно выделить два пути:

- выбор приоритетных направлений деятельности, увязанных со стратегическими целями;
- разработка реалистичного бюджета.

ФСА-информация позволяет принимать осознанные и целенаправленные решения о распределении ресурсов, опирающиеся на понимание взаимосвязей функций и стоимостных объектов, стоимостных факторов и объема работ.

Развитием ФСА-метода стал метод функционально-стоимостного управления (ФСУ, Activity-Based Management). ФСУ – это метод, который включает управление издержками на основе применения более точного отнесения издержек на процессы и продукцию.

Особо обращаем внимание на то, что ФСУ-метод позволяет не только определять издержки, но и управлять ими. Однако, не стоит ставить знак равенства между управлением и контролем. Данные ФСА/ФСУ используются больше для «предсказательного» моделирования, чем для контроля. На сегодняшний день использование данных об издержках для нужд контроля вытесняется более оперативной информацией от TQM-метода, реализованного в виде функций статистического контроля процессов (Statistical Process Control, SPC), или от интегрированных информационных систем, работающих в режиме реального времени.

В процессе построения функционально-стоимостных моделей удалось установить методологическую и технологическую взаимосвязь между IDEF0- и ФСА-моделями.

Связанность методов IDEF0 и ФСА заключается в том, что оба метода рассматривают предприятие, как множество последовательно выполняемых функций, а дуги входов, выходов, управления и механизмов IDEF0-



модели соответствуют стоимостным объектам и ресурсам ФСА-модели. На рис. 1.31 представлена концептуальная модель ФСА-метода, из которой четко видно, что Ресурсы (Затраты) в ФСА-модели – это входные дуги, дуги управления и механизмов в IDEF0-модели (см. рис. 1.19), Продукты (Стоимостные объекты) ФСА-модели – это выходные дуги IDEF0-модели, а Действия ФСА-метода – это Функции в IDEF0-модели.



Рис. 1.31 Концептуальная схема ФСА-метода

На более низком уровне, а именно, уровне функционального блока (рис. 1.32), связь IDEF0- и ФСА-моделей базируется на следующих принципах:

1 Функция характеризуется числом, которое представляет собой стоимость или время выполнения этой функции.

2 Стоимость или время функции, которая не имеет декомпозиции, определяется разработчиком системы.

Стоимость или время функции, которая имеет декомпозицию, определяется, как сумма стоимостей (времен) всех подфункций на данном уровне декомпозиции.

В заключение приведем классификацию базовых методологий для построения моделей бизнес-процессов (информационной модели КИС) (табл. 1.3).

В результате обследования предприятия строится функциональная модель существующей организации работы AS-IS («как есть»). На основе этой модели достигается консенсус между различными единицами бизнеса по вопросам, кто что сделал, и что каждая единица бизнеса добавляет в процесс. Модель AS-IS позволяет выяснить, что следует сделать сегодня перед тем, как решить, что следует сделать завтра. Внедрение информационной системы неизбежно приведет к перестройке существующих бизнес-процессов предприятия. Анализ функциональной модели позволяет понять, где находятся самые «узкие места», в чем будут состоять преимущества новых бизнес-процессов и насколько глубоким изменениям подвергнется существующая структура организации предприятия. Детализация бизнес-процессов позволяет выявить недостатки организации даже там, где функциональность на



Рис. 1.32 Функциональный блок и интерфейсные дуги

### 1.3 Классификация базовых методологий

Методы	Основные задачи, решаемые методом	Методологии	Инструментальные средства
Функциональное моделирование	Описание бизнес-процессов в виде системы взаимосвязанных функций	IDEF0	Design IDEF0 (Meta Software), BPwin (Logic Works)
Имитационное моделирование	Моделирование поведения системы в различных условиях: анализ крити-	IDEF3	Lotus Workflow, WorkRoute

	ческих режимов работы; анализ динамических характеристик бизнес-процессов		
Информационное моделирование	Описание информационной структуры объектов, идентификация отношений между объектами	IDEFIX IDEF4	ERwin (Logic Works), S-Designor (Powersoft), Power Désigner (Sybase)

первый взгляд кажется очевидной. Признаками несовершенной деятельности могут быть бесполезные, неуправляемые и дублирующиеся работы, неэффективный документооборот (нужного документа не оказывается в нужном месте в нужное время), отсутствие обратных связей по управлению (проведение работы не зависит от результата) и по входу (объекты или информация используются нерационально) и т.д.

Найденные в модели AS-IS недостатки можно исправить при создании модели TO-BE («как будет») – модели новой организации бизнес-процессов. Модель TO-BE нужна для оценки последствий внедрения КИС и анализа альтернативных, в том числе лучших, путей выполнения работы и документирования того, как предприятие будет функционировать в будущем. Как правило, строится несколько моделей TO-BE, из которых по какому-либо критерию выбирается наилучшая. Например, каждая из моделей TO-BE может соответствовать определенной ИС.

Проблема состоит в том, что таких критериев может быть много и непросто выделить важнейший. Для того, чтобы определить эффективность бизнес-процессов после внедрения КИС, необходима система метрики, т.е. качество следует оценивать количественно.

BPwin предоставляет аналитику два инструмента для оценки модели предприятия – стоимостной анализ, основанный на работах (Activity Based Costing, ABC), и свойства, определяемые пользователем (User Defined Properties, UDP). ABC является широко распространенной методикой, используемой международными корпорациями и государственными организациями для идентификации движителей затрат в организации. Стоимостной анализ представляет собой соглашение об учете, используемое для сбора затрат, связанных с работами, с целью определить общую стоимость процесса. ABC основан на модели работ, поскольку количественная оценка невозможна без детального понимания функциональности предприятия. Обычно ABC применяется для того, чтобы понять, как складываются выходные затраты, и облегчить выбор нужной модели работ при реорганизации деятельности предприятия (Business Process Reengineering, BPR). С помощью стоимостного анализа можно определить действительную стоимость производства продукта и поддержки клиента, идентифицировать самые затратные работы (те, которые должны быть улучшены в первую очередь) и т.д. в каждой из моделей AS-IS и TO-BE. Следовательно, стоимостной анализ позволяет оценить последствия внедрения КИС и выяснить, приведет ли информационная система к повышению производительности и экономическому эффекту и к какому именно.

ABC может проводиться только тогда, когда модель работы является последовательной (следует синтаксическим правилам IDEF0), корректной (отражает бизнес-процесс), полной (охватывает всю рассматриваемую область) и стабильной (проходит цикл экспертизы без изменений). Эта методика включает в себя такие понятия, как объект затрат (причина, по которой работа выполняется, обычно это – основной выход работы, стоимость работ есть стоимость объектов затрат), движитель затрат (характеристики входов и управлений работы, которые влияют на то, как выполняется и как долго длится работа) и центры затрат (центры затрат можно трактовать как статьи расхода).

При проведении стоимостного анализа в BPwin сначала задаются единицы измерения времени и денег, далее описываются центры затрат (cost centers) и затем для каждой работы на диаграмме декомпозиции назначаются продолжительность (duration), частота проведения данной работы в рамках общего процесса (frequency) и суммы по каждому центру затрат, т.е. задается стоимость каждой работы по каждой статье расхода. Затраты вышестоящей работы определяются как сумма затрат дочерних работ по каждому центру затрат. Такой достаточно упрощенный принцип подсчета справедлив в случае, если работы выполняются последовательно. Если же схема выполнения более сложная, можно отказаться от подсчета и задать итоговые суммы вручную или воспользоваться специализированным средством стоимостного анализа EasyABC. Результаты стоимостного анализа наглядно представляются на специальном отчете BPwin – Activity Cost Report (ACR). ABC позволяет оценить стоимостные и временные характеристики системы. Если стоимостных показателей недостаточно, имеется возможность внесения собственных метрик – свойств, определенных пользователем (UDP). Каждой работе можно поставить в соответствие набор UDP и проанализировать результат в специальном отчете Diagram Object Report.

## 2 ОСНОВЫ МЕТОДОЛОГИИ ПРОЕКТИРОВАНИЯ ИС

Информационные системы могут принести огромную пользу для предприятий (корпораций) за счет автоматизации задач, которые решались вручную. Если говорить коротко, то преимущества информационных систем сводятся к следующим ключевым понятиям: быстрее, лучше и больше. Тем не менее, для того, чтобы осознать пользу информационных систем, необходимо иметь возможность разрабатывать их вовремя и с минимальными затратами. Другими словами, информационные системы должны удовлетворять интересам бизнеса, а также быть легко модифицируемыми и недорогими. Плохо спроектированная система, в конечном счете, требует больших затрат и времени для ее содержания и обновления.

## 2.1 ЖИЗНЕННЫЙ ЦИКЛ ПО ИС

Одним из базовых понятий методологии проектирования ИС является понятие жизненного цикла ее программного обеспечения (ЖЦ ПО). ЖЦ ПО – это непрерывный процесс, который начинается с момента принятия решения о необходимости его создания и заканчивается в момент его полного изъятия из эксплуатации.

Основным нормативным документом, регламентирующим ЖЦ ПО, является международный стандарт ISO/IEC 12207 (ISO – International Organization of Standardization – Международная организация по стандартизации; IEC – International Electrotechnical Commission – Международная комиссия по электротехнике). Он определяет структуру ЖЦ, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания ПО.

Структура ЖЦ ПО по стандарту ISO/IEC 12207 базируется на трех группах процессов:

- 1) основные процессы ЖЦ ПО (приобретение, поставка, разработка, эксплуатация, сопровождение);
- 2) вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, решение проблем);
- 3) организационные процессы (управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого ЖЦ, обучение).

Разработка включает в себя все работы по созданию ПО и его компонент в соответствии с заданными требованиями, а также оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для проверки работоспособности и соответствующего качества программных продуктов, материалов, необходимых для организации обучения персонала и т.д. Разработка ПО включает в себя, как правило, анализ, проектирование и реализацию (программирование).

Эксплуатация включает в себя работы по внедрению компонентов ПО в эксплуатацию, в том числе конфигурирование базы данных и рабочих мест пользователей, обеспечение эксплуатационной документацией, проведение обучения персонала и т.д., и непосредственно эксплуатацию, в том числе локализацию проблем и устранение причин их возникновения, модификацию ПО в рамках установленного регламента, подготовку предложений по совершенствованию, развитию и модернизации системы.

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 выбор методов и инструментальных средств для реализации проекта, определение методов описания промежуточных состояний разработки, разработку методов и средств испытаний ПО, обучение персонала и т.п. Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования ПО. Верификация – это процесс определения того, отвечает ли текущее состояние разработки, достигнутое на данном этапе, требованиям этого этапа. Проверка позволяет оценить соответствие параметров разработки исходным требованиям. Проверка частично совпадает с тестированием, которое связано с идентификацией различий между действительными и ожидаемыми результатами и оценкой соответствия характеристик ПО исходным требованиям. В процессе реализации проекта важное место занимают вопросы идентификации, описания и контроля конфигурации отдельных компонентов и всей системы в целом.

Управление конфигурацией является одним из вспомогательных процессов, поддерживающих основные процессы жизненного цикла ПО, прежде всего процессы разработки и сопровождения ПО. При создании проектов сложных ИС, состоящих из многих компонентов, каждый из которых может иметь разновидности или версии, возникают проблемы учета их связей и функций, создания унифицированной структуры и обеспечения развития всей системы. Управление конфигурацией позволяет организовать, систематически учитывать и контролировать внесение изменений в ПО на всех стадиях ЖЦ. Общие принципы и рекомендации конфигурационного учета, планирования и управления конфигурациями ПО отражены в проекте стандарта ISO 12207-2.

Каждый процесс характеризуется определенными задачами и методами их решения, исходными данными, полученными на предыдущем этапе, и результатами. Результатами анализа, в частности, являются функциональные модели, информационные модели и соответствующие им диаграммы. ЖЦ ПО носит итерационный характер: результаты очередного этапа часто вызывают изменения в проектных решениях, выработанных на более ранних этапах.

## 2.2 МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ПО

Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ и методы разработки ПО (под моделью ЖЦ понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач, выполняемых на протяжении ЖЦ. Модель ЖЦ зависит от специфики ИС и условий, в которых последняя создается и функционирует). Его регламенты являются общими для любых моделей ЖЦ, методологий и технологий разработки. Стандарт ISO/IEC 12207 описывает структуру процессов ЖЦ ПО, но не конкретизирует в деталях, как реализовать или выполнить действия и задачи, включенные в эти процессы.

К настоящему времени наибольшее распространение получили следующие две основные модели ЖЦ: каскадная и спиральная.

В изначально существовавших однородных ИС каждое приложение представляло собой единое целое. Для разработки такого типа приложений применялся каскадный способ. Его основной характеристикой является разбиение всей разработки на этапы, причем переход с одного этапа на следующий происходит только после того, как будет полностью завершена работа на текущем (рис. 2.1). Каждый этап завершается выпуском полно-

го комплекта документации, достаточной для того, чтобы разработка могла быть продолжена другой командой разработчиков.

Положительные стороны применения каскадного подхода заключаются в следующем:

- на каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности;
- выполняемые в логичной последовательности этапы работ позволяют планировать сроки завершения всех работ и соответствующие затраты.

Каскадный подход хорошо зарекомендовал себя при построении ИС, для которых в самом начале разработки можно достаточно точно и полно сформулировать все требования, с тем чтобы предоставить разработчикам свободу реализовать их как можно лучше с технической точки зрения. В эту категорию попадают сложные расчетные системы, системы реального времени и другие подобные задачи. Однако в процессе использования этого подхода обнаружился ряд его недостатков,

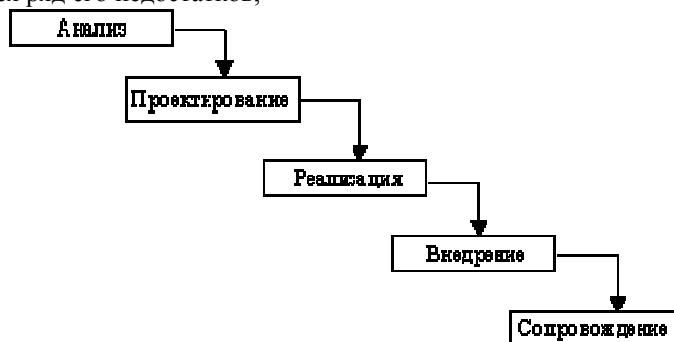


Рис. 2.1 Каскадная схема разработки ПО

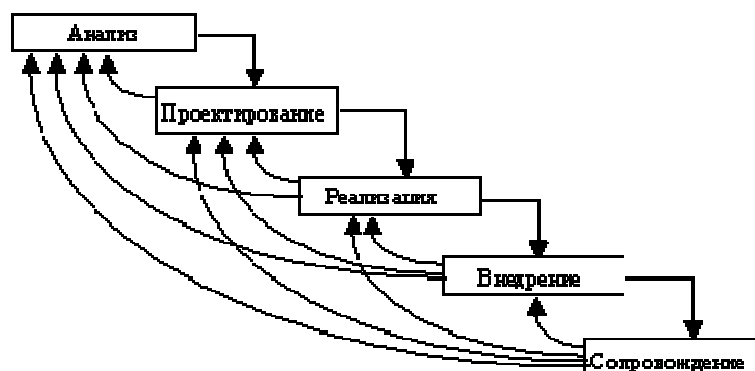


Рис. 2.2 Реальный процесс разработки ПО по каскадной схеме

вызванных прежде всего тем, что реальный процесс создания ПО никогда полностью не укладывался в такую жесткую схему. В процессе создания ПО постоянно возникала потребность в возврате к предыдущим этапам и уточнении или пересмотре ранее принятых решений. В результате реальный процесс создания ПО принимал вид, представленный на рис. 2.2.

Основным недостатком каскадного подхода является существенное запаздывание с получением результатов. Согласование результатов с пользователями производится только в точках, планируемых после завершения каждого этапа работ, требования к ИС «заморожены» в виде технического задания на все время ее создания. Таким образом, пользователи могут внести свои замечания только после того, как работа над системой будет полностью завершена. В случае неточного изложения требований или их изменения в течение длительного периода создания ПО пользователи получают систему, не удовлетворяющую их потребностям. Модели (как функциональные, так и информационные) автоматизируемого объекта могут устареть одновременно с их утверждением.

Для преодоления перечисленных проблем была предложена спиральная модель ЖЦ (рис. 2.3), делающая упор на начальные этапы ЖЦ: анализ и проектирование. На этих этапах реализуемость технических решений проверяется путем создания прототипов. Каждый виток спирали соответствует созданию фрагмента или версии ПО, на нем уточняются цели и характеристики проекта, определяется его качество и планируются работы следующего витка спирали. Таким образом углубляются и последовательно конкретизируются детали проекта и в результате выбирается обоснованный вариант, который доводится до реализации.

Разработка итерациями отражает объективно существующий спиральный цикл создания системы. Неполное завершение работ на каждом этапе позволяет переходить на следующий этап, не дожидаясь полного завершения работы на текущем. При итеративном способе разработки недостающую работу можно будет выполнить

на следующей итерации. Главная же задача – как можно быстрее показать пользователям системы работоспособный продукт, тем самым активизируя процесс уточнения и дополнения требований.

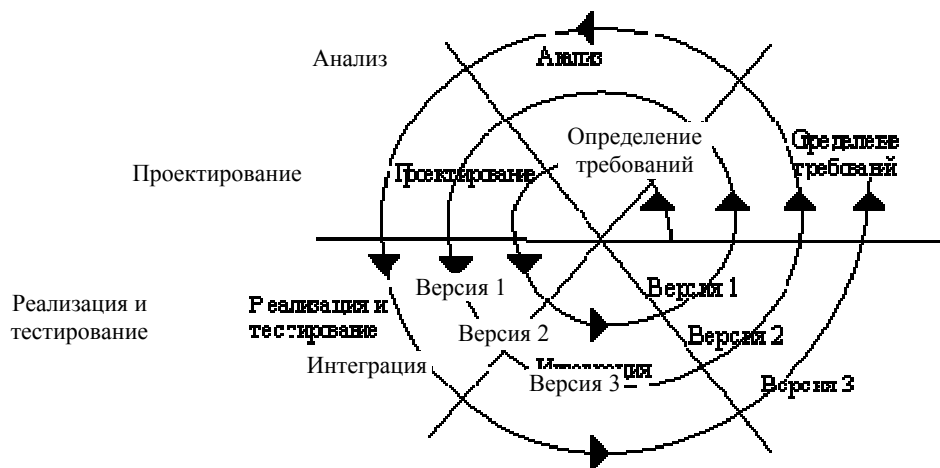


Рис. 2.3 Спиральная модель ЖЦ

Основная проблема спирального цикла – определение момента перехода на следующий этап. Для ее решения необходимо ввести временные ограничения на каждый из этапов жизненного цикла. Переход осуществляется в соответствии с планом, даже если не вся запланированная работа закончена. План составляется на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков.

### 2.3 МЕТОДОЛОГИИ И ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ИС (CASE-СРЕДСТВА)

Методологии, технологии и инструментальные средства проектирования (CASE-средства) составляют основу проекта любой ИС. Методология реализуется через конкретные технологии и поддерживающие их стандарты, методики и инструментальные средства, которые обеспечивают выполнение процессов ЖЦ.

Технология проектирования определяется как совокупность трех составляющих:

- пошаговой процедуры, определяющей последовательность технологических операций проектирования (рис. 2.4);

- критериев и правил, используемых для оценки результатов выполнения технологических операций;
- нотаций (графических и текстовых средств), используемых для описания проектируемой системы.

Технологические инструкции, составляющие основное содержание технологии, должны состоять из описания последовательности технологических операций, условий, в зависимости от которых выполняется та или иная операция, и описаний самих операций.

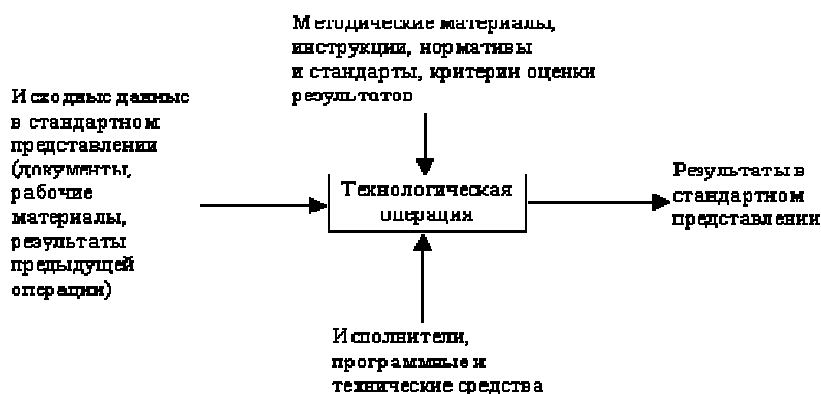


Рис. 2.4 Представление технологической операции проектирования

Технология проектирования, разработки и сопровождения ИС должна удовлетворять следующим общим требованиям:

- поддерживать полный ЖЦ ПО;
- гарантировать достижение целей разработки ИС с заданным качеством и в установленное время;
- обеспечивать возможность выполнения крупных проектов в виде подсистем (т.е. возможность декомпозиции проекта на составные части, разрабатываемые группами исполнителей ограниченной численности с последующей интеграцией составных частей);
- обеспечивать возможность ведения работ по проектированию отдельных подсистем небольшими группами (3 – 7 человек). Это обусловлено принципами управляемости коллектива и повышения производительности за счет минимизации числа внешних связей;

- обеспечивать минимальное время получения работоспособной ИС. Речь идет не о сроках готовности всей ИС, а о сроках реализации отдельных подсистем. Практика показывает, что даже при наличии полностью завершеного проекта внедрение идет последовательно по отдельным подсистемам;

- предусматривать возможность управления конфигурацией проекта, ведения версий проекта и его составляющих, возможность автоматического выпуска проектной документации и синхронизацию ее версий с версиями проекта;

- обеспечивать независимость выполняемых проектных решений от средств реализации ИС (систем управления базами данных (СУБД), операционных систем, языков и систем программирования);

- поддерживаться комплексом согласованных CASE-средств, обеспечивающих автоматизацию процессов, выполняемых на всех стадиях ЖЦ.

Реальное применение любой технологии проектирования, разработки и сопровождения ИС в конкретной организации и конкретном проекте невозможно без выработки ряда стандартов (правил, соглашений), которые должны соблюдаться всеми участниками проекта. К таким стандартам относятся следующие:

- стандарт проектирования, который должен устанавливать: набор необходимых моделей (диаграмм) на каждой стадии проектирования и степень их детализации; правила фиксации проектных решений на диаграммах, в том числе: правила именования объектов (включая соглашения по терминологии), набор атрибутов для всех объектов и правила их заполнения на каждой стадии, правила оформления диаграмм, включая требования к форме и размерам объектов, и т.д.; требования к конфигурации рабочих мест разработчиков, включая настройки операционной системы, настройки CASE-средств, общие настройки проекта и т.д.; механизм обеспечения совместной работы над проектом, в том числе: правила интеграции подсистем проекта, правила поддержания проекта в одинаковом для всех разработчиков состоянии (регламент обмена проектной информацией, механизм фиксации общих объектов и т.д.), правила проверки проектных решений на непротиворечивость и т.д.

- стандарт оформления проектной документации, который должен устанавливать: комплектность, состав и структуру документации на каждой стадии проектирования; требования к ее оформлению (включая требования к содержанию разделов, подразделов, пунктов, таблиц и т.д.); правила подготовки, рассмотрения, согласования и утверждения документации с указанием предельных сроков для каждой стадии; требования к настройке издательской системы, используемой в качестве встроенного средства подготовки документации; требования к настройке CASE-средств для обеспечения подготовки документации в соответствии с установленными требованиями.

- стандарт пользовательского интерфейса, который должен устанавливать: правила оформления экранов (шрифты и цветовая палитра), состав и расположение окон и элементов управления; правила использования клавиатуры и мыши; правила оформления текстов помощи; перечень стандартных сообщений; правила обработки реакции пользователя.

## 2.4 СТРУКТУРНЫЙ ПОДХОД К ПРОЕКТИРОВАНИЮ ИС

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 так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимозавязаны. При разработке системы «снизу-вверх» от отдельных задач ко всей системе целостность теряется, возникают проблемы при информационной стыковке отдельных компонентов.

Все наиболее распространенные методологии структурного подхода базируются на ряде общих принципов. Основными из этих принципов являются следующие:

- принцип абстрагирования – заключается в выделении существенных аспектов системы и отвлечении от несущественных;

- принцип формализации – заключается в необходимости строгого методического подхода к решению проблемы;

- принцип непротиворечивости – заключается в обоснованности и согласованности элементов;

- принцип структурирования данных – заключается в том, что данные должны быть структурированы и иерархически организованы.

В структурном анализе используются в основном две 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)-модели и соответствующие функциональные диаграммы (подраздел 2.4.1);

- DFD (Data Flow Diagrams)-диаграммы потоков данных (подраздел 2.4.2);

- ERD (Entity-Relationship Diagrams)-диаграммы «сущность-связь» (подраздел 2.4.3).

На стадии проектирования ИС модели расширяются, уточняются и дополняются диаграммами, отражающими структуру программного обеспечения: архитектуру ПО, структурные схемы программ и диаграммы экранов форм.

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.

#### 2.4.1 Моделирование потоков данных (процессов) DFD

В соответствии с методологией модель системы определяется как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи пользователю. Диаграммы верхних уровней иерархии (контекстные диаграммы) определяют основные процессы или подсистемы ИС с внешними входами и выходами. Они детализируются при помощи диаграмм нижнего уровня. Такая декомпозиция продолжается, создавая многоуровневую иерархию диаграмм, до тех пор, пока не будет достигнут такой уровень декомпозиции, на котором процессы становятся элементарными и детализировать их далее невозможно.

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

Основными компонентами диаграмм потоков данных являются: внешние сущности, процессы, системы/подсистемы, накопители данных, потоки данных.

*Внешняя сущность* обозначается квадратом (рис. 2.5), расположенным как бы «над» диаграммой и бросающим на нее тень, для того, чтобы можно было выделить этот символ среди других обозначений.

*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 выпуск отчетов, программа, аппаратно реализованное логическое устройство и т.д.

Процесс на диаграмме потоков данных изображается, как показано на рис. 2.6.

Номер процесса служит для его идентификации. В поле имени вводится наименование процесса в виде предложения с активным недвусмысленным глаголом в неопределенной форме (вычислить, рассчитать, проверить, определить, создать, получить), за которым следуют существительные в винительном падеже, например: ввести сведения о клиентах; выдать информацию о текущих расходах. Использование таких глаголов, как «обработать», «модернизировать» или «отредактировать» означает, как правило, недостаточно глубокое понимание данного процесса и требует дальнейшего анализа.



Рис. 2.5 Внешняя сущность

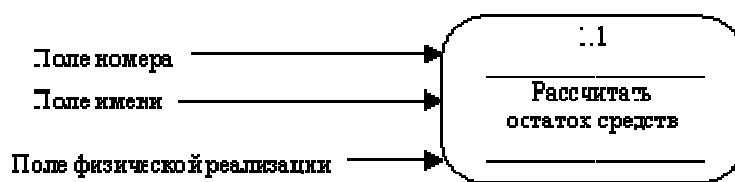


Рис. 2.6 Процесс

Информация в поле физической реализации показывает, какое подразделение организации, программа или аппаратное устройство выполняет данный процесс.

При построении модели сложной ИС она может быть представлена в самом общем виде на так называемой контекстной диаграмме в виде одной системы как единого целого либо может быть декомпозирована на ряд подсистем.

*Подсистема* (или система) на контекстной диаграмме изображается, как показано на рис. 2.7.

Номер подсистемы служит для ее идентификации. В поле имени вводится наименование подсистемы в виде предложения с подлежащим и соответствующими определениями и дополнениями.

*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 микрофиши, ящика в картотеке, таблицы в оперативной памяти, файла на магнитном носителе и т.д. Накопитель данных на диаграмме потоков данных изображается, как показано на рис. 2.8.

Накопитель данных идентифицируется буквой «D» и произвольным числом. Имя накопителя выбирается из соображения наибольшей информативности для проектировщика.

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

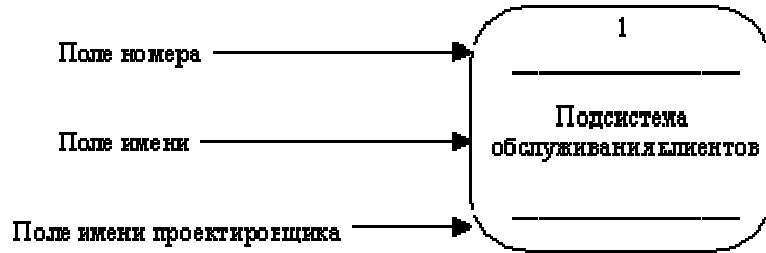


Рис. 2.7 Подсистема

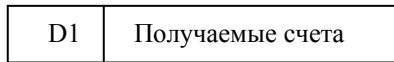


Рис. 2.8 Накопитель данных

*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.

Поток данных на диаграмме изображается линией, оканчивающейся стрелкой, которая показывает направление потока (рис. 2.9). Каждый поток данных имеет имя, отражающее его содержание.

Первым шагом при построении иерархии DFD является построение контекстных диаграмм. Обычно при проектировании относительно простых ИС строится единственная контекстная диаграмма со звездообразной топологией, в центре которой находится так называемый главный процесс, соединенный с приемниками и источниками информации, посредством которых с системой взаимодействуют пользователи и другие внешние системы.

Если же для сложной системы ограничиться единственной контекстной диаграммой, то она будет содержать слишком большое количество источников и приемников информации, которые трудно расположить на листе бумаги нормального формата, и кроме того, единственный главный процесс не раскрывает структуры распределенной системы. Признаками сложности (в смысле контекста) могут быть:

- наличие большого количества внешних сущностей (десять и более);
- распределенная природа системы;
- многофункциональность системы с уже сложившейся или выявленной группировкой функций в отдельные подсистемы.

Для сложных ИС строится иерархия контекстных диаграмм. При этом контекстная диаграмма верхнего уровня содержит не единственный главный процесс, а набор подсистем, соединенных потоками данных. Контекстные диаграммы следующего уровня детализируют контекст и структуру подсистем.

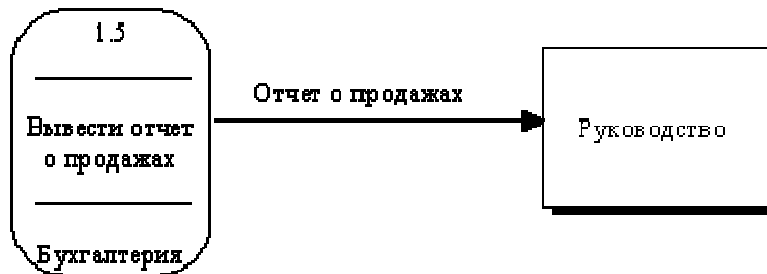


Рис. 2.9 Поток данных

Иерархия контекстных диаграмм определяет взаимодействие основных функциональных подсистем проектируемой ИС как между собой, так и с внешними входными и выходными потоками данных и внешними объектами (источниками и приемниками информации), с которыми взаимодействует ИС.

Разработка контекстных диаграмм решает проблему строгого определения функциональной структуры ИС на самой ранней стадии ее проектирования, что особенно важно для сложных многофункциональных систем, в разработке которых участвуют разные организации и коллективы разработчиков.

После построения контекстных диаграмм полученную модель следует проверить на полноту исходных данных об объектах системы и изолированность объектов (отсутствие информационных связей с другими объектами).



Для каждой подсистемы, присутствующей на контекстных диаграммах, выполняется ее детализация при помощи DFD. Каждый процесс на DFD, в свою очередь, может быть детализирован при помощи DFD или миниспецификации.

Миниспецификация (описание логики процесса) должна формулировать его основные функции таким образом, чтобы в дальнейшем специалист, выполняющий реализацию проекта, смог выполнить их или разработать соответствующую программу.

При построении иерархии DFD переходить к детализации процессов следует только после определения содержания всех потоков и накопителей данных, которое описывается при помощи структур данных. Структуры данных конструируются из элементов данных и могут содержать альтернативы, условные вхождения и итерации. Условное вхождение означает, что данный компонент может отсутствовать в структуре. Альтернатива означает, что в структуру может входить один из перечисленных элементов. Итерация означает вхождение любого числа элементов в указанном диапазоне. Для каждого элемента данных может указываться его тип (непрерывные или дискретные данные). Для непрерывных данных может указываться единица измерения (кг, см и т.п.), диапазон значений, точность представления и форма физического кодирования. Для дискретных данных может указываться таблица допустимых значений.

После построения законченной модели системы ее необходимо верифицировать (проверить на полноту и согласованность). В полной модели все ее объекты (подсистемы, процессы, потоки данных) должны быть подробно описаны и детализированы. Выявленные недетализированные объекты следует детализировать, вернувшись на предыдущие шаги разработки. В согласованной модели для всех потоков данных и накопителей данных должно выполняться правило сохранения информации: все поступающие куда-либо данные должны быть считаны, а все считываемые данные должны быть записаны.

#### 2.4.2 CASE-метод Баркера (ERD)

Цель моделирования данных состоит в обеспечении разработчика ИС концептуальной схемой базы данных в форме одной модели или нескольких локальных моделей, которые относительно легко могут быть отображены в любую систему баз данных.

Наиболее распространенным средством моделирования данных являются диаграммы «сущность-связь» (ERD). С их помощью определяются важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи). ERD непосредственно используются для проектирования реляционных баз данных.

Нотация ERD была впервые введена П. Ченом и получила дальнейшее развитие в работах Баркера. Метод Баркера будет излагаться на примере моделирования деятельности предприятия по торговле автомобилями

Главный менеджер: одна из основных обязанностей – содержание автомобильного имущества. Он должен знать, сколько заплачено за машины и как велики накладные расходы. Обладая этой информацией, он может установить нижнюю цену, за которую мог бы продать данный экземпляр. Кроме того, он несет ответственность за продавцов и ему нужно знать, кто что продает и сколько машин продал каждый из них.

Продавец: ему нужно знать, какую цену запрашивать и какова нижняя цена, за которую можно совершить сделку. Кроме того, ему нужна основная информация о машинах: год выпуска, марка, модель и т.п.

Администратор: его задача сводится к составлению контрактов, для чего нужна информация о покупателе, автомашине и продавце, поскольку именно контракты приносят продавцам вознаграждения за продажи.

Первый шаг моделирования – извлечение информации из интервью с персоналом предприятия и выделение сущностей.

*Сущность* (Entity) – реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области, информация о котором подлежит хранению (рис. 2.10).

Каждая сущность должна обладать уникальным идентификатором. Каждый экземпляр сущности должен однозначно идентифицироваться и отличаться от всех других экземпляров данного типа сущности. Каждая сущность обладает некоторыми свойствами:

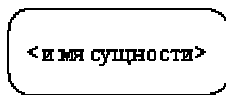


Рис. 2.10 Графическое изображение сущности

- каждая сущность должна иметь уникальное имя, и к одному и тому же имени должна всегда применяться одна и та же интерпретация. Одна и та же интерпретация не может применяться к различным именам, если только они не являются псевдонимами;

- сущность обладает одним или несколькими атрибутами, которые либо принадлежат сущности, либо наследуются через связь;

- сущность обладает одним или несколькими атрибутами, которые однозначно идентифицируют каждый экземпляр сущности;

- каждая сущность может обладать любым количеством связей с другими сущностями модели.

Сущности, которые могут быть идентифицированы с главным менеджером – это автомашины и продавцы. Продавцу важны автомашины и связанные с их продажей данные. Для администратора важны покупатели, автомашины, продавцы и контракты. Исходя из этого, выделяются 4 сущности (автомашина, продавец, покупатель, контракт), которые изображаются на диаграмме следующим образом (рис. 2.11).

Следующим шагом моделирования является идентификация связей.

**Связь (Relationship)** – поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области. Связь – это ассоциация между сущностями, при которой, как правило, каждый экземпляр одной сущности, называемой родительской сущностью, ассоциирован с произвольным (в том числе нулевым) количеством экземпляров второй сущности, называемой сущностью-потомком, а каждый экземпляр сущности-потомка ассоциирован в точности с одним экземпляром сущности-родителя. Таким образом, экземпляр сущности-потомка может существовать только при существовании сущности родителя.

Связи может даваться имя, выражаемое грамматическим оборотом глагола и помещаемое возле линии связи. Имя каждой связи между двумя данными сущностями должно быть уникальным, но имена связей в модели не обязаны быть уникальными. Имя связи всегда формируется с точки зрения родителя, так что предложение может быть образовано соединением имени сущности-родителя, имени связи, выражения степени и имени сущности-потомка.

Например, связь продавца с контрактом может быть выражена следующим образом:

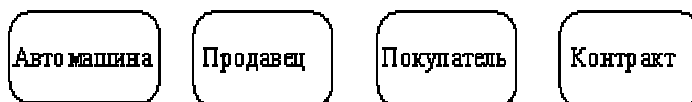


Рис. 2.11 Сущности

- продавец может получить вознаграждение за один или более контрактов;
- контракт должен быть инициирован ровно одним продавцом.

Степень связи и обязательность графически изображаются так, как показано на рис. 2.12.

Таким образом, два предложения, описывающие связь продавца с контрактом, графически будут выражены следующим образом (рис. 2.13).

Описав также связи остальных сущностей, получим схему на рис. 2.14.

Последним шагом моделирования является идентификация атрибутов.

**Атрибут** – любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут представляет тип характеристик или свойств, ассоциированных с множеством реальных или абстрактных объектов (людей, мест, событий, состояний, идей, пар

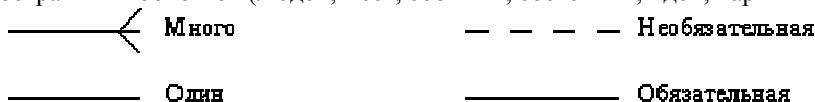


Рис. 2.12 Степень связи

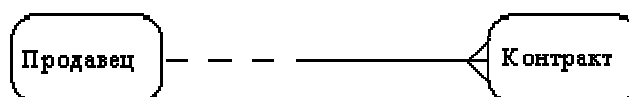


Рис. 2.13 Связь продавца с контрактом

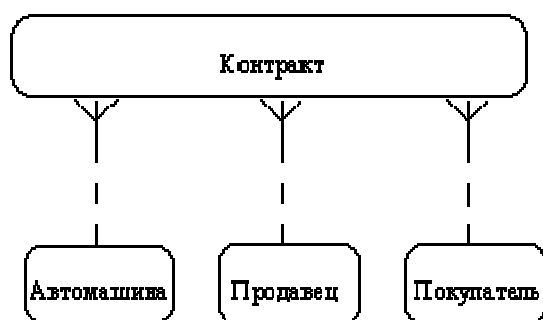


Рис. 2.14 Связь сущностей

предметов и т.д.). Экземпляр атрибута – это определенная характеристика отдельного элемента множества. Экземпляр атрибута определяется типом характеристики и ее значением, называемым значением атрибута. В ERD-модели атрибуты ассоциируются с конкретными сущностями. Таким образом, экземпляр сущности должен обладать единственным определенным значением для ассоциированного атрибута.

Атрибут может быть либо обязательным, либо необязательным (рис. 2.15). Обязательность означает, что атрибут не может принимать неопределенных значений (null values). Атрибут может быть либо описательным (т.е. обычным дескриптором сущности), либо входит в состав уникального идентификатора (первичного ключа).

**Уникальный идентификатор** – это атрибут или совокупность атрибутов и/или связей, предназначенная для уникальной идентификации каждого экземпляра данного типа сущности. В случае полной идентификации каждый экземпляр данного типа сущности полностью идентифицируется своими собственными ключевыми

атрибутами, в противном случае в его идентификации участвуют также атрибуты другой сущности-родителя (рис. 2.16).

Каждый атрибут идентифицируется уникальным именем, выражаемым грамматическим оборотом существительного, описывающим представляемую атрибутом характеристику. Атрибуты изображаются в виде списка имен внутри блока ассоциированной сущности, причем каждый атрибут занимает отдельную строку. Атрибуты, определяющие первичный ключ, размещаются наверху списка и выделяются знаком «#».

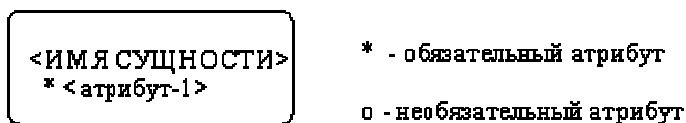


Рис. 2.15 Атрибут

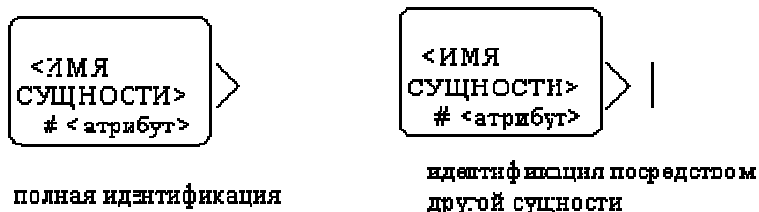


Рис. 2.16 Уникальный идентификатор

Каждая сущность должна обладать хотя бы одним возможным ключом. Возможный ключ сущности – это один или несколько атрибутов, чьи значения однозначно определяют каждый экземпляр сущности. При существовании нескольких возможных ключей один из них обозначается в качестве первичного ключа, а остальные – как альтернативные ключи.

С учетом имеющейся информации дополним построенную ранее диаграмму (рис. 2.17).

Помимо перечисленных основных конструкций модель данных может содержать ряд дополнительных.

*Подтипы и супертипы:* одна сущность является обобщающим понятием для группы подобных сущностей (рис. 2.18).

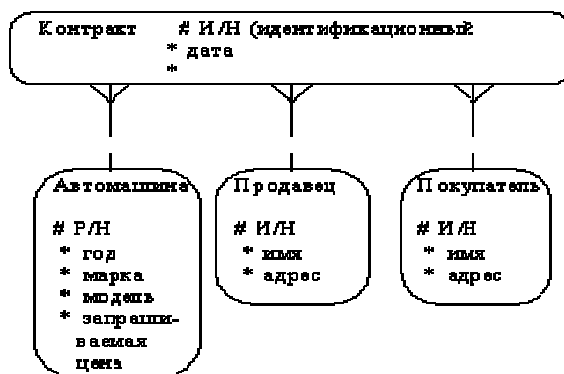


Рис. 2.17 Связь сущностей

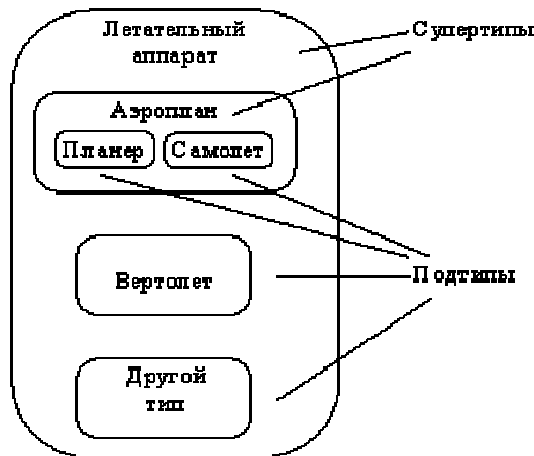


Рис. 2.18 Подтипы и супертипы

*Взаимно исключающие связи:* каждый экземпляр сущности участвует только в одной связи из группы взаимно исключающих связей (рис. 2.19).

*Рекурсивная связь*: сущность может быть связана сама с собой (рис. 2.20).

*Неперемещаемые (non-transferrable) связи*: экземпляр сущности не может быть перенесен из одного экземпляра связи в другой (рис. 2.21).

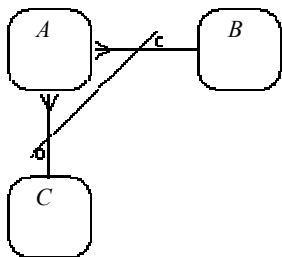


Рис. 2.19 Взаимно исключающие связи

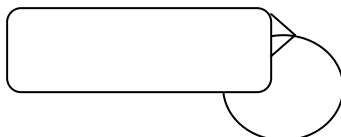


Рис. 2.20 Рекурсивная связь

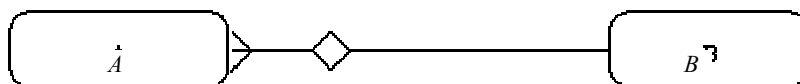


Рис. 2.21 Неperемещаемая связь

### 2.4.3 CASE-СРЕДСТВО ERWIN

ERWIN – СРЕДСТВО РАЗРАБОТКИ СТРУКТУРЫ БАЗЫ ДАННЫХ (БД). ERWIN СОЧЕТАЕТ ГРАФИЧЕСКИЙ ИНТЕРФЕЙС WINDOWS, ИНСТРУМЕНТЫ ДЛЯ ПОСТРОЕНИЯ ER-ДИАГРАММ, РЕДАКТОРЫ ДЛЯ СОЗДАНИЯ ЛОГИЧЕСКОГО И ФИЗИЧЕСКОГО ОПИСАНИЯ МОДЕЛИ ДАННЫХ И ПРОЗРАЧНУЮ ПОДДЕРЖКУ ВЕДУЩИХ РЕЛЯЦИОННЫХ СУБД И НАСТОЛЬНЫХ БАЗ ДАННЫХ. С ПОМОЩЬЮ ERWIN МОЖНО СОЗДАВАТЬ ИЛИ ПРОВОДИТЬ ОБРАТНОЕ ПРОЕКТИРОВАНИЕ (РЕИНЖИНИРИНГ) БАЗ ДАННЫХ.

Реализация моделирования в ERwin базируется на теории реляционных баз данных и на методологии IDEF1X.

Рассмотрим пример разработки модели в ERwin для следующей задачи: ведется учет служащих; для каждого служащего хранится информация о детях и о списке должностей, занимавшихся этим служащим; для должностей хранится информация по установленным должностным окладам.

Сначала создается логический уровень модели. Для этого задается режим отображения сущностей (Display/Entity Level). Создаются при помощи линейки инструментов сущности «служащий», «дети», «история работы», «история зарплаты».

Выбрав каждую сущность, зададим для нее подробное описание на русском языке в редакторе «Entity Definition». Это описание появится в отчетах ERwin и может быть отображено на диаграмме.

Укажем связи между сущностями. Например, «служащий» связан идентифицирующей связью, «является родителем», с сущностью «дети». Описание связи вводится в редакторе «Editor/Relationship» (диаграмме ERwin, рис. 2.22).

Теперь перейдем в режим задания атрибутов (Display/Attribute Level). В редакторе «Entity/Attribute» задаются на русском языке имена ключевых и неключевых атрибутов. Заметим, что для дочерней сущности «дети» ключевой атрибут «номер служащего» не указывается вручную. ERwin обеспечивает его миграцию из родительской сущности. То же происходит с другими дочерними сущностями.

Для атрибута «имя» сущности «служащий» укажем, что он является альтернативным ключом (будем считать, что у всех служащих уникальные имена/фамилии). Для этого после имени атрибута поместим указатель АК1 в скобках (диаграмме ERwin, рис. 2.23).

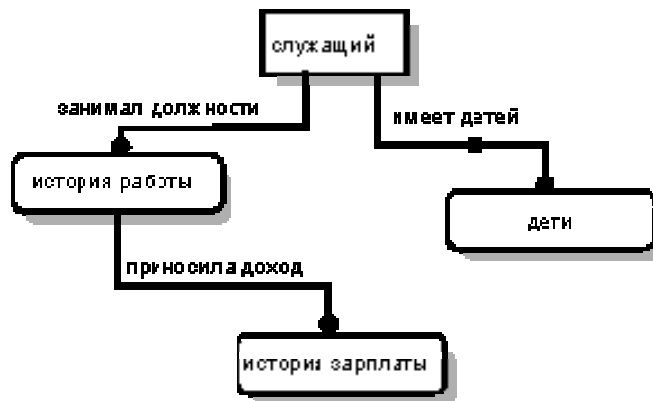


Рис. 2.22 Диаграмма уровня сущности служащий



Рис. 2.23 Диаграмма уровня атрибутов

Так как имена атрибутов и сущностей задавались на русском языке, для перехода к физическому уровню модели следует поставить им в соответствие идентификаторы таблиц, колонок и ограничений, удовлетворяющие правилам целевой СУБД (обычно это означает использование латинских букв, цифр и некоторых специальных символов).

В редакторе «Database Schema» указывается для каждой сущности соответствующее имя таблицы. Затем в редакторе «Attribute Definition» задаются имена колонок таблиц, соответствующие атрибутам сущностей. ERwin и здесь обеспечивает миграцию имен колонок в подчиненные таблицы.

На этом этапе можно воспользоваться редактором «Extended Attributes» для определения расширенных атрибутов PowerBuilder (формата отображения, маски редактирования, правила контроля, выравнивания, заголовков и комментариев).

В редакторе «Relationship Definitions» указывается физическое имя связи, которое соответствует имени ограничения (constraint), создаваемого ERwin в базе данных.

Теперь все готово к созданию БД и нужно выбрать целевую СУБД (если этого не было сделано раньше). Выбирается, например, Sybase System 10.

В редакторе SYBASE Database Schema задаются типы данных для колонок таблиц.

Диалог, в котором происходит выбор типа данных, приведен на рис. 2.24.

Теперь можно перейти к созданию базы данных. Для этого выполняется команда «Sybase schema generation». ERwin построит пакет SQL-предложений генерации базы данных.

Обратное проектирование, т.е. восстановление информационной модели по существующей базе данных, используется при выборе оптимальной платформы (rightsizing) для существующей настольной (desktop) базы данных или базы данных на mainframe, а также при расширении (или модификации) существующей структуры, которая была построена без необходимой сопроводительной документации. После завершения процесса восстановления модели ERwin автоматически «раскладывает» таблицы на диаграмме. Теперь можно выполнять модификации уже с использованием логической схемы – добавлять сущности, атрибуты, комментарии, связи и т.д. По завершении изменений одна команда – синхронизировать модель с базой данных – актуализирует все проведенные изменения.

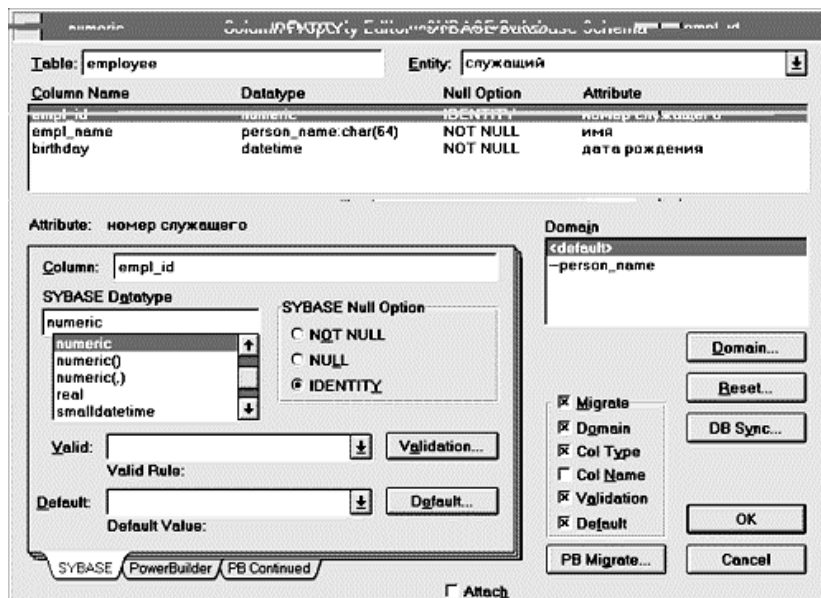


Рис. 2.24 Определение физической модели

Построение модели может быть выполнено как на основании данных каталога базы данных, так и на основании пакета операторов SQL, с помощью которого была создана база данных.

В процессе разработки информационной системы может возникнуть ситуация, когда структура базы данных и информационная модель не соответствуют друг другу. ERwin предоставляет возможность привести их в соответствие.

Для этого предусмотрена функция синхронизации с базой данных. После подключения к СУБД предлагается список несоответствий между существующей структурой данных и моделью. Например, если в базе данных создана новая таблица, то ERwin предложит провести включение ее в модель. Если в модель добавлена новая таблица, ERwin предложит создать ее в реальной базе данных. Аналогично, при добавлении колонок в базу данных или в модели ERwin предлагается провести соответствующие операции по синхронизации.

ERwin «знает» о таких особенностях хранения данных в отдельных СУБД, как сегменты (в Sybase) и табличное пространство (в Oracle). Информация о физическом размещении может быть включена в модель и использована при прямом и обратном проектировании.

ERwin поддерживает прямой интерфейс с основными СУБД: DB2, Informix, Ingres, NetWare SQL, ORACLE, Progress, Rdb, SQL/400, SQLBase, SQL Server, Sybase System 10, Watcom SQL.

ERwin поддерживает также настольные (desktop) СУБД: Microsoft Access, FoxPro, Clipper, dBASE III, dBASE IV и Paradox.

Отметим, что поддерживаются как самые современные, так и предыдущие версии основных СУБД. Например, для Sybase поддерживается System 10 и версия 4.2; для Oracle – 7 и 6; SQLBase – 6 и 5; Watcom – 4 и 3; Informix – 6 и 5; Access – 2.0 и 1.1.

Проектирование на физическом уровне выполняется в терминах той базы данных, которую предполагается использовать в системе. Важно, что ERwin «известны» соответствия между возможностями СУБД различных производителей, вследствие чего возможна конвертация физической схемы, спроектированной для одной СУБД, в другую. Например, если при описании ссылочной целостности указана опция «on delete cascade», а СУБД не поддерживает такой режим, ERwin сгенерирует соответствующий триггер.

Для создания физической структуры БД может быть запрошена генерация DDL-скрипта (data definition language). При этом используется диалект SQL для выбранного типа и версии сервера. Хотя сгенерированный код не нуждается в модификации, имеется возможность его сохранить в файл или распечатать.

ERwin реализует собственный макроязык для подготовки прототипов триггеров и процедур. Схема использования прототипов заключается в подготовке шаблона для различных типов триггеров (например, триггер, реализующий логику каскадного удаления – ON DELETE CASCADE). Базовые шаблоны встроены в ERwin, но пользователь может определить свои собственные шаблоны и использовать их вместо стандартных.

Все макрофункции, которые могут применяться в триггерах, могут использоваться также и в процедурах. Существенно, что процедуры, как и триггеры, связываются с таблицей.

Такой подход позволяет полностью исключить хаотичное внесение изменений в базу данных, так как модель в ERwin описывает все аспекты базы, в том числе обеспечиваемые триггерами.

В ERwin поддерживаются два типа правил (проверок допустимости значений) и начальных (по умолчанию) значений. Правило и умолчание может быть указано для проверки со стороны клиента (например, в PowerBuilder) и со стороны сервера.

При задании правила или умолчания для клиентской части эти атрибуты переносятся в репозиторий средства 4GL.

Часто используемые комбинации свойств можно поименовать. Такая комбинация свойств, называемая доменом, может наследоваться. Например, можно определить домен «Дата» для отображения всех колонок с датами в приложении в одном стиле, домен «Дата рождения ребенка» наследует все атрибуты от домена «Дата» и вносит дополнительный атрибут – цвет отображения.

Назначение доменов для сервера аналогично назначению доменов для клиента. Различие заключается в том, что правила и начальные значения для сервера определяются в генерации схемы базы данных, а аналогичные атрибуты для клиента – сохраняются в репозитории средства 4GL.

Другое назначение доменов для сервера – определение пользовательских типов данных. Пользовательскому типу данных ставится в соответствие тип, «известный» СУБД. При выполнении синхронизации с базой данных для СУБД, поддерживающих пользовательские типы, выполняются соответствующие команды. Например, для Sybase выполняется команда:

```
sp_addtype person_name, «char(64)», «NOT NULL».
```

По завершении работы над информационной моделью, как правило, распечатываются логический и физический уровни диаграммы, а также отчет по соответствиям сущность-таблица, атрибут-имя колонки, сущность-атрибуты. Диаграмма физической модели является необходимым, почти достаточным и очень удобным материалом для разработчиков программ. Дополнительная информация для группы разработчиков прикладных программ содержится в отчете «имена таблиц и колонок», который может быть построен с помощью ERwin.

Сгенерированный отчет может быть сохранен на диск (колонки разделяются запятыми, выравниваются или разделяются табуляцией) или передан в текстовый процессор (или электронную таблицу) через интерфейс DDE.

Диаграммы информационных моделей современных информационных систем обычно весьма велики, вследствие чего работать со всей диаграммой достаточно сложно как на стадии проектирования информационной модели, так и разработчикам прикладного программного обеспечения. ERwin дает возможность работать не со всей диаграммой, а с логически законченной группой сущностей, называемой предметной областью (Subject Area). Переключение отображения с одной предметной области на другую производится выбором из раскрывающегося списка.

Например, рассмотрим информационную модель для некоторого абстрактного предприятия. В информационной системе задействованы бухгалтерия, склад, кадры. В этом рассмотренные в первом примере сущности (сотрудник, история работы, доход) могут быть выделены в отдельную предметную область «кадры».

Такой подход обладает рядом важных преимуществ. Во-первых, группа разработчиков программного обеспечения снабжается диаграммой той предметной области, с которой она работает. Во-вторых, при разработке информационной модели проектировщик может удалить с экрана уже спроектированные блоки, чтобы они не загромождали диаграмму. В-третьих, использование предметных областей стимулирует структурный подход к разработке информационной модели, т.е. выделение логических блоков с последующей их детальной разработкой.

Уровень детализации диаграммы информационной модели может изменяться проектировщиком. Например, могут отображаться только имена сущностей (таблиц), может быть включено/выключено отображение мощности связи, может быть включено/выключено отображение альтернативных ключей, может отображаться физическая или концептуальная модель. Для удобства проектировщика предусмотрена возможность присвоить имя группе параметров отображения. Определенные пользователем имена показываются на экране в виде закладок, что обеспечивает переключение с одного режима отображения на другой одним щелчком мыши.

Проектировщик информационной модели имеет возможности использовать цветное и шрифтовое выделение для различных компонентов диаграммы. Выделение может быть выполнено как для всей модели (например, все внешние ключи отображать синим цветом), так и для отдельного компонента (одна таблица, все атрибуты таблицы, один атрибут таблицы, одна связь и т.д.). Использование цветного и шрифтового выделения на диаграмме информационной модели делает ее более наглядной и позволяет проектировщику обратить внимание читателей диаграммы на ее отдельные элементы.

### **3 КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ (КИС)**

---

#### **3.1 ОПИСАНИЕ БАЗОВЫХ ПРИНЦИПОВ MRP**

В начале 60-х гг. XX в., в связи с ростом популярности вычислительных систем, возникла идея использовать их возможности для планирования деятельности предприятия, в том числе для планирования производственных процессов. Необходимость планирования обусловлена тем, что основная масса задержек в процессе производства связана с запаздыванием поступления отдельных комплектующих, в результате чего, как правило, параллельно с уменьшением эффективности производства на складах возникает избыток материалов, поступивших в срок или ранее намеченного срока. Кроме того, вследствие нарушения баланса поставок комплек-

тующих возникают дополнительные осложнения с учетом и отслеживанием их состояния в процессе производства, т.е. фактически невозможно определить, например, к какой партии принадлежит данный составляющий элемент в уже собранном готовом продукте. С целью предотвращения подобных проблем, была разработана методология планирования потребности в материалах MRP (Material Requirements Planning). Реализация КИС, работающей по этой методологии, представляет собой компьютерную программу, позволяющую оптимально регулировать поставки комплектующих в производственный процесс, контролируя запасы на складе и саму технологию производства. Главной задачей MRP является обеспечение гарантии наличия необходимого количества требуемых материалов-комплектующих в любой момент времени в рамках срока планирования наряду с возможным уменьшением постоянных запасов, а следовательно – разгрузкой склада.

Первым стандартом управления бизнесом был MPS (Master Planning Scheduling), или объемно-календарное планирование (рис. 3.1). Идея была проста – формируется план продаж («объем», с разбивкой по календарным периодам, отсюда – объемно-календарное), по нему формируется план пополнения запасов (за счет производства или закупки) и оцениваются финансовые результаты по периодам (в качестве которых используются периоды планирования или финансовые периоды).

Пока производство было мелким и простым, все получалось относительно неплохо. Но потом стали возникать проблемы. Первые проблемы начались с логистики. Действительно, кажется просто сформировать заказ, но не всегда удается полностью избавиться от проблем с

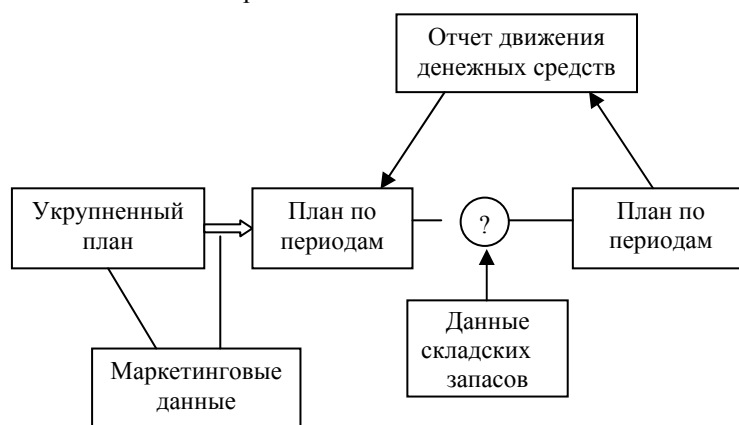


Рис. 3.1 MPS-система

доставкой и ассортиментом, потом скидки при увеличении объема, замена моделей и т.д. и т.п. Одной из наиболее сложных проблем, возникших при формировании заказа, была проблема прогнозирования необходимого объема и срока поставки. Действительно, например, чай из Индии не доставляется мгновенно, да и собирают определенные сорта в определенное время и т.д.. Следовательно, нужно прогнозировать спрос на длительное время вперед, учитывать длительность (а часто и сезон) производства и потребности в складских площадях.

### 3.1.1 MRP-система

Прежде чем описывать саму структуру MRP, введем краткий глоссарий основных ее понятий.

*Материалами* будем называть все сырье и отдельные комплектующие, составляющие конечный продукт. В дальнейшем не будем делать различий между понятиями материал и комплектующий.

*MRP-система*, MRP-программа – компьютерная программа, работающая по алгоритму, регламентированному MRP-методологией. Как и любая компьютерная программа, она обрабатывает файлы данных (входные элементы) и формирует на их основе файлы-результаты.

*Статус материала* является основным указателем на текущее состояние материала. Каждый отдельный материал в каждый момент времени имеет статус в рамках MRP-системы, который определяет, имеется ли данный материал в наличии на складе, зарезервирован ли он для других целей, присутствует ли в текущих заказах или заказ на него только планируется. Таким образом, статус материала однозначно описывает степень готовности каждого материала быть пущенным в производственный процесс.

*Страховой запас* материала необходим для поддержания процесса производства в случае возникновения непредвиденных и неустраняемых задержек в его поставках. В идеальном случае, если механизм поставок полагать безупречным, MRP-методология не постулирует обязательное наличие страхового запаса, и его объемы устанавливаются различными для каждого конкретного случая, в зависимости от сложившейся ситуации с поступлением материалов. Подробней об этом будет рассказано ниже.

*Потребность в материале* в MRP-программе представляет собой определенную количественную единицу, отображающую возникшую в некоторый момент времени в течение периода планирования необходимость в заказе данного материала. Различают понятия полной потребности в материале, которое отображает то количество, которое требуется пустить в производство, и чистой потребности, при вычислении которой учитывается наличие всех страховых и зарезервированных запасов данного материала. Заказ в КИС автоматически создается по возникновению отличной от нуля чистой потребности.



*Процесс планирования* включает в себя функции автоматического создания проектов заказов на закупку и/или внутреннее производство необходимых материалов. Другими словами, система MRP оптимизирует время поставки комплектующих, тем самым уменьшая затраты на производство и повышая его эффективность. Основными преимуществами использования подобной системы в производстве являются:

- гарантия наличия требуемых комплектующих и уменьшение временных задержек в их доставке, и, следовательно, увеличение выпуска готовых изделий без увеличения числа рабочих мест и нагрузок на производственное оборудование.
- уменьшение производственного брака в процессе сборки готовой продукции, возникающего из-за использования неправильных комплектующих.
- упорядочивание производства ввиду контроля статуса каждого материала, позволяющего однозначно отслеживать весь его конвейерный путь, начиная от создания заказа на данный материал, до его положения в уже собранном готовом изделии. Благодаря этому достигается также полная достоверность и эффективность производственного учета.

Все эти преимущества фактически вытекают из самой философии MRP, базирующейся на том принципе, что все материалы-комплектующие, составные части и блоки готового изделия должны поступать в производство одновременно, в запланированное время, чтобы обеспечить создание конечного продукта без дополнительных задержек. MRP-система ускоряет доставку тех материалов, которые в данный момент нужны в первую очередь и задерживает преждевременные поступления таким образом, что все комплектующие, представляющие собой полный список составляющих конечного продукта, поступают в производство одновременно. Это необходимо во избежание ситуации, когда задерживается поставка одного из материалов, и производство вынуждено приостановиться даже при наличии всех остальных комплектующих конечного продукта. Основная цель MRP-системы формировать, контролировать и при необходимости изменять даты поступления заказов таким образом, чтобы все материалы, необходимые для производства, поступали одновременно.

На практике MRP-система представляет собой компьютерную программу, которая логически представлена на рис. 3.2, где отображены основные информационные элементы MRP-системы.

Опишем основные входные элементы MRP-системы. Описание состояния материалов (Inventory Status File) является основным входным элементом MRP-программы. В нем должна быть отражена максимально полная информация о всех материалах-комплектующих, необходимых для производства конечного продукта. В этом элементе должен быть указан статус каждого материала, определяющий, имеется ли



**Рис. 3.2 Входные элементы и результаты работы MRP-программы**

он на руках, на складе, в текущих заказах или его заказ только планируется, а также описания, его запасы, расположения, цены, возможные задержки поставок, реквизиты поставщиков. Информация по всем вышеперечисленным позициям должна быть заложена отдельно по каждому материалу, участвующему в производственном процессе.

Программа производства – заинтересованность в получении тех или иных номенклатурных позиций, которую проявляет непосредственно потребитель продукции предприятия, которому эта продукция отгружается. Примерами таких номенклатурных позиций могут быть готовые изделия, запасные части, продаваемые на сторону полуфабрикаты и комплектующие и т.п. Потребность может быть представлена или прогнозом продаж, или уже имеющимися в наличии заказами покупателей, или и тем и другим одновременно. Информация о прогнозах продаж и заказах на продажу является основанием для формирования главного календарного плана производства (MPS – Master Production Schedule), охватывающего все включаемые в план производства номенклатурные позиции. MPS формируется как в объемном, так и в календарном исполнении.

Сначала создается пробная программа производства, впоследствии тестируемая на выполнимость дополнительно прогнозом через CRP-систему, которая определяет, достаточно ли производственных мощностей для ее осуществления. Если производственная программа признана выполнимой, то она автоматически формируется в основную и становится входным элементом MRP-системы. Это необходимо, потому как рамки требований

по производственным ресурсам являются прозрачными для MRP-системы, которая формирует на основе производственной программы график возникновения потребностей в материалах. Однако, в случае недоступности ряда материалов или невозможности выполнить план заказов, необходимый для поддержания реализуемой с точки зрения CPR производственной программы, MRP-система в свою очередь указывает о необходимости внести в нее корректировки.

Перечень составляющих конечного продукта (Bills of Material File) – это список материалов и их количество, требуемое для производства конечного продукта. Таким образом, каждый конечный продукт имеет свой перечень составляющих. Кроме того, здесь содержится описание структуры конечного продукта, т.е. полную информацию по технологии его сборки. Чрезвычайно важно поддерживать точность всех записей в этом элементе и соответственно корректировать их всякий раз при внесении изменений в структуру и/или технологию производства конечного продукта.

Напомним, что каждый из вышеуказанных входных элементов представляет собой компьютерный файл данных, используемый MRP-программой. В настоящий момент MRP-системы реализованы на самых разнообразных аппаратных платформах и включены в качестве модулей в большинство финансово-экономических систем. Не будем останавливаться на техническом аспекте вопроса и перейдем к описанию логических шагов работы MRP-программы. Цикл ее работы состоит из следующих основных этапов.

Прежде всего, MRP-система, анализируя принятую программу производства, определяет оптимальный график производства на планируемый период.

Далее, материалы, не включенные в производственную программу, но присутствующие в текущих заказах, включаются в планирование как отдельный пункт. На этом шаге, на основе утвержденной программы производства и заказов на комплектующие, не входящие в нее, для каждого отдельно взятого материала вычисляется полная потребность в соответствии с перечнем составляющих конечного продукта.

Чистая потребность = Полная потребность – Инвентаризовано на руках –

– Страховой запас – Зарезервировано для других целей.

Далее, на основе полной потребности, учитывая текущий статус материала, для каждого периода времени и для каждого материала вычисляется чистая потребность по указанной формуле. Если чистая потребность в материале больше нуля, то системой автоматически создается заказ на материал.

И наконец, все заказы, созданные ранее текущего периода планирования, рассматриваются, и в них, при необходимости, вносятся изменения, чтобы предотвратить преждевременные поставки и задержки поставок от поставщиков.

Таким образом, в результате работы MRP-программы производится ряд изменений в имеющихся заказах и, при необходимости, создаются новые для обеспечения оптимальной динамики хода производственного процесса. Эти изменения автоматически модифицируют Описание Состояния Материалов, так как создание, отмена или модификация заказа, соответственно, влияют на статус материала, к которому он относится. В результате работы MRP-программы создается план заказов на каждый отдельный материал на весь срок планирования, обеспечение выполнения которого необходимо для поддержки программы производства.

План Заказов (Planned Order Schedule) определяет, какое количество каждого материала должно быть заказано в каждый рассматриваемый период времени в течение срока планирования. План заказов является руководством для дальнейшей работы с поставщиками и, в частности, определяет производственную программу для внутреннего производства комплектующих при наличии такового.

Изменения к плану заказов (Changes in planned orders) являются модификациями к ранее спланированным заказам. Ряд заказов могут быть отменены, изменены или задержаны, а также перенесены на другой период.

MRP-система формирует некоторые второстепенные результаты в виде отчетов, целью которых является привлечь внимание на узкие места в течение планируемого периода, т.е. те промежутки времени, когда требуется дополнительный контроль за текущими заказами, а также для того, чтобы вовремя известить о возможных системных ошибках, возникших при работе программы. Итак, MRP-система формирует следующие дополнительные результаты-отчеты.

Отчет об узких местах планирования (Exception report) предназначен для того, чтобы заблаговременно проинформировать пользователя о промежутках времени в течение срока планирования, которые требуют особого внимания и в которые может возникнуть необходимость внешнего управленческого вмешательства. Типичными примерами ситуаций, которые должны быть отражены в этом отчете, могут быть непредвиденно запаздывающие заказы на комплектующие, избытки комплектующих на складах и т.п.

Исполнительный отчет (Performance Report) является основным индикатором правильности работы MRP-системы и имеет целью оповещать пользователя о возникших критических ситуациях в процессе планирования, таких как, например, полное израсходование страховых запасов по отдельным комплектующим, а также о всех возникающих системных ошибках в процессе работы MRP-программы.

Отчет о прогнозах (Planning Report) представляет собой информацию, используемую для составления прогнозов о возможном будущем изменении объемов и характеристик выпускаемой продукции, полученную в результате анализа текущего хода производственного процесса и отчетах о продажах. Отчет о прогнозах может использоваться и для долгосрочного планирования потребностей в материалах.

Таким образом, использование MRP-системы для планирования производственных потребностей позволяет оптимизировать время поступления каждого материала, тем самым значительно снижая складские издержки и облегчая ведение производственного учета. Однако, среди пользователей MRP-программ существует расхождение в мнениях относительно использования страхового запаса для каждого материала. Сторонники использования страхового запаса утверждают, что он необходим в силу того, что зачастую механизм доставки грузов не является достаточно надежным, и возникшее, в силу различных факторов, полное израсходование запасов на какой-либо материал, автоматически приводящее к остановке производства, обходится гораздо дороже, чем постоянно поддерживаемый его страховой запас. Противники использования страхового запаса утверждают, что его отсутствие является одной из центральных особенностей концепции MRP, поскольку MRP-система должна быть гибкой по отношению к внешним факторам, вовремя внося изменения в план заказов в случае непредвиденных и неустраняемых задержек поставок. Но в реальной ситуации, как правило, вторая точка зрения может быть реализована для планирования потребностей в производстве изделий, спрос на которые относительно прогнозируем и контролируем, и объем производства может быть установлен в производственной программе постоянным в течение некоторого, относительно длительного периода. Следует заметить, что в российских условиях, когда задержки в процессах поставки являются скорее правилом, чем исключением, на практике целесообразно применять планирование с учетом страхового запаса, объемы которого устанавливаются в каждом отдельном случае.

Данная фаза развития стандарта MRP имела место при преобладающем характере пакетной обработки информации на удаленных вычислительных центрах (кустовых или корпоративных). Интерактивные технологии тогда развития еще не получили, и анализ «а что будет, если?..» практически не проводился. По сути MRP просто фиксировала ситуацию в «развернутом» виде.

### 3.1.2 Планирование производственных мощностей с помощью CRP-системы

Система планирования производственных мощностей по методологии CRP (Capacity Requirements Planning) применяется для проверки пробной программы производства, созданной в соответствии с прогнозами спроса на продукцию, на возможность ее осуществления имеющимися в наличии производственными мощностями.

Очевидно, что с ростом возможностей в области обработки данных присущие MRP ограничения перестали удовлетворять менеджеров и плановиков.

Поэтому следующим шагом стала необходимость обрабатывать ситуацию с загрузкой производственных мощностей и учитывать ресурсные ограничения производства. Эта технология известна как CRP и представлена на рис. 3.3.

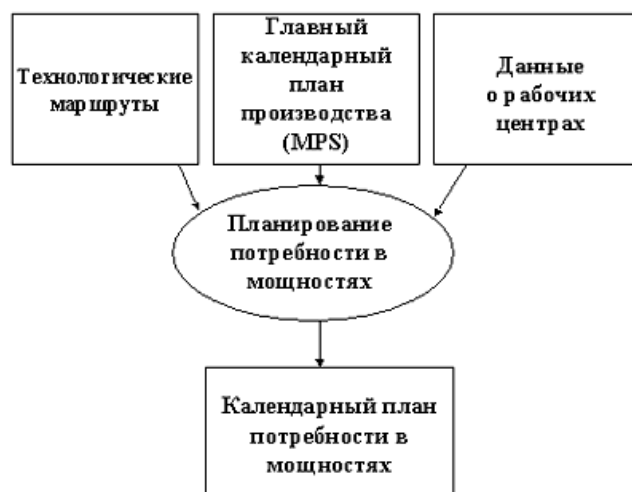


Рис. 3.3 Планирование потребности в мощностях

Для работы механизма CRP необходимы три массива исходных данных.

1 *Данные о календарном плане производства*, содержащие сведения о производственных заказах. Они являются исходными и для MRP. Стоит отметить, что запуск CRP возможен только после того, как отработало MRP, потому что исходными данными для CRP являются также результаты работы MRP в виде плановых заказов по номенклатурным позициям зависимого спроса, а не только по номенклатурным позициям независимого спроса.

2 *Данные о рабочих центрах*. Рабочий центр – это определенная производственная мощность, состоящая из одной или нескольких машин (людей и/или оборудования), которая в целях планирования потребности в мощностях (CRP) и подробного календарного планирования может рассматриваться как одна производственная единица. Можно сказать, что рабочий центр – это группа взаимозаменяемого оборудования, расположенная на локальном производственном участке. Для работы CRP необходимо предварительное формирование рабочего календаря рабочих центров с целью вычисления доступной производственной мощности.

3 *Данные о технологических маршрутах* изготовления номенклатурных позиций. Здесь указываются все сведения о порядке осуществления технологических операций и их характеристиках (технологические времена, персонал, другая информация). Этот массив данных вместе с первым массивом (MPS) формирует загрузку рабочих центров.

CRP информирует обо всех расхождениях между планируемой загрузкой и имеющимися мощностями, позволяя предпринять необходимые регулирующие воздействия. При этом каждому изготавливаемому изделию назначается соответствующий технологический маршрут с описанием ресурсов, требуемых на каждой его операции, на каждом рабочем центре.

Если все доступные возможности увеличения производительности не достаточны, чтобы удовлетворить требования MRP, то может возникнуть потребность заново перепланировать MPS. В простейших бизнес-моделях MRP-систем производительность рабочих центров обычно считается неограниченной и такие проблемы не возникают, однако, поскольку реальная производительность всегда ограничена, то современные MRP-системы предоставляют возможность производить планирование в условиях ограниченных ресурсов.

На рис. 3.4 приведен пример «адаптации» загрузочного профиля к реальным производственным мощностям.

Так как очевидна перегрузка во 2 – 4 периодах, то необходимо принять меры к ее ликвидации. Стандартно применяются следующие варианты:

- распределение нагрузки на другие периоды, когда загрузка не достигает нормального уровня;
- увеличить доступную мощность – например, объявить сверхурочные работы;
- передать работы на субконтракт.

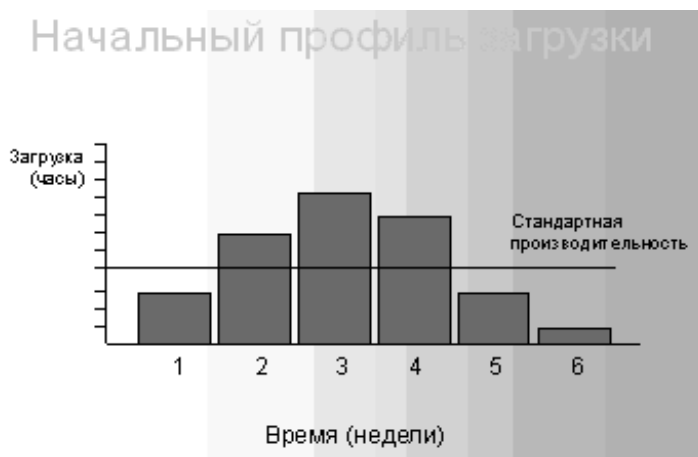


Рис. 3.4 Начальный профиль загрузки

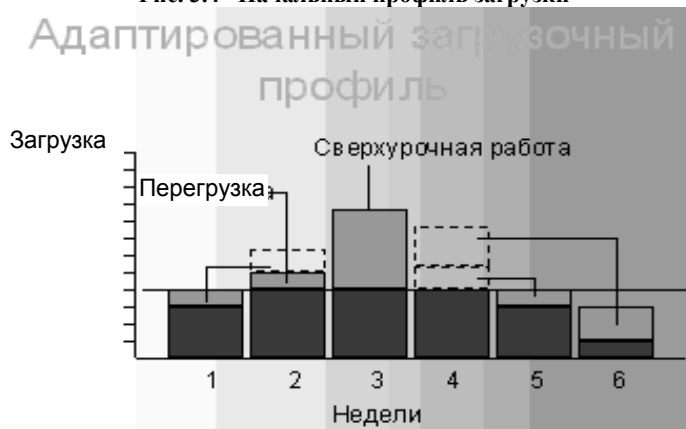


Рис. 3.5 Адаптированный загрузочный профиль

На рис. 3.5 представлены стандартные рецепты, позволяющие добиться равномерной загрузки производственных мощностей в пределах норм загрузки.

Итак, в MRP-системе функция CRP вычисляет производственные мощности, требуемые, чтобы произвести запланированный производственный заказ, сгенерированный MPS, MRP.

MPS и MRP используются, чтобы формировать плановый производственный заказ прежде, чем процесс CRP вычисляет требуемую производительность. Запланированный производственный заказ, сгенерированный этими функциями, обеспечивает основные исходные данные для процесса CRP. Если компоненты назначены к системе заказа, то запланированный производственный заказ для пополнения складских запасов (полуфабрика-

тов собственного производства) должен также быть сгенерирован прежде, чем запустить CRP. Планирование производительности должно быть выполнено до того, как плановый производственный заказ, сгенерированный MPS, MRP, и SIC, может быть передан управлению цехом.

Другая важная функция CRP состоит в том, чтобы проанализировать финансовые последствия запланированного производства. В дополнение к вычислению требуемой производительности процесс CRP также выполняет финансовый анализ отложенного приобретения и производственного заказа. Финансовый анализ в CRP использует информацию о закупках, сбыте, складских запасах, MPS, планировании потребности.

Финансовая информация, анализируемая процессом CRP, включает доступные складские запасы, открытые заказы на закупку, открытые заказы на продажу, открытые производственные заказы и запланированные (плановые) заказы. Финансовый анализ включает все запланированные движения (перемещения) складских запасов сбыта, MPS, планирования потребности и плановые потребности, сгенерированные системой управления проектом.

Следует отметить, что CRP не занимается оптимизацией загрузки, осуществляя лишь расчетные функции по заранее определенной производственной программе согласно описанной нормативной информации.

В этом смысле и MRP, и CRP – плановые механизмы, позволяющие получать корректный и реальный план-график производства на основе использования опыта и знаний лиц, принимающих решения. Иногда технологию MRP называют еще MRP I. Можно отметить, что налаженная технология MRP/CRP при наличии достаточных вычислительных мощностей позволяет, по сути, осуществлять моделирование ситуации.

В процессе работы CRP-системы разрабатывается план распределения производственных мощностей для обработки каждого конкретного цикла производства в течение планируемого периода. Также устанавливается технологический план последовательности производственных процедур, и в соответствии с пробной программой производства определяется степень загрузки каждой производственной единицы на срок планирования. Если после цикла работы CRP-модуля программа производства признается реально осуществимой, то она автоматически подтверждается и становится основной для MRP-системы. В противном случае в нее вносятся изменения, и она подвергается повторному тестированию с помощью CRP-модуля. В дальнейшем эволюционном развитии системы планирования производства стали представлять собой интеграцию многих отдельных модулей, которые, взаимодействуя, увеличивали гибкость системы в целом.

### 3.1.3 Замокнутый цикл MRP (Closed loop MRP)

Следующим после MRP/CRP шагом по пути развития стандарта MRP стало создание технологии «Замокнутый цикл MRP» (Closed loop MRP), предложенной в конце 70-х гг. XX в. О. Уайтом, Д. Плослом и другими.

Термин «Замокнутый цикл» отражает основную особенность модифицированной системы, заключающуюся в создании замкнутого цикла путем налаживания обратных связей, улучшающих отслеживание текущего состояния производственной системы. Дополнительная реализация мониторинга выполнения плана снабжения и производственных операций позволила снять те ограничения степени достоверности результата планирования, ранее присущие MRP, которые существовали из-за невозможности отследить состояние открытых заказов.

С добавлением указанных функций к MRP /CRP был сформирован стандарт «Замокнутый цикл MRP». Отличие MRP / CRP от Closed-loop MRP хорошо поясняется схемой на рис. 3.6.

Из рис. 3.6 видим, что в случае с технологией «Замокнутый цикл MRP» в процесс вовлечены только операции, связанные со снабжением и производством, а процессы сбыта (продаж) и финансового учета технологией не задействованы.

Следует отметить, что бизнес-планирование и планирование продаж и операций (Sales & Operations Planning) в контур MRP/CRP не входят, а приведены лишь для иллюстрации связи MRP/CRP с вышестоящими уровнями планирования.

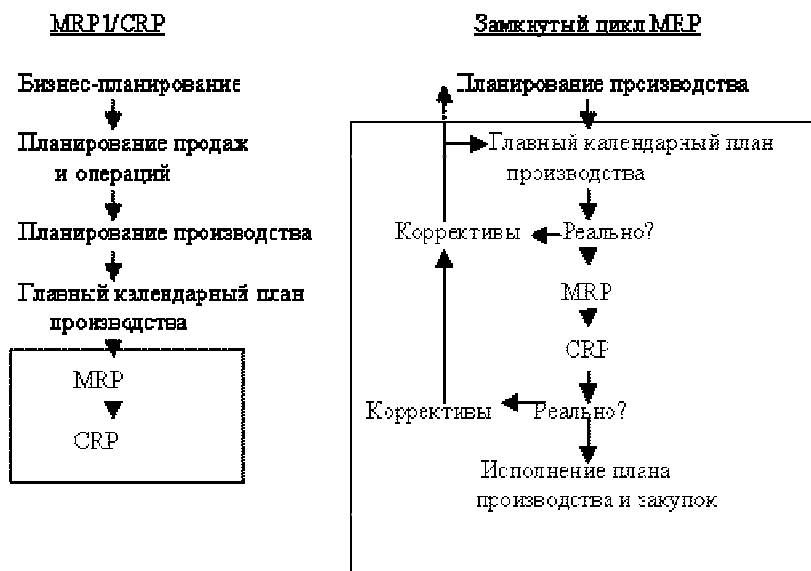


Рис. 3.6 Сравнение MRP/CRP и «Замокнутый цикл MRP»  
3.1.4 MRP II-система

В дальнейшем усовершенствование системы привело к трансформации системы MRP с замкнутым циклом в расширенную модификацию, которую впоследствии назвали MRP II (Manufactory Resource Planning), ввиду идентичности аббревиатур. Эта система была создана для эффективного планирования всех ресурсов производственного пред-

приятия, в том числе финансовых и кадровых. Кроме того, система класса MRP II способна адаптироваться к изменениям внешней ситуации и эмулировать ответ на вопрос «Что если?» MRP II представляет собой интеграцию большого количества отдельных модулей, таких как планирование бизнес-процессов, планирование потребностей в материалах, планирование производственных мощностей, планирование финансов, управление инвестициями и т.д. Результаты работы каждого из модуля анализируются всей системой в целом, что существенно и обеспечивает ее гибкость по отношению к внешним факторам. Именно это свойство является краеугольным камнем современных систем планирования, поскольку большое количество производителей производят продукцию с заведомо коротким жизненным циклом, требующую регулярных доработок. В таком случае появляется необходимость в КИС, которая позволяет оптимизировать объемы и характеристики выпускаемой продукции, анализируя текущий спрос и положение на рынке в целом.

MRP II – это набор проверенных на практике разумных принципов, моделей и процедур управления и контроля, служащих повышению показателей экономической деятельности предприятия. Идея MRP II опирается на несколько простых принципов, например, разделение спроса на зависимый и независимый. MRP II Standart System содержит описание 16 групп функций системы:

1. Планирование продаж и производства (Sales and Operation Planning).
2. Управление спросом (Demand Management).
3. Составление плана производства (Master Production Scheduling).
4. Планирование материальных потребностей (Material Requirement Planning).
5. Спецификации продуктов (Bill of Materials).
6. Управление складом (Inventory Transaction Subsystem).
7. Плановые поставки (Scheduled Receipts Subsystem).
8. Управление на уровне производственного цеха (Shop Flow Control).
9. Планирование производственных мощностей (Capacity Requirement Planning).
10. Контроль входа/выхода (Input/output control).
11. Материально-техническое снабжение (Purchasing).
12. Планирование ресурсов распределения (Distribution Resource Planning).
13. Планирование и контроль производственных операций (Tooling Planning and Control).
14. Управление финансами (Financial Planning).
15. Моделирование (Simulation).
16. Оценка результатов деятельности (Performance Measurement).

С накоплением опыта моделирования производственных и непроизводственных операций эти понятия постоянно уточняются, постепенно охватывая все больше функций.

Задачей КИС класса MRP II является оптимальное формирование потока материалов (сырья), полуфабрикатов (в том числе находящихся в производстве) и готовых изделий. Система класса MRP II имеет целью интеграцию всех основных процессов, реализуемых предприятием, таких как снабжение, запасы, производство, продажа и дистрибуция, планирование, контроль за выполнением плана, затраты, финансы, основные средства и т.д.

Результаты использования КИС стандарта MRP II:

- получение оперативной 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;
- решение задач оптимизации производственных и материальных потоков;
- реальное сокращение материальных ресурсов на складах;
- планирование и контроль за всем циклом производства с возможностью влияния на него в целях достижения оптимальной эффективности в использовании производственных мощностей, всех видов ресурсов и удовлетворения потребностей заказчиков;
- автоматизация работ договорного отдела с полным контролем за платежами, отгрузкой продукции и сроками выполнения договорных обязательств;
- финансовое отражение деятельности предприятия в целом;
- значительное сокращение непроизводственных затрат;
- защита инвестиций, произведенных в информационные технологии;
- возможность поэтапного внедрения системы с учетом инвестиционной политики конкретного предприятия.

В основу MRP II положена иерархия планов. Планы нижних уровней зависят от планов более высоких уровней, т.е. план высшего уровня предоставляет входные данные, намечаемые показатели и/или какие-то ограничительные рамки для планов низшего уровня. Кроме того, эти планы связаны между собой таким образом, что результаты планов нижнего уровня оказывают обратное воздействие на планы высшего уровня.

Если результаты плана нереалистичны, то этот план или планы высшего уровня должны быть пересмотрены. Таким образом можно проводить координацию спроса и предложения ресурсов на определенном уровне планирования и ресурсов на высших уровнях планирования.

Следует отметить, что практически все основные системы планирования очень тесно взаимосвязаны между собой и поэтому, рассматривая MRP II, будем вынуждены затронуть все остальные системы планирования. Для большей ясности нарисуем картину условной взаимосвязи основных плановых систем, которая представлена на рис. 3.7 блок-схемой иерархии планов.

Приведенная блок-схема показывает только «технологический взгляд» на систему планирования, кроме него возможен например логистический взгляд, который упрощенно может быть представлен так, как на рис. 3.8 или, например, финансовый взгляд (рис. 3.9).

Следует обратить внимание на то, что при использовании методов функционального управления, в частности MRP II, бюджетирование используется только как специфическая методика в кассовом планировании и в управлении (т.е. при планировании движения денежных средств, платежей и поступлений) и некоторых других случаях, имеющих отношение к управлению финансами. Большинство же бюджетов, часто используемых в Российской практике бюджетирования, либо имеют функциональные эквиваленты в методике MRP, как например бюджет продаж – это обычно бизнес-план или прогноз продаж (в зависимости от производственной модели), бюджет закупок – это зависимая потребность в закупаемых материалах и компонентах, полученная в результате MRP-процесса (точнее его части – разузлования), либо получаются расчетными методами из компонент методологии. В частности, например бюджеты накладных производственных расходов и бюджет заработной платы получаются в результате пересчета полученных профилей загрузки рабочих центров по нормативам накладных расходов и заработной платы. Принципиальное достоинство MRP II,

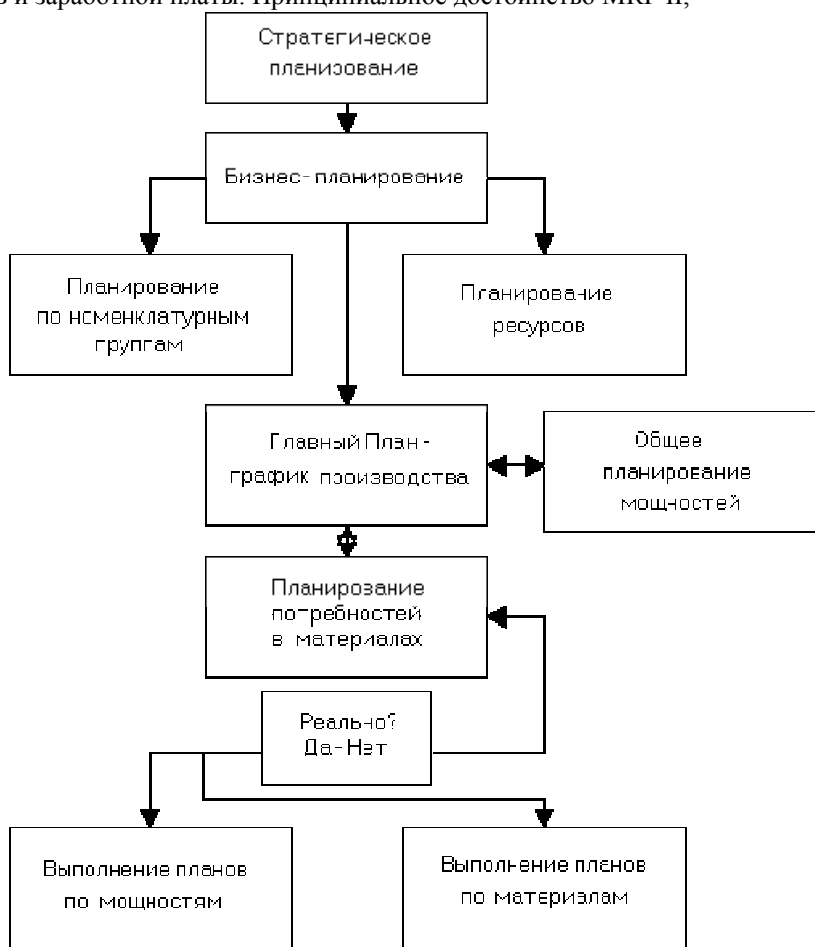


Рис. 3.7 Блок-схема иерархии планов



Рис. 3.8 Логистический взгляд на систему планирования

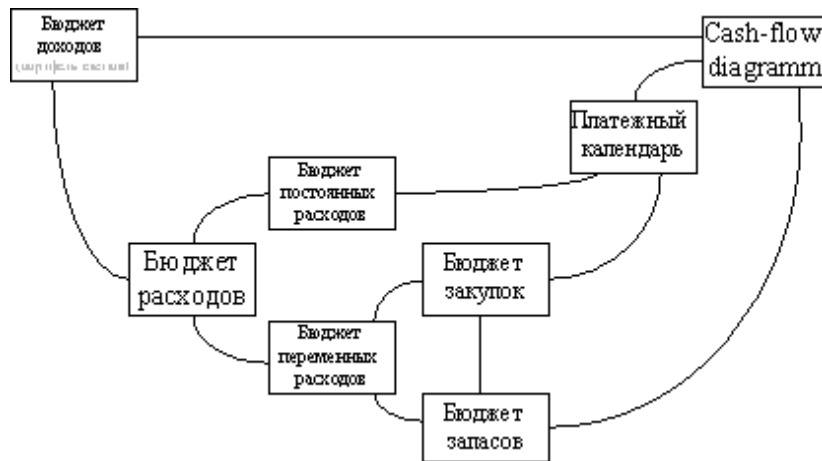


Рис. 3.9 Финансовый взгляд на систему планирования

особенно в ее современных реализациях – это динамический характер полученных данных, их оперативность и обновляемость «по потребности», в отличие от статического по своей сущности метода бюджетирования.

Практически для управления предприятием и MRP II-систем «технологический взгляд» является наиболее сложным как с точки зрения реализации, так и концептуально, поэтому сначала остановимся именно на нем (см. рис. 3.7).

**Стратегическое планирование.** Стратегическое планирование – это долгосрочное планирование. Оно обычно составляется на срок от одного до пяти лет и основано на макроэкономических показателях, таких как тенденции развития экономики, изменение технологий, состояние рынка и конкуренции. Стратегическое планирование обычно распространяется на каждый год пятилетки и представляет собой плановые показатели (цели) высшего уровня.

**Бизнес-планирование.** Бизнес-план – это план на год, который также составляется на ежегодной основе. Иногда он неоднократно пересматривается в течение года. Как правило, он является результатом совещания управленческого состава, на котором сводятся планы продаж, инвестиций, развития основных средств и потребности в капитале и бюджетировании. Эта информация подается в денежном выражении. Бизнес-план определяет плановые показатели по объемам продаж и производства, а также другие планы низшего уровня.

**Планирование объемов продаж и производства.** Если бизнес-план предоставляет итоговые данные по объемам продаж помесечно (в денежном выражении), то план объемов продаж и производства разбивает эту информацию по 10 – 15 ассортиментным группам. В результате получают план производства, который ежемесячно пересматривается, принимая во внимание план предыдущего месяца, реальные результаты и данные бизнес-плана.

План объемов продаж и производства обычно включает следующие элементы: объем продаж, производство, запасы, незавершенный объем производства, отгрузка.

Из этих элементов «Объем продаж» и «Отгрузка» – это прогнозы, так как это внешние данные, которые прямому контролю не поддаются. Объем производства планируется, это внутренний показатель, поддающийся прямому контролю. Планы по объемам запасов и незавершенным объемам производства контролируются косвенно, манипулируя данными прогнозов объема продаж, прогнозов объема отгрузки и/или плана объемов производства.

Объемы запасов и незавершенки управляются по-разному, в зависимости от типов продукции, выпускаемой или продаваемой компанией. Плановый объем запасов – это важный фактор, особенно для тех компаний, которые производят продукцию на склад. Плановый объем незавершенки является важным фактором для тех компаний, которые производят продукцию на заказ.

Фокусом планирования объема продаж и производства является план производства. Хотя он и называется планом производства, это в принципе не просто план выпуска продукции. Он требует наличия необходимого объема ресурсов по всей компании в целом. Если отдел маркетинга планирует скачок в продажах определенного ассортимента продуктов, инженеры должны обеспечить наличие необходимого объема оборудования; отдел МТС должен будет обеспечить дополнительные поставки материалов (наличие новых поставщиков); отдел кадров – дополнительный объем трудовых ресурсов, а также организовать новые рабочие смены. Плюс ко всему необходимо обеспечить наличие достаточного объема капитала (для оплаты дополнительного объема ресурсов и запасов).

**Планирование ресурсов.** План производства будет нереален, если не обеспечено наличие необходимого объема ресурсов. Планирование ресурсов – это долгосрочное планирование, которое позволяет оценить необходимый (для выполнения плана производства) и наличный объем ключевых ресурсов, таких как люди, оборудование, здания и сооружения. Если возникнет потребность в наличии необходимого объема дополнительных ресурсов, то, возможно, потребуется пересмотреть бизнес-план.



Планирование ресурсов затрагивает только ключевые ресурсы и составляется на срок действия плана по производству (обычно один год). Ресурс может считаться ключевым, если его стоимость достаточно велика, или если срок его поставки достаточно велик или если от него зависят другие ресурсы. Ресурсы могут быть как внешними (возможности поставщиков), так и внутренними (оборудование, складские площади, деньги).

**Главный план-график производства (ГПП).** Задача начальника отдела планирования – перевод производственного плана в специфичный план-график производства. Этот план – ГПП – план производства, наложенный на шкалу времени. ГПП показывает, что будет производиться, когда и в каких объемах.

Производственный план выражен в таких единицах, как рубли, часы, тонны, поэтому для того, чтобы получить ГПП, необходимо произвести некоторые шаги по трансформации производственного плана. Плановые объемные показатели по ассортиментной группе необходимо перевести в плановые объемы и сроки по каждому продукту этой группы в отдельности. В зависимости от типа и объема выпускаемой продукции ГПП можно разбить на недельные, дневные и даже сменные планы.

Одна из основных целей ГПП – это обеспечение буфера: ГПП отличает прогнозы и потребности отдела сбыта от MRP (планирование потребностей в материалах). Философия такова: прогнозы и заказы на продажу (заказы клиентов) выражают спрос (или отгрузку), в то время как ГПП отображает то, что реально будет произведено в соответствии с имеющимся спросом. В соответствии с ГПП возможно производство продукции в период, когда спрос на нее не высок, и наоборот. Это может иметь место при производстве продукции, спрос на которую сезонен.

Начальник отдела планирования должен принимать во внимание все источники независимого спроса. Независимый спрос – это спрос, который может быть прогнозом, обычно это спрос на готовую продукцию и запчасти. Он в корне отличается от зависимого спроса (спрос, который можно рассчитать, исходя из данных по составу изделия). Источники независимого спроса: производственный план, прогнозируемый объем отгрузки, заказы клиентов (при производстве или сборке под заказ), спрос на запчасти, межзаводской спрос и страховой запас.

Основная проблема в составлении ГПП – это определение того, планирование по каким изделиям/комплектующим должно вестись отделом планирования, а по каким должно вестись автоматически (системой MRP). Изделия, планируемые отделом планирования, – это те изделия, планирование которых должно вестись под контролем людей. Изделия, планируемые системой MRP, т.е. автоматически, не требуют такой степени контроля (они зависят от ГПП). Определение того, как должно вестись планирование того или иного вида изделия, зависит от типов изделий и технологических процессов. Обычно очень маленькое количество изделий должно контролироваться отделом планирования.

**Общее планирование мощностей.** Как и планирование ресурсов, общее планирование мощностей является долгосрочным и ведется по ключевым ресурсам. Этот процесс использует данные ГПП, а не данные производственного плана. Так, если ГПП выражен в объемных и временных характеристиках, то общее планирование мощностей используется для создания более детализированного плана, который может быть очень полезен при оценке средних потребностей компании в целом, а также для оценки ГПП.

**MRP или планирование потребностей в материалах.** Исторически MRP (планирование потребностей материалов) предназначалось для контроля за запасами и их пополнения. В рамках MRP II (планирование ресурсов предприятия) его использование было расширено до планирования потребностей в мощностях, проведения приоритизации и до замыкания всей цепочки планирования.

MRP отвечает на четыре основных вопроса:

- 1) Что мы собираемся производить?
- 2) Что нам для этого необходимо?
- 3) Чем мы уже располагаем?
- 4) Что нам необходимо дополнить?

ГПП отвечает на первый вопрос «Что мы собираемся произвести?». В целях достижения целей, поставленных ГПП, ведется планирование всей производственной и дистрибьюторской деятельности. Так как ГПП – это график, то он также отвечает и на такие вопросы, как: Сколько? Когда?

Второй вопрос «Что нам для этого необходимо?» по сути спрашивает: Какие изделия/комплектующие нам нужно произвести (или закупить), чтобы выполнить планы ГПП? Чтобы ответить на этот вопрос, нам нужно знать две вещи: ГПП и правильные данные о составе изделия (структуре продукта, формуле продукта). ГПП и данные о составе изделия позволяют системе определить: что, сколько и когда потребуется для того, чтобы произвести то, что нам нужно.

Вопрос «Чем мы уже располагаем?» можно разделить на два вопроса: Что у нас уже есть на руках? и Что мы ожидаем по заказам? Наличный запас на складе – это ответ на первый вопрос, а плановый объем поступлений продукции с производства и от поставщиков – это ответ на второй вопрос. Все вместе эти данные не только дают информацию о наличном объеме запасов, но они также позволяют системе оценить ожидаемый объем запаса. Чтобы ответить на последний вопрос, нужно знать ответы на три предыдущих. Взяв то, что нужно произвести (брутто-потребности), отняв то, что уже есть (на складе и плановые поступления), мы узнаем то, что нам нужно дополнить (нетто-потребности).

**CRP или планирование потребностей в мощностях.** Наличие необходимого объема необходимых материалов ничего не значит без наличия достаточного свободного объема рабочего времени. CRP (или планирование потребностей в мощностях) – это планирование среднего уровня, которое использует данные запланиро-

ванных MRP-заказов и заказов на производство для определения необходимого объема рабочего времени (как по трудовым, так и по техническим ресурсам).

Планирование ресурсов и общее планирование мощностей – это планирование высшего уровня, используемое для планирования таких ресурсов, как физическое оборудование. CRP является более детализированным планированием. Загрузка рабочих мест рассчитывается на основе технологического маршрута изготовления продукта, который определяет, каким именно образом производится данный вид продукта. Технологический маршрут похож на инструкцию к применению – набор шагов (или техноопераций), которые необходимо совершить для изготовления чего-то. Каждая технооперация совершается на каком-то рабочем месте, которое может состоять из одного или нескольких человек и/или оборудования.

**DRP или планирование потребностей в распределении.** Когда какие-то материалы передвигаются от поставщика к потребителю, они передвигаются по цепи поставок (или рыночному каналу). Если изобразить это графически, то цепь поставок представляет собой потоки спроса и предложения между поставщиками и какими-то подразделениями компании Заказчика, между этими подразделениями и клиентами или между различными подразделениями одной компании. DRP (планирование потребностей в распределении) координирует спрос, предложение и ресурсы между подразделениями одной или нескольких компаний (рис. 3.10).

В цепи поставок могут быть два и более уровней производственных и/или дистрибуторских подразделений. Эти подразделения могут находиться в различной зависимости друг от друга; важным моментом является то, что одно подразделение может поставить продукцию другому подразделению.

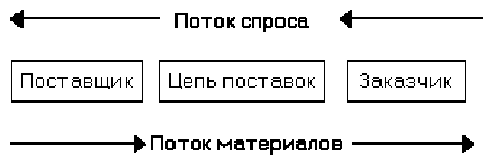


Рис. 3.10 Планирование потребностей в распределении

Например, предприятие производит товары на территории одного подразделения, а продает их с отдельного склада продаж (рис. 3.11).

Другое предприятие может иметь центральный центр дистрибуции, который поставляет продукцию на склады региональных отделений (рис. 3.12).

И третий пример: компания имеет производственные мощности в двух городах (рис. 3.13).

При планировании спроса и предложения материалов между подразделениями отвечают на три основных вопроса:

- 1) Что нам нужно получить (с других подразделений)?
- 2) Что мы собираемся поставить (другим подразделениям)?
- 3) Что мы можем поставить?

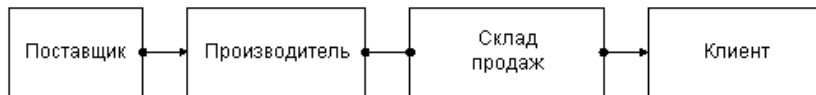


Рис. 3.11 Продажа товаров с отдельного склада

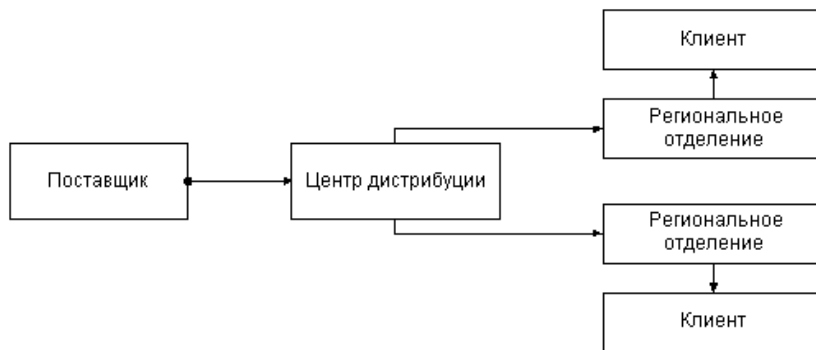


Рис. 3.12 Продажа со складов региональных отделений



**Рис. 3.13 Производство в двух городах**

Хотя эти вопросы и похожи на вопросы, задаваемые MRP (планирование потребностей в материалах), однако существует одно принципиальное отличие. В MRP достаточно знать «Какой» и «Когда» ожидается спрос и предложение. Когда же существует несколько подразделений, между которыми постоянно передвигается продукция, тогда DRP необходимо знать плюс ко всему, где (каким подразделением) возник спрос/предложение.

Ответ на вопрос «Что нам нужно получить?» создает спрос на материалы, которые необходимо поставить с другого подразделения. DRP рассчитывает полностью все эти потребности (после запуска MRP).

На вопрос «Что мы собираемся поставить?» ответ возникает при оценке всех источников спроса на продукт, включая заказы клиентов, прогноз отгрузок, потребности в запчастях, страховой запас и межзаводской спрос.

Используя данные по межзаводским запросам и заказам на распределение, между подразделениями ведется контроль спроса и предложения. На основе данных о потребностях подразделения на материалы, поставляемые другим подразделением, DRP создает запросы между этими подразделениями.

Ответ на последний вопрос «Что мы можем поставить?» зависит от наличия материалов (предложение) и транспорта (ресурсов). Если спрос (потребности) превышает предложение, DRP можно использовать для закрепления материалов за несколькими подразделениями в указанной пропорции.

Отметим «стандартные» понятия и определения, применяемые для описания MRP II-системы.

Основные входные данные в MRP II-системе следующие:

- данные изделия, включая BOM и маршрутизацию;
- данные потребности, сформированные MPS, из системы продаж и/или системы управления проектами;
- данные материального обеспечения, включая существующие материальные запасы, уже сделанный производственный заказ и заказы на приобретение

«Изделие» (Item) – базовое понятие MRP II-системы. Изделие – это может быть сырье, компонента, «сборка», или законченная продукция, или любая другая материальная «вещь». В русском языке для Item, так же как и для BOM, нет полностью адекватного перевода, в рамках данного пособия будем употреблять термины «Изделие» и «Компонента» как взаимозаменяемые эквиваленты Item. Все «компоненты» в пределах MRP-системы должны сначала быть определены путем создания «главной записи изделия». Главная запись изделия включает (точнее задает в рамках автоматизированной системы) большинство общих данных относительно каждого изделия, типа единиц измерения, всевозможного описания, уникального номера изделия и т.д. Данные, связывающие каждую компоненту изделия в главной записи изделия, используются всеми модулями, функциями и процессами в пределах MRP II-системы. Главная запись изделия, как правило, включает четыре компонента специфических для управления плановым процессом в MRP II-системе: тип изделия, политика заказа, метод заказа и система заказа. Отношения между типом изделия, политикой заказа, системой заказа и методом заказа очень важны в MRP II-системе.

MRP II-система может использовать пять типов изделия или более:

- производимое (производственное);
- покупное (заказное);
- обобщенное;
- фантомное (стоимостное);
- субподрядное (субподрядный договор).

Потребность (объем заказа) для производимых и покупных компонент может быть сформирована MPS, MRP, SIC, FAS и PS (проектной системой).

Цель процесса планирования потребности (MRP-процесса) состоит в преобразовании информации о спросе в производственный заказ, который будет доведен до управления цехом для исполнения, и в формировании распоряжений на закупку, на основании которых будет сформирован календарный план закупок.

Информация о спросе (потребности) может быть сформирована четырьмя источниками (рис. 3.14):

- заказы на продажу (включая полученные из заключенных контрактов и заказов на продажу, а в отдельных случаях и из коммерческих предложений);
- «запланированные» в системе MPS-заказы;

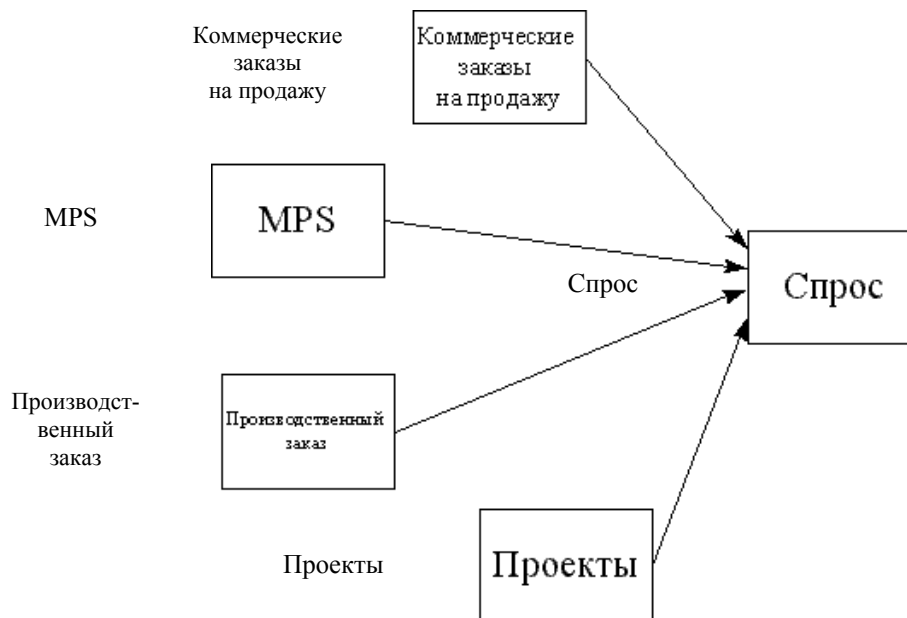


Рис. 3.14 Информация о спросе

- фактический производственный заказ (например, переходящий из предыдущих периодов);
- потребности из системы управления проектами (планирования проектов).

В некоторых бизнес-моделях потребность для вариантов изделия может быть сформирована процессом планирования потребности проекта, который является частью системы управления проектами. Следовательно, в зависимости от реализации MRP-системы заказы на закупку могут быть получены из MRP-потребности или непосредственно из системы управления закупками или, наконец, из функции управления проектами.

Источник потребности в отдельных компонентах зависит от политики заказа и системы заказа, к которой назначено изделие. Размер закупки для выдачи заказа на поставку (материально-техническое снабжение) вычисляется из заказов на закупку, производственного заказа, планов производства и уровней складских запасов.

Информация в главной записи изделия используется, чтобы выяснить, является ли политика заказа анонимной (anonymous) или «по заказу» (to order) или производственной (MRP), чтобы определить систему заказа, соответствующую изделию. Система заказа определяет, каким образом (в частности какой системой планирования) формируется потребность в данном компоненте готовой продукции.

Функция планирования потребности использует информацию в BOM, чтобы определить количество каждого материала, компоненты или сырья более низкого уровня, чем рассматриваемый на данном этапе планирования, необходимого, чтобы произвести запланированные количества каждого изделия данного более высокого уровня.

Например, запланированный производственный заказ в объеме 33 автомобилей поступает от функции MPS к функции планирования потребности. Состав изделия (BOM) показывает, что для каждого автомобиля требуются 4 колеса. Планирование потребности будет генерировать «запланированный» заказ приобретения (если это закупаемая компонента) или производственный заказ (если колеса планируются и производятся в рамках единого технологического процесса) в размере  $33 \times 4$ , т.е. 132 колеса. Частным случаем такой ситуации может быть вариант, когда требуемые колеса являются субконтрактным изделием, причем контрактором по данному заказу выступает «родственное» или «дочернее» предприятие.

Функция планирования потребности использует тип изделия, устанавливаемый в главной записи изделия для колеса, чтобы определить, планировать ли производственный заказ или заказ на закупку. Если тип изделия для колес установлен – «приобретаемое», то запланированный заказ на закупку будет сформирован. Если тип изделия установлен «производимое», плановый производственный заказ будет сформирован.

Хотя тип изделия определит, что необходимо сформировать: запланированный производственный заказ или запланированный заказ на закупку в ходе процесса планирования, диктуемый данным атрибутом выбор не окончательный. В случае необходимости производимые компоненты могут быть приобретены. Приобретаемые компоненты также, если это необходимо, могут быть произведены. Однако, MRP II-система не будет позволять определять BOM или маршрутизацию для данной компоненты или изделия, если тип изделия установлен как «приобретаемое». Следовательно, компонентам, которые могут быть как произведены, так и приобретены, должен быть назначен «производимый» тип изделия, так, чтобы было возможно использовать BOM и маршрутизацию для «дуального» планирования.

Процесс планирования потребности использует для каждого изделия «времена задержек» (lead time – интерпретируется и как «время опережения») приобретения или производства, чтобы определить, когда конкретно планировать их производство или приобретение с учетом особенностей логистического процесса.

В примере выше можно представить, что колеса назначены к «приобретаемому» типу изделия, и время задержки приобретения – две недели. Если производство требует семи дней, то процесс планирования потребно-

сти будет формировать запланированный заказ на закупку для колес, который будет передан исполнителю на 21 день прежде, чем автомобили должны быть отгружены. В различных системах может быть установлено различное количество времен задержки, учитывающих, например, такие особенности, как время доставки, время таможенных операций, время на обработку заказа контрагентом по поставкам.

Реально это тот же самый базисный процесс планирования потребности, используемый во всем производстве.

Чтобы улучшить точность планирования потребности, MRP-система допускает, чтобы даже заявки на продажу (т.е. неподтвержденные коммерческие предложения) были включены во входные данные процесса планирования потребности. При этом система должна давать возможность получить информацию о возможных сроках выполнения заказа с учетом уже спланированных работ (заказов) и дать прогноз (проект) реальной себестоимости товаров в заказе с учетом имеющихся запасов и/или прогноза цен на исходные материалы и комплектующие.

Конечно, маловероятно, что все заявки на продажу в конце концов превратятся в заказы.

Способность MRP II-системы включать заявки на продажу в процесс планирования потребности может быть очень полезна в специфических типах бизнеса, особенно когда период конечной сборки требует мало времени или когда бизнес-цикл очень короток. Этот метод, конечно, имеет ограничения. Действительно, если спецификации заявок сильно отличаются (например, принципиальное значение имеет цвет изделия в малых партиях), то оценка их реализации процентом становится бессмысленной. Ввиду этого целесообразно использовать данный механизм в случаях, когда спецификации заявленных и заказанных товаров «статистически» подобны. Системы современного уровня (COMMS-CSRP) позволяют работать с заказами на порядок более «тесно», что позволяет решить данную проблему. В частности, эти системы позволяют производить процесс «модельного» и затем «окончательного» планирования под каждый заказ клиента с целью получения адекватной информации о возможных сроках выполнения заказа и его реальной стоимости. Более того, при наличии модуля APS (advanced planning and scheduling – расширенное управление производственными заданиями) возможно производить процесс планирования хоть при каждом появлении нового заказа, а также устанавливать приоритеты выполнения заказов.

Как показано выше, объемно-календарный план (MPS) представляет собой список объемов производства на каждое запланированное изделие для каждого периода планирования, т.е. планировочная таблица является главной для следующих за ней операций планирования. MPS подготавливается на уровне управления бизнесом «в целом» или предприятием и обычно включает развернутые категории готовых изделий.

Однако, во многих случаях большинство из компонент (изделий), включенных в систему планирования MPS – «стандартные ( типовые, обобщенные) изделия».

Например, предприятие точно не знает, каковы будут спецификации закупаемых товаров, но для составления бизнес-плана необходимо провести модельное планирование. Тогда, например, используются «усредненные» по предыдущим периодам (или точнее – «спрогнозированные») спецификации.

Затем запускается функция модельного планирования потребности, чтобы подготовить детализированные, время-структурированные вычисления потребности в материалах и потребности в производственных мощностях, необходимые для того, чтобы удовлетворить плановой потребности MPS. Так как в «стандартной системе» такие расчеты достаточно трудоемки, то часто вместо них используются упрощенные методы, типа «чернового планирования производственных мощностей» (RCCP). На основании этих данных рассчитываются финансовые результаты объемно-календарного плана. В свою очередь, на основании произведенных расчетов принимается решение о пригодности плана к исполнению или о его изменении.

MRP II часто планируется для более коротких периодов времени, чем MPS. Например, если MPS обычно рассматривает месячные периоды времени как основу планирования, планирование потребности может быть основано на недельных или даже суточных (сменных) интервалах (периодах). Требуемые плановые периоды времени должны быть выбраны, основываясь на среднем производственном цикле и/или на среднем цикле продаж (динамике движения запасов). Так, если «в среднем», от момента предварительного согласования спецификации до заключения договора проходит несколько недель – то это требует одного горизонта планирования. Если же клиент готов забирать товар на следующий день или даже через несколько часов – то, соответственно, другого. При этом и в том, и в другом случае производственный цикл может составлять 1 – 2 дня.

После согласования предварительного MPS он превращается в «запущенный» (запланированный) объемно-календарный план. После этого проводится полноценная процедура планирования потребности в материалах и производственных мощностях.

Задача процесса планирования потребности – подготовить детализированные планы производства продукции и приобретения компонент и сборок, необходимых для удовлетворения объемно-календарного плана (MPS).

### 3.2 ERP-СИСТЕМА

В последние годы системы планирования класса MRP II в интеграции с модулем финансового планирования FRP (Finance Requirements Planning) получили название систем бизнес-планирования ERP (Enterprise Requirements Planning), которые позволяют наиболее эффективно планировать всю коммерческую деятельность современного предприятия, в том числе финансовые затраты на проекты обновления оборудования и инвестиции в производство новой линейки изделий. В российской практике целесообразность применения систем по-

добного класса обуславливается, кроме того, необходимостью управлять бизнес-процессами в условиях инфляции, а также жесткого налогового прессинга, поэтому системы ERP необходимы не только для крупных предприятий, но и для небольших фирм, ведущих активный бизнес. На рис. 3.15 представлена логическая схема системы планирования ресурсов производственного предприятия.

Схематично модель большинства ERP-систем можно описать следующим образом: в единую базу данных поступают все первичные сведения о деятельности предприятия, и на их основе программа строит различные отчеты, графики, прогнозы, словом, поставляет полноценную аналитическую информацию. Хозяйственные операции регистрируются в системе один раз, и их влияние на результативность работы предприятия можно оценить сразу, получив соответствующий отчет. Итак, основная ценность ERP-системы – в обеспечении информационной интеграции всех функциональных областей деятельности компании.

Несмотря на то, что возможности современных ERP-систем достаточно развиты и постоянно возрастают, чуда может и не произойти. Зачастую после внедрения КИС руководство по-прежнему не довольно качеством информационного обеспечения. Например, вопреки всем ожиданиям, не сокращаются трудозатраты на выполнение рутинных операций и, что еще важнее, сохраняются все недостатки, присущие ранее сложившейся практике осуществления производственно-хозяйственных операций. Речь обычно идет о некорректном оформлении



Рис. 3.15 Логическая структура системы планирования ресурсов предприятия

первичных документов, наличии сверхнормативных запасов, нарушениях в сбытовой политике, в частности об отпуске продукции клиентам, имеющим неисполненные обязательства, и т.д. Более того, нередко спроектированная ERP-система настолько сложна и неадекватна текущим задачам, что вообще не используется в компании. И это не единичные случаи! По некоторым данным, на Западе однозначно успешными считаются менее 50 % внедрений ERP-систем. Достоверных сведений по ситуации в России пока нет, но вряд ли тенденция будет отличаться.

Причин неудачных внедрений сотни. Но в их основе, как правило, лежит нарушение основополагающих принципов проектирования КИС. Обычно проекты внедрения ERP-систем не дают ожидаемых результатов вследствие их проектирования без учета стратегии развития бизнеса; нарушения принципа построения системы «сверху – вниз»; чрезмерного увлечения реинжинирингом бизнес-процессов и их порой неоправданного подчинения требованиям стандартной функциональности базовой

ERP-системы и, наоборот, вследствие кардинальной переработки базовой функциональности.

**Проектирование системы ERP без учета стратегии развития предприятия.** При настройке ERP-системы невозможно учесть все потенциальные пути развития предприятия в отдаленном будущем. За прошедшие годы экономическая среда, в которой работают российские предприятия, сильно изменилась. Например, в предприятиях нефтегазовой отрасли трансформировалась структура собственности: предприятия вывели практически все непрофильные активы, информация о которых была неотъемлемой частью КИС и учитывалась при составлении консолидированной отчетности. Многие предприятия металлургической отрасли заметно сократили численность сотрудников (некоторые практически в два раза), что естественно отразилось на количестве автоматизированных рабочих мест.

Понятно, что со временем ERP-системы, созданные в отрыве от планов реструктуризации бизнеса, потребуют кардинальной модернизации. Иначе они превратятся в обузу, мешающую текущему управлению и даже документообороту. Создание и внедрение полнофункциональной ERP-системы – длительный процесс, который на крупных предприятиях может занимать 3 и даже 5 лет. Более того, систему необходимо проектировать так, чтобы она работала в течение 2 – 3 лет без проведения модернизации. Поэтому при проектировании важно представлять структуру и масштабы бизнеса в перспективе, как минимум, на 3 года. Ошибки в прогнозировании

нии могут привести к неоправданно большим расходам, в частности на покупку дополнительного сетевого оборудования и оплату Интернет-трафика, составляющих значительную долю в стоимости владения ERP-системой. Совсем неприятный вариант, когда спустя год или два становится очевидна необходимость перевести ERP-систему на другую техническую платформу.

К числу других типовых направлений развития бизнеса можно отнести расширение практики мелкосерийного производства, создание филиальной сети, замену поставщиков, сокращение резервных запасов, ужесточение требований к срокам поставок. В этой связи производительность внутренних телекоммуникационных каналов должна быть рассчитана на повышенную нагрузку, например, когда поток данных возрастает из-за сокращения периодичности обновления информации. Если производительности каналов и самой базы данных недостаточно, то регистрация хозяйственных операций будет проходить менее оперативно. Следовательно, любые аналитические данные по текущей ситуации окажутся не совсем достоверными.

Заложить в ERP-систему цели предприятия и перспективы ее развития можно только при проектировании «сверху – вниз», а не наоборот. Создание информационной управленческой системы – удовольствие дорогое. Регистрация в ней всех данных, появляющихся на предприятии, в принципе невозможна. И естественно, каждый разработчик при проектировании сталкивается с необходимостью перехода от этого полного, в некотором смысле «неограниченного» объема информации к какому-то «лимиту». Поэтому, создавая ERP-систему, проектировщик всегда решает задачу выбора значимых для принятия управленческих решений данных в увязке с «ценой вопроса» на ее реализацию. На каждом предприятии ежедневно циркулируют огромные информационные потоки данных о материально-технических ресурсах, клиентах, персонале, производственном потенциале и т.д. Возникает вопрос: нужны ли в ERP-системе специфические сведения, скажем, о производительности какого-либо станка в последние 2 часа или о количестве полуфабрикатов на столе конкретного работника в текущий момент при том, что эта информация, бесспорно, используется в управленческой деятельности?

У каждого уровня управления – свои потребности в информационном обеспечении. Но эти данные ни в коем случае не должны оказаться избыточными.

Распределение информационных потоков будет верным, если начать построение системы с уточнения потребностей в сведениях верхних уровней управления, постепенно спускаясь «вниз». При таком подходе в первую очередь формируются и определяются показатели, необходимые высшему руководству, а также частота их расчета. Затем устанавливаются данные, требующиеся следующему в иерархии управленческому звену, и т.д. Таким образом исключается риск создания системы, которая будет генерировать информацию, недостаточную для принятия управленческих решений высшим руководством.

На практике проектировщики, не задаваясь целью обеспечить информационную поддержку принятия управленческих решений, либо пытаются ввести в систему максимальное количество данных, тем самым неоправданно увеличивая стоимость КИС, либо упускают часть важных для какого-то уровня управления сведений. В результате менеджмент страдает из-за недостаточности и несвоевременности информационного обеспечения.

Руководство предприятия в лучшем случае получает доступ к информационному пространству, содержащему огромные массивы данных. Но ему практически никогда не выдается точечная агрегированная информация, необходимая для принятия управленческих решений. Естественно, такая важная цель создания и внедрения ERP-системы, как усиление контроля, также не достигается.

На практике существует немало примеров, когда даже полнофункциональная КИС класса ERP не удовлетворяет потребности управленческого аппарата в информации. Например, руководитель предприятия при анализе ситуации с дебиторской задолженностью столкнулся со следующей проблемой: используемая в работе система могла предоставить лишь неструктурированный перечень дебиторов без какой-либо группировки по важности, по удельному весу в общем объеме задолженности, срокам и т.д.

Чтобы предприятие, затратив значительные средства, не получило в результате неэффективную из-за фрагментарности учетную систему, ERP-систему нужно проектировать, ориентируясь на цели компании, последовательно определять вид и характеристики информации, необходимой каждому уровню управления, начиная с высшего руководства.

**Избыточный реинжиниринг бизнес-процессов.** Достаточно часто предприятия, внедряющие ERP-систему, либо соглашаются на реинжиниринг всех бизнес-процессов и их подчинение требованиям базовой функциональности выбранной системы, либо настаивают на сохранении сложившейся практики работы и, соответственно, на кардинальной перестройке выбранной системы (а порой и на полном ее переписывании). Эти две крайности пополняют список причин неудач при создании и внедрении ERP-систем.

В первом случае велик риск того, что система, созданная в расчете на кардинальную перестройку бизнес-процессов, вообще не будет использоваться. Опыт показывает, что принципиальные изменения бизнес-процессов трудно и редко приживаются и совершенствовать систему управления компании все же лучше эволюционным путем.

Западные ERP-системы разработаны с учетом мирового опыта построения и оптимизации бизнес-процессов. Конечно же, все это должно учитываться при совершенствовании системы управления российских предприятий. Вместе с тем часто используемые проектировщиками ссылки на западную практику не совсем корректны, так как отечественные компании работают в другой экономической среде, и переход на западные стандарты не всегда целесообразен.

Во втором случае полученная система вследствие доработок и переработок теряет свою надежность. Соответственно, резко возрастают риски ошибочной обработки вводимой информации. Более того, никакой пользы от автоматизации неэффективных бизнес-процессов предприятию не будет. Наоборот, оно лишится возможности совершенствовать свою деятельность, так как будет заковано в жесткие рамки работы программы. В этой связи крайне важно правильно определить оптимальное соотношение между реинжинирингом бизнес-процессов и доработкой системы.

**Неверная оценка экономической эффективности внедрения ERP-системы.** Экономическая эффективность внедрения ERP-системы – это, наверное, самый сложный вопрос, на который предстоит ответить руководителю. Понятно, что внедрение подразумевает немалые затраты на общую автоматизацию (компьютеры, серверы, сетевое оборудование, лицензии, консультационные услуги и т.д.). В этой связи важно сопоставлять расходы на автоматизацию того или иного процесса, учитывая его место в ERP-системе, с итоговыми экономическими результатами проекта в целом. Необходимо ответить на вопрос, что даст ведение учета соответствующих операций в системе или предоставление таких-то данных такому-то менеджеру? Каких потерь это поможет избежать? Как повысить эффективность используемых ресурсов? Какие резервы позволит вовлечь в производственную деятельность? В противном случае возрастает риск того, что затраты на автоматизацию процессов не окупятся.

Отвечать на вопрос, какова цена включения какой-либо информации, необходимо на всех этапах проектирования ERP-системы. Сначала при определении ее функциональной структуры, выборе базовой платформы, технического обеспечения и других общих решений по системе на этапе разработки ее концепции, затем при составлении технического задания. Кроме того, задавать вопрос об экономической эффективности важно на этапе доводки прототипа системы до окончательного варианта.

При этом необходимо помнить, что наилучшие результаты от внедрения ERP-системы достигаются, если она проектируется для предприятия с хорошо выстроенной системой управления.

**Технология и практика проектирования ERP-систем.** На практике процесс проектирования выглядит следующим образом. Заказчик формулирует укрупненные требования к создаваемой системе, которые ложатся в основу Концепции Технического задания (ТЗ), разрабатываемые внешней проектной организацией. Этому процессу всегда предшествует анализ производственно-хозяйственной деятельности предприятия-заказчика. Цель изучения бизнес-процессов – определение «узких мест» в информационном обеспечении и выявление резервов для повышения эффективности работы компании. На основе вариантных проработок в Концепции определяется контур создаваемой системы, а именно: базовая ERP-система; функциональная структура; информационное обеспечение; количество необходимых автоматизированных рабочих мест; техническое обеспечение. При разработке Концепции особое внимание уделяется перспективам развития бизнеса заказчика. Рамки, в пределах которых важно понимать (представлять) перспективы развития бизнеса, определяются исходя из сроков разработки и внедрения системы (которые зависят от масштабов компании: от 6 месяцев до 3 лет); периода функционирования системы без необходимости ее модернизации (желательно, чтобы период морального старения системы составлял не менее двух лет). Сформированная на основе предпроектного обследования Концепция может содержать несколько основных альтернативных вариантов развития автоматизации системы управления заказчиком. Каждый вариант должен быть оценен исходя из соотношения «стоимость/эффективность». Выбор Концепции осуществляется заказчиком и служит основой для разработки ТЗ. Здесь детально прорабатываются требования к системе. ТЗ – основной документ, определяющий требования, организацию и проведение работ, в соответствии с которыми осуществляются проектирование ERP-системы и ее сдача заказчику. Следующий этап – разработка модели бизнес-процессов to be (бизнес-процессы в условиях функционирования ERP-системы). Модель создается на основе ТЗ и укрупненно описывает управленческие и информационные взаимосвязи в системе. Данный этап является ключевым в работах по созданию ERP-системы. Это объясняется тем, что его результаты позволяют сформировать у сотрудников и руководства предприятия-заказчика видение функционирования их предприятия в условиях использования ERP-системы и уточнить требования к ней. Для крупных предприятий целесообразна разработка эскизного проекта (до разработки модели бизнес-процессов to be), в котором определяются основные проектные решения (с количеством рабочих мест свыше 60). На основе ТЗ, проектных решений, утвержденных на этапе эскизного проектирования, и модели бизнес-процессов to be осуществляется рабочее проектирование. На данном этапе должен быть проведен анализ соответствия алгоритмов расчетов и методов управления, заложенных в программном обеспечении внедряемой ERP-системы, специфическим алгоритмам и методам, применяемым в практике функционирования компании-заказчика. По результатам анализа определяются «пробелы» в функциональности системы и принимаются необходимые проектные решения по их устранению. Эффективность проектов по созданию и внедрению комплексных автоматизированных систем управления тем выше, чем теснее сотрудничество заказчика и разработчика на всех этапах проектирования.

### 3.2.1 Логистика в ERP-системах

Одной из современных базовых управленческих технологий является логистика. Логистика – это наука о планировании, организации, контроле и регулировании движения материальных и информационных потоков в пространстве и времени от первичного источника до конечного потребителя.

В современном производстве и сбыте существенно возросло время прохождения товаров по различным каналам материально-технического обеспечения (свыше 90 % временных затрат), поэтому логистика делает



основные акценты на оценку эффективности производства в области ресурсного потенциала и экономии издержек обращения.

В применении к предприятию логистика представляет собой общую точку зрения (стратегическую, тактическую и операционную) на предприятие и его партнеров по бизнесу с материальным потоком в качестве интегратора. Интеграция различных функций товародвижения дает возможность устанавливать оптимальное соотношение интересов различных предприятий и подразделений предприятия, достигая на этой основе минимизации совокупных издержек и получая такой общий результат деятельности, который превосходит сумму отдельных эффектов.

Логистические системы реализуют методы решения задач закупочной, производственной, сбытовой и транспортной логистик на распределенной базе данных предприятия. Они позволяют решать множество задач, среди которых можно выделить:

- управление закупками сырья, материалов, товаров;
- учет в натуральной и денежной формах движения материалов, полуфабрикатов, продукции на складах и в подразделениях предприятия;
- анализ сбыта готовой продукции и продаж.

Упрощенный логистический взгляд на предприятие может быть представлен следующей диаграммой (рис. 3.16).

Обычно в качестве функциональных подсистем логистики различают:

- логистику материально-технического снабжения;
- логистику производства;
- логистику сбыта товара (маркетинговую логистику).



Рис. 3.16 Логистический взгляд на предприятие

По сферам объектов система логистики подразделяется на:

- производство;
- систему складского хозяйства;
- систему транспортировки, упаковки и обращение с материалами;
- информационную систему, включая обработку заказов и систему хранения.

Архитектура логистических цепочек предприятия зависит от взаимосвязи технологических цепочек и бизнес-процессов. Множество логистических цепочек предприятия, поддерживаемых в ERP-системах, представлено на рис. 3.17.

На уровне предприятия различаются:

- Закупочная логистика – обеспечение предприятия материальными ресурсами.
- Распределительная логистика – обеспечение рационализации процесса физического продвижения продукции к потребителю и формирование системы эффективного логистического сервиса.

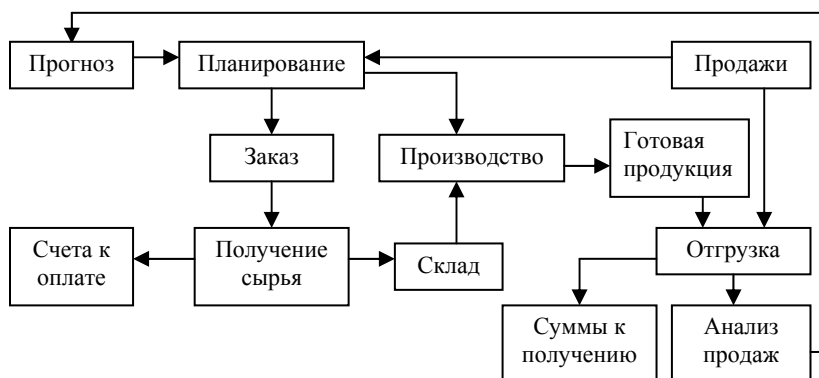


Рис. 3.17 Логистические цепочки предприятия

- Производственная логистика – обеспечение качественного своевременного и комплектного производства продукции в соответствии с хозяйственными договорами, сокращение производственного цикла и оптимизация затрат на производств.

- Складская логистика – операции, непосредственно связанные с переработкой и оформлением грузов и координацией со службами закупок и продаж, расчетом оптимального количества складов и места их расположения.

- Транспортная логистика – оптимизация транспортных систем, выбор вида и типа транспортных средств; определение разноканальных маршрутов доставки; обеспечение технологического единства транспортно-складского процесса.

В качестве примера рассмотрим распределительную логистику. С помощью ERP-систем можно наглядно сопоставить прогнозируемый и фактический спрос с поставками продукции, чтобы разрабатывать эффективные планы, обеспечивающие современную доставку. Благодаря единому информационному хранилищу и средствам обработки заказов ERP-системы предоставляют в реальном времени сведения о состоянии заказов, историю платежей, данные о кредитах и доставке.

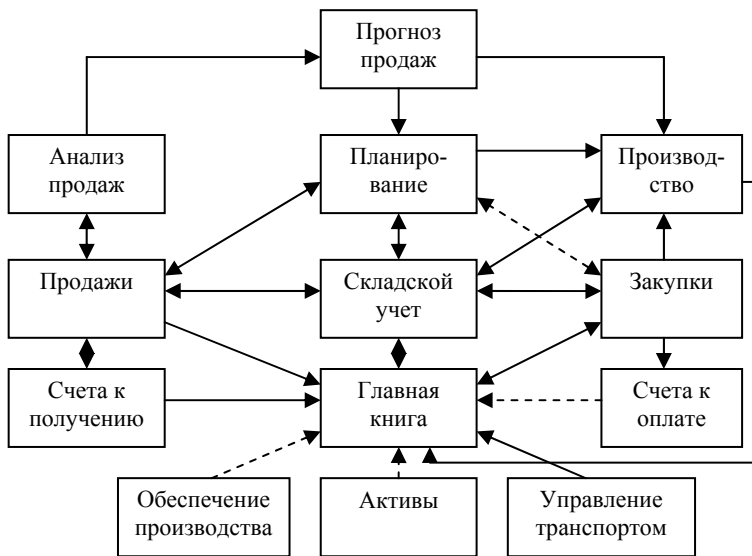
Распределительная логистика поддерживается следующими ERP-процессами, представленными на рис. 3.18. Цепочка состоит из следующих основных компонентов: обработка заказов на поставки, регистрация и анализ продаж, прогноз продаж, планирование ресурсов и материалов, управление запасами и складами, обработка запасов на приобретение, управление транспортом.

Обработка заказов на поставки позволяет повысить качество обслуживания потребителей продукции за счет быстрого приема заказов, осуществлять проверку поставок существующим запасам и запланированному производству.

Анализ продаж обеспечивает сбор статистических данных по количеству, цене и подробным затратам, а также возможности анализа и сравнения по продуктам или категориям продуктов или заказчикам для принятия решений и учета результатов при планировании.

В заключении данного пункта отметим, что составление оптимального плана перевозок – классическая математическая задача. Как известно, усилия, необходимые для ее решения, при увеличении размера задачи растут экспоненциально.

Поэтому вручную составить план, близкий к оптимальному или хотя бы достаточно экономичный, практически невозможно. Более того, если не использовать средства автоматизации, при значительных объемах перевозок, при больших количествах потребителей, складов,



**Рис. 3.18** Логистические цепочки распределительной логистики

возможных маршрутов, транспортных средств вообще вряд ли удастся вовремя доставить заказанную продукцию. Конечно, реальная ситуация далека от идеальной математической модели. Тем не менее, современные информационные технологии позволят, если не найти оптимальные решения (многочисленные случайные обстоятельства исключают возможность существования оптимального решения в математическом смысле), то значительно сократить транспортные расходы.

### 3.2.2 ЦЕНТРЫ ОБРАБОТКИ ДАННЫХ ДЛЯ ERP-СИСТЕМ

Для того чтобы ERP-система предприятия была способна выполнять стоящие перед ней бизнес-задачи, предоставляемые ею сервисы должны обладать требуемой функциональностью, производительностью и доступностью. Функциональность достигается за счет тщательного проектирования системы. Для обеспечения заданной производительности необходимо правильно определить параметры используемых технических средств. С этой целью специалисты, устанавливающие новую ERP-систему, либо проводят технический аудит той системы, которая уже имеется, измеряют ее производительность, моделируют будущие нагрузки, либо (например, когда система создается с нуля) рассчитывают значения параметров по специальным методикам, раз-

работанным производителями оборудования и ПО. Чтобы гарантировать высокую доступность (availability) сервисов ERP-системы, важно в первую очередь грамотно проработать архитектуру системы и разработать комплекс специальных мер.

Одновременно предприятие стремится по возможности сократить совокупную стоимость владения системой. Главными путями для этого являются, во-первых, внедрение системы эксплуатации, позволяющей удешевить администрирование ERP-системы и спланировать расходы на его модернизацию, а во-вторых, консолидация технических средств и обслуживающего персонала в центре обработки данных (ЦОД). Такой центр представляет собой комплексное организационно-техническое решение для создания высокопроизводительной и отказоустойчивой информационной инфраструктуры.

Информационные сервисы ERP-системы делятся на критичные для выполнения основных бизнес-функций и вспомогательные. Критичными считаются такие сервисы, недоступность которых в течение длительного времени способна повлечь серьезные потери, причем не только финансовые. Так, задержка в обслуживании клиентов может привести к ухудшению имиджа предприятия, из-за чего, в свою очередь, будут потеряны потенциальные или даже текущие клиенты.

Приемлемое время недоступности информационных сервисов зависит от требований бизнеса и для каждого предприятия определяется индивидуально. Оно варьируется от нескольких минут до нескольких часов, и обычно эта цифра фиксируется в соглашении об уровне сервиса (Service Level Agreement – SLA).

Все методы, применяемые для обеспечения высокого уровня доступности ERP-систем, взаимосвязаны, но условно их часто делят на две группы: архитектурные и организационные. Архитектурные методы можно охарактеризовать одним словом – дублирование. Дублирование сокращает время восстановления работы ERP-системы после сбоя. Оно может осуществляться на всех уровнях ИТ-инфраструктуры: установка резервных компонентов в серверы, выполняющие критичные функции, создание кластера таких серверов, резервное копирование информации и создание резервного центра для ЦОД ERP-системы. Организационные методы предназначены для построения такого процесса эксплуатации, который свел бы до минимума время обнаружения и устранения неисправностей в ERP-системе. К ним относятся мониторинг, план восстановления после аварий и тренинг персонала.

Кластером называется группа соединенных между собой серверов (рис. 3.19), которые функционируют как единое целое и работа которых координируется для решения поставленной задачи. Результатом



Рис. 3.19 Схема кластера

создания кластера может стать, например, обеспечение высокой доступности информационного сервиса (так называемые кластеры высокой доступности, или HA-кластеры – от англ. high availability) или распараллеливание процесса решения сложной вычислительной задачи. Современное кластерное ПО позволяет избежать длительного простоя системы в случае сбоя или выхода из строя оборудования, ошибки в программе или ошибочного действия обслуживающего персонала при условии, что не произошло разрушения данных. Восстановить разрушенные данные кластер не может, что делает необходимым создание высоконадежной системы хранения данных, включающей подсистему резервного копирования.

Кластер состоит из аппаратной и программной частей. Аппаратная часть, помимо узлов кластера, т.е. самих объединенных серверов, включает среду их взаимодействия – heartbeat, реализуемую с помощью либо обычного Ethernet, либо специальных высокоскоростных сред передачи данных с низкой латентностью, таких как SCI, HP HyperPlex, Sun WildCat. Для того чтобы все узлы кластера могли обеспечить работу одного информационного сервиса, они должны иметь доступ к общему дисковому пространству. Оно реализуется созданием общих для узлов кластера томов на одном отказоустойчивом дисковом массиве (или двух зеркалированных дисковых массивах), подключенном ко всем узлам кластера. Для кластера, состоящего из многих узлов, наилучшей инфраструктурой доступа серверов к общему дисковому массиву является сеть хранения данных (SAN), поскольку протокол SCSI не обеспечивает устойчивой работы шины более чем с двумя инициаторами.

Высокая доступность информационных сервисов, предоставляемых узлами кластера, обеспечивается кластерным ПО. Это ПО с помощью специальных сервисов или скриптов отслеживает работоспособность информационных сервисов, выполняемых узлами кластера. В случае сбоя, вызванного отказом диска, сетевого интерфейса или самого приложения, кластерное ПО переносит соответствующий сервис на другой узел. Под переносом здесь понимается: остановка приложения (если оно еще работало) на первом узле, размонтирование общих дисковых томов, монтирование их на втором узле, перенос IP-адреса (алиаса) с первого на второй узел, запуск приложения. Если в кластере больше двух узлов, то информационные сервисы вышедшего из строя узла переносятся на другие узлы в зависимости от правил, либо заданных администратором, либо определенных самим кластерным ПО на основе данных о загрузке работоспособных узлов.

Система резервного копирования представляет собой служебную подсистему системы хранения данных (СХД) и является обязательным компонентом решения по обеспечению высокой доступности ERP-системы. Она позволяет восстановить работоспособность информационных сервисов даже в тех случаях, когда повреждены данные.

Создание централизованной системы резервного копирования дает возможность сократить (по сравнению с децентрализованной) совокупную стоимость владения ИТ-инфраструктурой за счет оптимального использования аппаратуры и сокращения расходов на администрирование. Такая система имеет многоуровневую архитектуру, включающую:

- сервер управления резервным копированием;
- один или несколько серверов копирования данных, к которым подключены устройства резервного хранения данных;
- компьютеры-клиенты с установленными на них программными агентами резервного копирования;
- консоль администратора системы резервного копирования.

Так как система резервного копирования относится к числу служебных, т.е. нагрузка на вычислительные средства, которую она создает, не является полезной с точки зрения предоставления информационных сервисов, эту нагрузку желательно уменьшить. Данная задача распадается на две: сокращение так называемого «окна резервного копирования» (т.е. времени, в течение которого компьютер-клиент выполняет резервное копирование) и уменьшение трафика соответствующих данных в корпоративной ЛВС. Внедрение системы резервного копирования в составе СХД позволяет сократить «окно» благодаря интеграции со средствами создания РИТ-копий, реализованными в современных дисковых массивах: с данных практически мгновенно делается «моментальный снимок», и резервное копирование выполняется уже с этого снимка, а сервер продолжает работу. Снизить нагрузку на локальную сеть помогут технологии LAN-free backup и Serverless backup, предоставляемые SAN-сетями.

Рассмотренные выше методы обеспечения высокой доступности инвариантны к типу информационной системы и применимы для всех ERP-систем. Целесообразность применения того или иного метода зависит от критичности конкретной системы для бизнеса предприятия и объема финансовых потерь в случае ее простоя. Однако при выборе конкретных мер необходимо учесть архитектурные особенности конкретного ERP-решения. Для того чтобы сократить время возможного простоя, в системе необходимо выявить единые точки отказа (Single Point Of Failure – SPOF) и постараться их ликвидировать путем применения перечисленных выше методов.

Рассмотрим ERP-систему на базе продукта SAP R/3. Одним из вариантов внедряемых архитектур SAP R/3 является классическая схема из трех серверов – «продуктивного», «тестового» и «сервера разработки» (рис. 3.20).

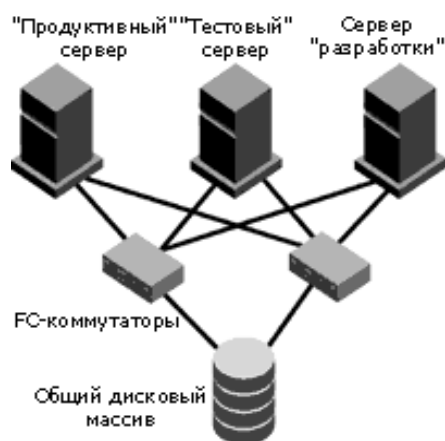


Рис. 3.20 Пример архитектуры ERP-системы на базе SAP R/3

Очевидно, что критичным для работы ERP-системы является «продуктивный» сервер. Снизить время его возможного простоя можно, продублировав все те его компоненты, которые являются SPOF. Но это не всегда

осуществимо и зависит от модели используемого сервера. Другой вариант – продублировать работу «продуктивного» сервера на одном из двух оставшихся – «тестовом» или «разработки». Для этого как минимум необходимо, чтобы «продуктивный» и дублирующий серверы имели доступ к одним и тем же данным, например, были подключены через SAN к общему дисковому массиву. Если у предприятия недостаточно квалифицированных кадров, чтобы организовать круглосуточное дежурство администраторов, которые смогут в случае неисправности «продуктивного» сервера перевести работу модулей SAP R/3 на дублирующий, рекомендуется создать кластер высокой доступности, способный делать это автоматически.

Ряд модулей SAP, особенно продукты серии mySAP, рекомендуется развертывать на серверном комплексе с трехуровневой архитектурой: клиент, сервер приложений, сервер баз данных (предыдущий рассмотренный вариант имел двухуровневую архитектуру – клиент и сервер). Трехуровневая архитектура характерна и для ERP-систем, построенных на основе продуктов Oracle Applications. В этой архитектуре сервер приложений является менее критичным элементом, чем сервер баз данных, отвечающий за работу с самым ценным элементом системы – данными. Если сбой сервера может быть ликвидирован путем перезагрузки (это довольно частый случай), то сервер приложений перезагрузится и продолжит работу намного быстрее, чем сервер баз данных, которому необходимо будет запускать СУБД и в случае аварийной остановки СУБД выполнять восстановление баз. Серверов приложений может быть несколько, и они могут дублировать работу друг друга без использования кластерных технологий. Организовать дублирование сервера приложений проще, чем сервера баз данных, поскольку серверы приложений не хранят информацию, а обращаются за ней к серверам баз данных. Таким образом, в системе с трехуровневой архитектурой основное внимание необходимо обратить на серверы баз данных и именно для них применить такие методы повышения доступности, как дублирование компонентов и кластеризация.

Еще одной особенностью реализации ERP-систем на основе продуктов SAP является то, что для ряда модулей SAP, например BW и HR, рекомендуется использовать выделенные серверы. Архитектура системы в таком случае будет двухуровневой и содержать несколько серверов. Если для обеспечения высокой доступности такой ERP-системы использовать кластерную технологию, то оптимальным будет кластер с топологией N+1, где один сервер резервирует работу остальных и в случае выхода из строя одного из них берет на себя выполнение его функций. Когда в качестве платформы ERP-системы применяется один из промышленных вариантов UNIX, проблем нет: все основные производители (Sun, HP и IBM) предлагают кластерные решения, поддерживающие топологию N+ 1 для большого числа (как минимум до восьми) узлов. Однако для платформы MS Windows NT/2000 кластер с четырьмя узлами поддерживается только в варианте ОС DataCenter и для фиксированного числа аппаратных конфигураций. Альтер-нативой в данном случае может служить кластер VERITAS Cluster Server 2.0 for Windows, который поддерживает конфигурации до 32 узлов.

### 3.2.3 ERP II-система

По определению, данному Gartner Group, ERP II – это бизнес-стратегия предприятия, принадлежащего к определенной отрасли, а также набор ключевых для данной отрасли программных приложений, помогающих клиентам и акционерам предприятий увеличивать стоимость бизнеса за счет эффективной ИС и оптимизации операционных и финансовых процессов как внутри своего предприятия, так и во внешнем мире, т.е. в рамках сотрудничества с другими предприятиями.

Основная идея ERP II заключается в выходе за рамки задач по оптимизации внутренних процессов организации предприятия: кроме интеграции традиционных для ERP-систем областей деятельности предприятия. Системы класса ERP II позволяют управлять взаимоотношениями с клиентами (телефонные звонки, визиты, рассылки рекламных и маркетинговых материалов, качество и скорость послепродажного обслуживания и т.д.), цепочками поставок (координация работы служб маркетинга, сбыта и снабжения), вести торговлю через Интернет.

До недавнего времени функциональностью ERP II обладали лишь некоторые интегрированные системы управления для крупных предприятий (mySAP.com., Oracle E-Business Suite и др.). Высокая стоимость и сложность внедрения этих систем делали их недоступными для большинства предприятий среднего масштаба.

В настоящее время сформировался рынок систем класса ERP II, ориентированных на средние предприятия, и развивающиеся предприятия получили возможность пользоваться всеми преимуществами ERP II, внедряя системы средней ценовой категории, не только не уступающие, а во многом превосходящие по функциональности и гибкости своих «тяжелых» собратьев. В то же время очевидно, что средние предприятия стремятся к более «низкому» ценовому порогу, более быстрому и ресурсосберегающему внедрению и гибкости систем при эксплуатации. Эти требования заставляют разработчиков применять новые технологии, которые и находят свое воплощение в новых системах класса ERP II, более гибких, более доступных и масштабируемых.

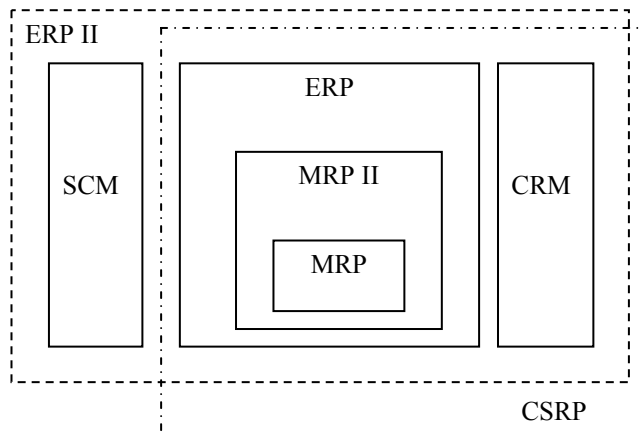
На рис. 3.21 приведена структурная схема ERP II-системы, которая является частным случаем обобщенной схемы, показанной на рис. 1.4.

Корпоративные ресурсы, охватываемые CSRP-системой (Customer Synchronized Resource Planning), обслуживают такие этапы производственной деятельности, как проектирование будущего изделия с учетом специфических требований заказчика, гарантийное и сервисное обслуживание.

По прогнозам Gartner Group системы SCM и CRM станут необходимыми модулями ERP II-систем.

SCM (Supply Chain Management) – система управления цепочкой поставок, т.е. к системам класса SCM можно отнести все решения, которые способствуют выработке стратегии, координации планирования и управления в сфере поставок, производства, складирования и доставки товаров.

Для ведения производственной деятельности большинству предприятий необходимы разные материалы и ресурсы. На этом уровне система SCM решает задачу взаимодействия с поставщиками: их поиск, оформление заказов, взаиморасчеты и т.д. При этом задача может решаться с помощью специальных АСУ обеспечения и электронных торговых площадок (e-commerce). SCM должна предоставлять мощный аналитический модуль, который позволяет определить фактические потребности – в объемах закупок для обеспечения производственного



**Рис. 3.21 Структурная схема ERP II-системы**

процесса. Система SCM может помочь определить оптимальный объем выпуска продукции, а также поддерживать процесс принятия соответствующих тактических решений о производственных мощностях и развитии производства (интеграция с ERP) – пользуясь данными спроса на продукцию и предложения от поставщиков (интеграция с CRM).

CRM (Customer Relations Management) – система управления отношениями с клиентами, основанная на применении новых управленческих и информационных технологий, с помощью которых предприятия аккумулируют знания о клиентах для выстраивания взаимовыгодных отношений с ними. Подобные отношения способствуют увеличению прибыли, так как привлекают новых клиентов и помогают удержать старых.

CRM реализуется с помощью специального набора программного обеспечения (приложений) и технологий, позволяющих автоматизировать, а значит, совершенствовать бизнес-процессы в сфере продаж, маркетинга и обслуживания клиентов. Это дает возможность предприятию обращаться к заказчикам услуг с интересными предложениями в наиболее удобный момент времени и по наиболее удобным каналам связи.

Во многих предприятиях отделы продаж, маркетинга и обслуживания клиентов пока еще действуют независимо друг от друга, и по этой причине их представления о заказчике зачастую противоречивы, а действия – несогласованны. Система CRM облегчает координацию действий различных отделов, обеспечивая их общей платформой для взаимодействия с клиентами, и дает каждому из них доступ к полной информации о них, что способствует наилучшему удовлетворению потребностей клиентов.

Основой системы CRM являются приложения:

- автоматизации продаж (Sales Force Automation – SFA), которые способствуют: увеличению прибыли; повышению точности прогнозов (планирования), а также эффективности контактов с клиентами при осуществлении продаж; повышению вероятности заключения сделки; снижению издержек на продажи;
- автоматизации маркетинга (Marketing Automation – MA), которые позволяют провести адресный маркетинг; увеличить количества маркетинговых каналов за счет использования возможностей Интернета;
- поддержки клиентов (Customer Service & Support – CSS): снижение издержек на службу поддержки, улучшение предоставляемого сервиса, повышение удовлетворенности клиентов, преобразование службы поддержки из затратного в прибыльный отдел;
- автоматизации продаж (Sales Force Automation – SFA). На них возлагаются следующие функции: ведение календаря событий и планирование работы; работа с клиентами (каждый клиент будет обслужен на высочайшем уровне, благодаря зафиксированной истории взаимодействия с ним); мониторинг потенциальных продаж (ни одна потенциальная возможность не будет упущена, каким бы плотным не было расписание сотрудника); автоматическая подготовка коммерческих предложений (освобождает сотрудников от рутинной работы); предоставление информации о ценах; автоматическое обновление данных о размере бонуса в зависимости от выполнения поставленных задач; формирование отчетов (эффективный инструмент автоматического создания отчетов по результатам деятельности).

Gartner Group дает некоторые рекомендации тем предприятиям, которые намерены использовать систему CRM. Рекомендации зависят от типа предприятия:

- тип А: агрессивно внедряющие новые технологии;
- тип В: умеренно внедряющие новые технологии;

- тип С: консервативно относящиеся к новым технологиям.

Gartner Group советует предприятиям типа А и В не искать единое приложение, удовлетворяющее всем потребностям в области продаж, маркетинга и сервиса. Им следует использовать предложения нескольких ведущих поставщиков, наилучшим образом решающих конкретные задачи (например, приложение автоматизации фронт-офиса может быть от одного поставщика, приложение для разработки и проведения маркетинговых кампаний – от другого, приложение обработки вызовов – от третьего). При этом рекомендуется пользоваться помощью консультантов для интеграции указанных приложений и рабочих процессов.

Предприятия, относящиеся к типу С, должны выбирать интегрированные приложения, обладающие достаточной функциональностью во всех трех сферах и способные обеспечить потребности компании на ближайшие три года.

Е-commerce (электронная коммерция) – это совокупность технических и организационных форм ведения коммерческой деятельности и совершения операций с использованием электронных систем и сети Интернет, как средство взаимодействия с партнерами, банком, поставщиками и потребителями товаров и услуг. Как правило, в системах е-commerce имеют место все этапы совершения операций: поиск нужной продукции, уточнение деталей операции, оплата, получение (доставка) заказа.

Вариантом электронной коммерции между предприятиями являются системы В2В (см. рис. 1.4). Такие системы используются для организации обеспечения предприятия и сбыта готовой продукции.



Рис. 3.22 Структурная схема электронной торговой площадки

На рис. 3.22 приведена типичная структура открытой системы электронного снабжения – электронной торговой площадки.

Электронные торговые площадки (ЭТП) подразделяются на buyer-driven, supp-lier-driven и third-party-driven, т.е. площадки, созданные продавцом, покупателем и третьей стороной соответственно.

В качестве создателей, а также основных пользователей ЭТП, обычно выступают: крупные поставщики, осуществляющие продажи большому числу корпоративных клиентов; розничные сети, закупающие товары крупным оптом; компании, организующие посредническую деятельность в сфере электронной торговли.

Различают четыре типа моделей third-party-driven: онлайн-каталог, аукцион, биржа, сообщество.

Модель «онлайн-каталог» представляет собой собрание «витрин» нескольких продавцов, на которых выставлены товары с фиксированными ценами.

Модель «аукцион» отличается от каталога аукционной системой формирования цены.

Третья модель – «биржа» – подразумевает саморегуляцию цены текущим спросом и предложением.

В модели «сообщество» участников объединяет общий интерес. Статья доходов в данной модели – взносы участников, рекламные поступления и спонсорское участие.

Для запуска В2В-продаж в сети можно либо создать собственную ЭТП, либо стать участником уже существующей. При создании собственной ЭТП выбор поставщика программно-аппаратного комплекса сводится к поиску компромисса между ценой и соответствием предлагаемой системы ряду условий. Среди основных требований, предъявляемых к современной ЭТП, можно отнести: доступность и комфортность работы, масштабируемость (рост числа участников не должен сказываться на скорости работы), интеграционные возможности, безопасность (надежный механизм идентификации и шифрования).

Из зарубежных поставщиков ПО отметим компании Oracle, Ariba, CommerceOne.

Выбор российского поставщика ПО может оказаться оптимальным решением, поскольку предлагаемые системы ориентированы на специфику отечественной e-коммерции и стоят меньше зарубежных аналогов. В качестве примера можно привести компанию B2R, предлагающую спектр решений для ведения межкорпоративного бизнеса в Internet.

### 3.3 ДРУГИЕ МЕТОДИКИ

Метод планирования и управления Just-in-time (JIT – точно вовремя) появился в Японии на предприятиях автомобильного концерна в 50-х гг. XX в. Он охватывает проектирование изделий, выбор поставщиков, обеспечение качества, планирование, учет производства и контроль (с использованием специальных бирок-ярлыков Kanban). Одна из важнейших концепций метода «точно вовремя» связана с минимизацией страховых и межоперационных заделов за счет стабилизации поставок, а также обеспечения резерва производственных мощностей. Метод «точно вовремя» не противоречит MRP и MRP II и часто предлагается в современных системах как одна из форм организации производства. Однако до сих пор он не соответствовал традиции отечественной промышленности, так как именно заделы и запасы сырья служат буфером от нестабильности поставок, смежников и растущих цен комплектующих изделий. Кроме того, в России считалось, что полезнее повышать значение коэффициента использования оборудования (вместо создания задела мощностей), чем рационально планировать объем межоперационных заделов, а эти два показателя – взаимосвязаны.

Методы OPT (Optimised Production Technology – оптимизированная тех-нология производства) созданы в Израиле в 1970-х гг. (работы Эли Голдрайт). На их основе был разработан ряд программных пакетов. Методы OPT предназначены для максимизации выпуска продукции при сокращении объема запасов и производственных затрат. В их основе лежит определение «узких мест» (производственных мощностей или материальных ресурсов) и наиболее точный их учет при планировании. Методика оценки «узких мест» сохраняет актуальность и применяется в алгоритмах планирования и определения ресурсов производственных мощностей MRP II.

Концепция компьютеризированного интегрированного производства (CIM, Computer Integrated Manufacturing) возникла в начале 1980-х гг. и связана с интеграцией гибкого производства и систем управления им. CIM с точки зрения систем управления и планирования (в качестве которых используются ERP и MRP II) предполагает интеграцию всех подсистем системы управления (управления снабжением, проектированием и подготовкой производства; планирования и изготовления; управления производственными участками и цехами; управления транспортно-складскими системами; управления обеспечением оборудованием, инструментом и оснасткой; систем обеспечения качества, сбыта, а также финансовых подсистем) [APICS92].

Методы CALS (Computer-aided Acquisition and Logistics Support – ком-пьютерная поддержка процесса поставок и логистики) [CALS99] возникли в 1980-х гг. в военном ведомстве США для повышения эффективности управления и планирования в процессе заказа, разработки, организации производства, поставок и эксплуатации военной техники. CALS предусматривает однократный ввод данных, их хранение в стандартных форматах, стандартизацию интерфейсов и электронный обмен информацией между всеми организациями и их подразделениями – участниками проекта. Методы доказали свою эффективность и переносятся в настоящее время на «гражданские» отрасли промышленности. Новая концепция сохранила аббревиатуру CALS с более широким смыслом (Continuous Acquisition and Life circle Support – поддержка непрерывного жизненного цикла продукции). Проводится стандартизация ряда аспектов CALS в международной организации стандартизации ISO. Методы CALS могут использоваться вместе с MRP II / ERP и CIM. В отличие от них CALS позволяет управлять всем жизненным циклом продукции, включая маркетинг, управление комплексными проектами, обслуживание при эксплуатации.

### ЗАКЛЮЧЕНИЕ

---

В условиях рыночной экономики предприятия нуждаются в решении задач управления на качественно новом уровне. Необходимость оперативного реагирования на конъюнктуру рынка и быстро меняющуюся экономическую ситуацию требует перестройки внутренней микроэкономики предприятий, постановки управленческого учета, оптимизации процессов управления.

В учебном пособии обсуждены основные задачи и особенности системного подхода к автоматизации управления предприятием, рассмотрены преимущества и недостатки различных подходов к автоматизации, используемых сегодня.

Рассмотрены теоретические и практические аспекты выбора системы комплексной автоматизации (так как выбор конкретного варианта КИС – один из самых ответственных моментов автоматизации управления бизнесом) и приводятся примеры применения современных информационных технологий для реализации элементов интегрированной системы управления предприятием.

Основная задача учебного пособия – дать студентам базовый уровень знаний, необходимых сегодня для эффективного управления работами по автоматизации финансово-хозяйственной деятельности предприятий.



## СПИСОК ЛИТЕРАТУРЫ

---

- 1 Хаммер М., Чампи Д. Реинжиниринг корпорации: манифест революции в бизнесе: Пер. с англ. СПб.: Изд-во Санкт-Петербургского университета, 1997. 332 с.
- 2 Гайфулин Б.Н., Обухов И.А. Автоматизированные системы управления предприятиями стандарта ERP/MRP II. М.: Богородский печатник, 2001. 104 с.
- 3 Логистика: Учеб. пособие / Под ред. Б.А. Аникина. М.: ИНФРА-М, 1997. 304 с.
- 4 Зиндер Е.З. Бизнес-реинжиниринг и технологии системного проектирования: Учеб. пособие. М.: Центр Информационных Технологий, 1999. 267 с.
- 5 Создание информационной системы предприятия // "Computer Direct". 1996. № 2.
- 6 Туровец О.Г. Организация производства на предприятиях в условиях кризисного развития // Организатор производства. Воронеж, 1996. № 2. С. 21 – 22.
- 7 Попов В.Н. Проблемы сохранения и развития отечественного товарного производства // Организатор производства, региональный выпуск. Пенза, 2000. № 1. С. 5 – 22.
- 8 Васкевич Д. Стратегии клиент/сервер. Руководство по выживанию для специалистов по реорганизации бизнеса: Пер. с англ. Киев: Диалектика, 1996. 384 с.
- 9 Волчков С.А. Мировые стандарты управления промышленным предприятием в информационных системах (ERP-системах) // Организатор производства. Воронеж: Международная академия науки и практики организации производства, 1999. № 1. С. 43.
- 10 Головки М.В. Проекты ИС для крупных предприятий: от бессистемного управления к системам управления знаниями // Директору информационной службы. 2000. № 4. С. 2.
- 11 Кутыркин С.Б., Волчков С.А., Балахонова И.В. Повышение качества предприятия с помощью информационных систем класса ERP // Методы менеджмента качества. 2000. № 4. С. 8.
- 12 Судов Е.В. CALS-технологии, или Информационная поддержка жизненного цикла продукта // PC Week. 1998. № 45.
- 13 Вендров А.М. CASE-технологии. Современные методы и средства проектирования информационных систем. М.: Финансы и статистика, 2000. 267 с.
- 14 Калянов Г.Н. Теория и практика реорганизации бизнес-процессов. М.: СИНТЕГ, 2000. 254 с.
- 15 Гаврилов Д.А. Управление производством на базе стандарта MRP II. СПб.: Питер, 2002. 320 с.
- 16 Гламаздин Е.С., Новиков Д.А., Цветков А.В. Управление корпоративными программами: информационные системы и математические модели. М.: ИПУ РАН, 2003. 159 с.
- 17 ГОРИН С.В., ТАНДОЕВ А.Ю. ПРИМЕНЕНИЕ CASE-СРЕДСТВА ERWIN 2.0 ДЛЯ ИНФОРМАЦИОННОГО МОДЕЛИРОВАНИЯ В СИСТЕМАХ ОБРАБОТКИ ДАННЫХ // СИСТЕМЫ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ. 1996. № 1. С. 11 – 14.
- 18 МАКЛАКОВ С. НОВЫЕ ВОЗМОЖНОСТИ СА ВРWIN 4.0 // КОМПЬЮТЕРПРЕСС. 2001. № 3. С. 11 – 14.
- 19 МАКЛАКОВ С. ИНСТРУМЕНТАЛЬНАЯ ПОДДЕРЖКА РАЗРАБОТКИ И ВНЕДРЕНИЯ КОРПОРАТИВНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ // КОМПЬЮТЕРПРЕСС. 2001. № 9. С. 5 – 7.
- 20 Чен Э. ERP II: жизнь после жизни // PC Week/RE. 2001. № 25. С. 20 – 23.
- 21 Соколов Н. Скромное обаяние ERP II. Тенденции развития ERP-систем // Компьютера. 2002. № 12. С. 5 – 8.
- 22 Бирюков В., Дрожжинов В. Введение в CRM // PC Week/RE. 2001. № 25. С. 9 – 12.
- 23 Уотерс Д. Логистика. Управление цепью поставок. М.: ЮНИТИ-ДАНА, 2003. 503 с.
- 24 Ойхман Е.Г., Попов Э.В. Реинжиниринг бизнеса: Реинжиниринг организаций и информационных технологий. М.: Финансы и статистика, 1999. 198 с.
- 25 Сапунцов В.Д., Лысенко М.А., Султанов Ф.Я. Применение CASE-средств BR-win и Erwin для проектирования информационных систем / Под ред. В.Д. Сапунцова. М.: РГУ Нефти и газа, 2000. 53 с.

- 26 Программное обеспечение систем автоматизации // Автоматизация в промышленности. 2003. № 1. С. 7 – 10.
- 27 Насакин Р. Плюсы и минусы В2В Интернет-коммерции // Sales. 2005. № 3. С. 56 – 60.