

Лекция 10-11. Структурный подход к моделированию бизнес-процессов

ПЛАН

10-11.1 Сущность структурного подхода

10-11.2 Структурная модель предметной области

10-11.3 Процессный подход в описании предметной области

10-11.4 Методология структурного моделирования и проектирования SADT

10-11.5 Метод функционального моделирования IDEF0

10-11.6 Метод процессного моделирования IDEF3

10-11.7 Моделирование потоков данных

10-11.8 Построение DFD-диаграмм

10-11.9 Особенности применения функциональных и объектно-ориентированных методологий моделирования предметной области

Виды и объем учебной работы

Виды занятий и аттестаций	Объем занятий и количество аттестаций		
	всего	6 семестр	7 семестр
Всего занятий, час	216	108	108
Всего аудиторных занятий, час.	80	42	38
в том числе			
лекции, час.	36	18	18
лабораторный практикум, час.	40	20	20
Всего самостоятельной работы студента, час.	91	57	34
в том числе			
курсовой проект, час.	24		24

Содержание разделов дисциплины, виды занятий и их объем

№ п. п.	Наименование разделов дисциплины	Объем занятий, час.		
		Лекции	Лаб. практикум	СРС
6с-1	Введение в дисциплину «Управление жизненным циклом информационных систем». Основные особенности современных проектов ИС, технологии и стандарты.	6		20
2	Жизненный цикл проекта ИС. Методологические основы проектирования ИС.	6	12	20
3	Технологии проектирования ИС. Принципы организации информационного обеспечения ИС.	6	8	17
Итого		18	20	57
7с-4	Структурный и объектно-ориентированный подходы к моделированию бизнес-процессов	6	4	3
5	CASE-технологии проектирования автоматизированных информационных систем. Инструментальные средства проектирования информационных систем	6	8	15
6	Проектирование корпоративных информационных систем на базе готовых решений. Методология и инструментальные средства поддержки проектирования открытых и распределенных систем	6	8	16
Итого		18	20	34
Всего		36	40	91

Рекомендуемые учебно-методические издания и иные информационные источники

Основная литература

1. Александров, Д. В. Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы : учебное пособие / Д. В. Александров.— М. : Финансы и статистика, 2011 .— 224 с. -<URL: http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=5306>
2. Балдин К.В., Уткин В.Б. Информационные системы в экономике. – М.: Издательско-торговая корпорация "Дашков и К", 2012. – 395 с.
-<URL:http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=3591>.
3. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2006. – 544 с.
4. Мартынов В.В., Никулина Н.О., Филосова Е.И. Проектирование информационных систем: учебное пособие.- Уфа:УГАТУ,2008.-379с.
5. Лабораторный практикум по дисциплине «Проектирование информационных систем». Часть 1 и Часть 2. Сост. Е.И.Филосова. - Уфа, 2008.-117с.

Дополнительная литература и иные информационные источники

1. Маклаков С. В. Моделирование бизнес-процессов с ALLFusion PM / С. В. Маклаков - М.: Диалог-МИФИ, 2008 - 224 с. – <URL:http://www.library.ugatu.ac.ru/pdf/diplom/Maklakov_modelirovanie_2008.pdf>.
2. Блюмин А.М., Печеная Л.Т., Феоктистов Н.А. Проектирование систем информационного, консультационного и инновационного обслуживания: Учебное пособие - М.: Издательско-торговая корпорация "Дашков и К", 2010 - 352 с. –ISBN 978-5-394-00685-2.
3. Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Проектирование информационных систем. – М.: Изд-во «Интернет-Ун-т Информ. Технологий», 2005. – 304 с.
4. Информационные системы и технологии в экономике и управлении: Учебник / Под ред. проф. В.В. Трофимова.- М.: Высшее образование, 2006. – 480 с.
5. Федотова Д.Э., Семенов Ю.Д., Чижик К.Н. CASE-технологии: Практикум. – М.: Горячая линия-Телеком, 2005. – 160 с.

10-11.1. Сущность структурного подхода

Окружающий нас мир представляет собой огромную единую систему, которая, в свою очередь состоит из множества менее крупных систем. Мы выделяем системы в природе и обществе. Систему можно понимать как совокупность взаимосвязанных и взаимодействующих компонент. Часто реальные системы имеют значительные размеры и сложность. Вследствие этого работа с такими системами и проведение с ними экспериментов часто становится или очень затруднительным или даже невозможным. Чтобы сделать возможным или хотя бы облегчить решение реальных задач, связанных с реальными системами, необходимо создавать модели систем.

Модель – это наиболее полное и точное описание системы, которое позволяет получить ответы на вопросы относительно системы. Необходимость изучения реальных систем и создания моделей систем потребовала разработки соответствующей методологии.

Структурный анализ (Structured analysis) – это основанная на моделировании, ориентированная на процессы техника, используемая для анализа существующей системы, определения требований новой системы или того и другого.

Модели представляются диаграммами, которые иллюстрируют компоненты системы: процессы и связанные с ними входы, выходы и файлы. Сложные объекты анализируются и декомпозируются на более простые и более документированные.

Структурное проектирование – это ориентированная на процессы техника для разбиения больших программ на иерархию модулей, в результате чего программа легче реализуется и изменяется.

Структурное проектирование отыскивает факторы программы в нисходящей иерархии модулей, которые имеют следующие **свойства**:

- Модули должны иметь сильную внутреннюю связность. Каждый модуль должен выполнять одну и только одну функцию. Это позволяет многократно использовать модули в будущих программах.
- Модули должны быть слабо связаны между собой; иными словами, модули должны быть минимально зависимы между собой. Это минимизирует эффект влияния будущих изменений одного модуля на другой.
- Группировать все модули (или процессы) вызванные одной операцией для формирования операционного центра. Например, все задачи, выполняемые при получении заказа от поставщика, связаны. Часто центр управления служит как модуль управления.

Сущность структурного подхода

Структурным анализом принято называть метод исследования системы, которое начинается с ее общего обзора, а затем детализируется, приобретая иерархическую структуру с все большим числом уровней. Для таких методов характерно: разбиение на уровни абстракции с ограниченным числом элементов (от 3 до 7); ограниченный контекст, включающий только существенные детали каждого уровня; использование строгих формальных правил записи; последовательное приближение к результату. Структурный анализ основан на двух базовых принципах – **«разделяй и властвуй»** и принципе **иерархической упорядоченности**. Решение трудных проблем путем их разбиения на множество меньших независимых задач (так называемых «черных ящиков») и организация этих задач в древовидные иерархические структуры значительно повышают понимание сложных систем. Определим **ключевые понятия** структурного анализа.

Сущность структурного подхода

- **Операция** – элементарное (неделимое) действие, выполняемое на одном рабочем месте.
- **Функция** – совокупность операций, сгруппированных по определенному признаку.
- **Бизнес-процесс** — связанная совокупность функций, в ходе выполнения которой потребляются определенные ресурсы и создается продукт (предмет, услуга, научное открытие, идея), представляющая ценность для потребителя.
- **Подпроцесс** – это бизнес-процесс, являющийся структурным элементом некоторого бизнес-процесса и представляющий ценность для потребителя.
- **Бизнес-модель** – структурированное графическое описание сети процессов и операций, связанных с данными, документами, организационными единицами и прочими объектами, отражающими существующую или предполагаемую деятельность предприятия.
- Существуют различные **методологии** структурного моделирования предметной области, среди которых следует выделить **функционально-ориентированные** и **объектно-ориентированные методологии**.

10-11.2. Структурная модель предметной области

На начальных этапах создания ИС необходимо понять, как работает организация, в которой планируется внедрение ИС. Никто в организации не знает, как она работает в той мере подробности, которая необходима для создания ИС. Руководитель хорошо представляет принципы работы организации в целом, но не в состоянии вникнуть в детали работы каждого рядового сотрудника. Рядовой сотрудник хорошо знает суть своей работы, но плохо осведомлен об обязанностях коллег и вышестоящего руководства. Поэтому для описания работы организации необходимо построить модель бизнес-процессов, адекватную предметной области. Эта модель в дальнейшем послужит основой проектирования ИС данной организации.

Предварительное моделирование предметной области позволяет сократить время и сроки проведения проектировочных работ и получить более эффективный и качественный проект. Без проведения моделирования предметной области велика вероятность допущения большого количества ошибок в решении стратегических вопросов, приводящих к экономическим потерям и высоким затратам на последующее перепроектирование системы. Вследствие этого все современные технологии проектирования ИС основываются на использовании методологии моделирования предметной области.

К моделям предметных областей предъявляются следующие требования:

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

Для реализации перечисленных требований, как правило, строится **комплекс моделей**, который отражает структурный и оценочный аспекты функционирования предметной области.

Для отображения **структурного аспекта** моделей предметных областей в основном используются графические методы, которые должны гарантировать представление информации о компонентах системы. Главное требование к графическим методам документирования — простота. Графические методы должны обеспечивать возможность декомпозиции спецификаций системы с максимальной степенью детализации и согласований описаний на смежных уровнях декомпозиции.

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

Структурный аспект предполагает построение:

- **объектной структуры**, отражающей состав взаимодействующих в процессах материальных и информационных объектов предметной области;
- **функциональной структуры**, отражающей взаимосвязь функций (действий) по преобразованию объектов в процессах;
- **структуры управления**, отражающей события и бизнес-правила, которые воздействуют на выполнение процессов;
- **организационной структуры**, отражающей взаимодействие организационных единиц предприятия и персонала в процессах;
- **технической структуры**, описывающей топологию расположения и способы коммуникации комплекса технических средств.

Оценочные аспекты моделирования предметной области связаны с разрабатываемыми показателями эффективности автоматизируемых процессов, к которым относятся:

- время решения задач;
- стоимостные затраты на обработку данных;
- надежность процессов;
- экономические показатели эффективности, (объемы производства, производительность труда, оборачиваемость капитала, рентабельность и т.д.).

Для расчета показателей эффективности, как правило, используются статические методы функционально-стоимостного анализа (АВС) и динамические методы имитационного моделирования.

В основе различных методологий моделирования предметной области ИС лежат принципы последовательной детализации абстрактных категорий. Одной из важнейших специфических особенностей, отличающих ИС от техн.систем, является тесная связь с внешней средой. Поэтому при разработке ИС обычно модели строятся на трех уровнях:

- на внешнем уровне (определении требований);
- на концептуальном уровне (спецификации требований);
- на внутреннем уровне (реализации требований).

Внешнее проектирование формулирует цель и критерий эффективности будущей системы, выявляет ограничения – создается, экспериментально проверяется и корректируется модель системы. Определяются границы системы; фиксируются факторы внешней среды, имеющие значение для системы; определяются существенные связи, виды входных сигналов, на которые должна реагировать система; связанные с ними изменения выходных параметров. Иными словами, это этап выяснения взаимодействия системы с внешней средой, определения того, **что** и **зачем** будет делать система, **почему** она должна действовать так, а не иначе. На внешнем уровне определяется состав основных компонентов системы: объектов, функций, событий, организационных единиц, технических средств.

Концептуальное проектирование определяет содержание самой системы, определяется характер взаимодействия компонентов системы. **На концептуальном уровне** модель отвечает на вопрос, *как* должна функционировать система, *кто, где, когда* будет выполнять необходимые операции и процедуры?

На внутреннем уровне модель отвечает на вопрос, *какими способами и средствами* будет система выполнять свои функции, с помощью каких программно-технических средств реализуются требования к системе?

С позиции жизненного цикла ИС описанные уровни моделей соответственно строятся на этапах анализа предметной области, логического (технического) и физического (рабочего) проектирования.

Внешнее и внутреннее проектирование не являются самостоятельными, независимыми друг от друга этапами, они пересекаются и требуют взаимного согласования. Общая идея заключается в том, что сначала проводится внешнее проектирование для некоторых идеальных, ничем не ограниченных внутренних возможностей системы, затем – в первом приближении внутреннее проектирование, выявляя при этом ограничения, не позволяющие системе функционировать так, как это требуется в результате предварительного внешнего проектирования. Согласование заключается в изменении либо требований внешнего проектирования, либо ограничений внутреннего, либо того и другого. После такого согласования переходят к детальной, углубленной проработке вопросов внутреннего проектирования.

Рассмотрим **особенности построения моделей предметной области на трех уровнях детализации.**

1. Объектная структура

Объект — это сущность, которая используется при выполнении некоторой функции или операции (преобразования, обработки, формирования и т.д.). Объекты могут иметь динамическую или статическую природу: динамические объекты используются в одном цикле воспроизводства, например заказы на продукцию, счета на оплату, платежи; статические объекты используются во многих циклах воспроизводства, например, оборудование, персонал, запасы материалов.

На внешнем уровне детализации модели выделяются основные классы материальных объектов (например, сырье и материалы, полуфабрикаты, готовые изделия, услуги) и основные классы информационных объектов или документов (например, заказы, накладные, счета и т.д.).

На концептуальном уровне построения модели предметной области уточняется состав классов объектов, определяются их атрибуты и взаимосвязи. Таким образом, строится обобщенное представление структуры предметной области.

Далее концептуальная модель на **внутреннем уровне** отображается в виде файлов базы данных, входных и выходных документов ИС. Причем динамические объекты представляются единицами переменной информации или документами, а статические объекты — единицами условно-постоянной информации в виде списков, номенклатур, справочников, классификаторов.

2. Функциональная структура

Функция (операция) представляет собой некоторый преобразователь входных объектов в выходные. Последовательность взаимосвязанных по входам и выходам функций составляет бизнес-процесс. Функция бизнес-процесса может порождать объекты любой природы (материальные, денежные, информационные). Причем бизнес-процессы и информационные процессы, как правило, неразрывны, то есть функции материального процесса не могут осуществляться без информационной поддержки. Например, отгрузка готовой продукции осуществляется на основе документа "Заказ", который, в свою очередь, порождает документ "Накладная", сопровождающий партию отгруженного товара.

На внешнем уровне моделирования определяется список основных бизнес-процессов (направлений деятельности). Обычно таких бизнес-процессов насчитывается 15–20.

На концептуальном уровне выделенные бизнес-процессы декомпозируются и строятся иерархии взаимосвязанных функций.

На внутреннем уровне отображается структура информационного процесса в компьютере: определяются иерархические структуры программных модулей, реализующих автоматизируемые функции.

3. Структура управления

При выполнении бизнес-процесса возможны альтернативные или циклические последовательности действий в зависимости от различных условий протекания процесса. Эти условия связаны с происходящими событиями во внешней среде или в самих процессах и с образованием определенных состояний объектов (например, заказ принят, отвергнут, отправлен на корректировку). **События** вызывают выполнение функций, которые, в свою очередь, изменяют состояния объектов и формируют новые события, и т. д., пока не будет завершен некоторый бизнес-процесс. Тогда последовательность событий составляет конкретную реализацию бизнес-процесса.

Каждое событие описывается с двух точек зрения: **информационной** и **процедурной**. **Информационно** событие отражается в виде некоторого сообщения, фиксирующего факт выполнения некоторой функции, изменения состояния объекта или появления нового состояния. **Процедурно** событие вызывает выполнение новой функции, и поэтому для каждого состояния объекта должны быть заданы описания этих вызовов. Таким образом, события выступают в связующей роли для выполнения функций бизнес-процессов.

На внешнем уровне определяются **список внешних событий**, вызываемых взаимодействием предприятия с внешней средой (платежи налогов, процентов по кредитам, поставки по контрактам и т.д.), и **список целевых установок**, которым должны соответствовать бизнес-процессы (регламент выполнения процессов, поддержка уровня материальных запасов, уровень качества продукции и т.д.).

На концептуальном уровне устанавливаются бизнес-правила, определяющие условия вызова функций при возникновении событий и достижении состояний объектов.

На внутреннем уровне выполняется формализация бизнес-правил в виде триггеров или вызовов программных модулей.

4. Организационная структура

Организационная структура представляет собой совокупность организационных единиц, как правило, связанных иерархическими и процессными отношениями. Организационная единица — это подразделение, представляющее собой объединение людей (персонала) для выполнения совокупности общих функций или бизнес-процессов. В функционально-ориентированной организационной структуре организационная единица выполняет набор функций, относящихся к одной функции управления и входящих в различные процессы. В процессно-ориентированной структуре организационная единица выполняет набор функций, входящих в один тип процесса и относящихся к разным функциям управления.

На внешнем уровне строится структурная модель предприятия в виде иерархии подчинения организационных единиц или списков взаимодействующих подразделений.

На концептуальном уровне для каждого подразделения задается организационно-штатная структура должностей (ролей персонала).

На внутреннем уровне определяются требования к правам доступа персонала к автоматизируемым функциям информационной системы.

5. Техническая структура

Топология определяет территориальное размещение технических средств по структурным подразделениям предприятия, а **коммуникация** — технический способ реализации взаимодействия структурных подразделений.

На внешнем уровне модели определяются типы технических средств обработки данных и их размещение по структурным подразделениям.

На концептуальном уровне определяется способы коммуникаций между техническими комплексами структурных подразделений: физическое перемещение документов, машинных носителей, обмен информацией по каналам связи и т.д.

На внутреннем уровне строится модель «клиент-серверной» архитектуры вычислительной сети.

Описанные модели предметной области нацелены на проектирование отдельных **компонентов ИС**:

- данных (объектная структура),
- функциональных программных модулей (функциональная структура),
- управляющих программных модулей (структура управления),
- программных модулей интерфейсов пользователей (организационная структура),
- структуры технического комплекса (техническая структура).

Для более качественного проектирования указанных компонентов требуется построение моделей, увязывающих различные компоненты ИС между собой. В простейшем случае в качестве таких моделей взаимодействия могут использоваться матрицы перекрестных ссылок: "объекты—функции", "функции—события", "организационные единицы—функции", "организационные единицы—объекты", "организационные единицы—технические средства" и т.д.

10-11.3. Процессный подход в описании предметной области

Экономическая информационная система предназначена, как известно, в первую очередь, для решения проблем бизнеса. Требования к ней формируются на основе бизнес-модели данного бизнеса. Моделирование бизнес-процессов является важной составной частью проектов по созданию крупномасштабных информационных систем. Отсутствие таких моделей является одной из главных причин неудач многих проектов.

Модели бизнес-процессов являются не просто промежуточным результатом, используемым консультантом для выработки каких-либо рекомендаций и заключений. Они представляют собой самостоятельный результат, имеющий большое практическое значение.

На сегодняшний день в моделировании бизнес-процессов преобладает процессный подход. **Основной принцип процессного подхода заключается в структурировании деятельности организации в соответствии с ее бизнес-процессами, а не организационно-штатной структурой.** Именно бизнес-процессы, формирующие значимый для потребителя результат, представляют ценность, и именно их улучшением предстоит в дальнейшем заниматься. Модель, основанная на организационно-штатной структуре, может продемонстрировать лишь хаос, царящий в организации (о котором в принципе руководству и так известно, иначе оно бы не инициировало соответствующие работы), на ее основе можно только внести предложения об изменении этой структуры. С другой стороны, модель, основанная на бизнес-процессах, содержит в себе и организационно-штатную структуру предприятия.

Процессные потоковые модели — это модели, описывающие процесс последовательного преобразования материальных и информационных потоков компании в ходе реализации какой-либо производственной функции или функции управления. На верхнем уровне модели описывается логика взаимодействия участников процесса, на нижнем — технология работы отдельных специалистов на своих рабочих местах.

Процессные потоковые модели отвечают на вопросы кто—что—как—кому.

Современное состояние экономики характеризуется переходом от традиционной позадачной модели деятельности компании, к модели процессной. Позадачная модель деятельности построена на принципах разделения труда, узкой специализации и жестких иерархических структурах. Процессная модель основана на интеграции работ вокруг бизнес-процессов.

Главными недостатками позадачного подхода являются:

- разбиение технологий выполнения работы на отдельные фрагменты, иногда между собой несвязанные, которые выполняются различными структурными подразделениями;
- отсутствие целостного описания технологий выполнения работы;
- сложность увязывания простейших задач в технологию, производящую реальный товар или услугу;
- отсутствие ответственности за конечный результат;
- высокие затраты на согласование, налаживание взаимодействия, контроль и т. д.;
- отсутствие ориентации на клиента.

Когда в начале 90-х гг. стало ясно, что именно эти недостатки влияют на снижение эффективности управления экономическими объектами, появилось понятие **«реинжиниринга бизнес-процессов»**, введенное М. Хаммером и Дж. Чампи. Оно предусматривает радикальное перепроектирование бизнес-процессов предприятий для достижения резких, скачкообразных улучшений показателей их деятельности: стоимости, качества, сервиса, темпов развития на базе новых информационных технологий.

Необходимость реинжиниринга связывается с высокой динамичностью современного делового мира. Непрерывные и довольно существенные изменения в технологиях, рынках сбыта и потребностях клиентов стали обычным явлением, и компании, стремясь сохранить свою конкурентоспособность, вынуждены непрерывно перестраивать корпоративную стратегию и тактику. Дело в том, что принцип разделения труда, послуживший основой успешного развития бизнеса в течение последних двухсот лет (эффективная организация железных дорог в Северной Америке в 1820-х годах; конвейер Генри Форда; принципы организации управления большими компаниями Альфреда Слоуна, внедренные в Дженерал Моторс, и др.), исходит из предположения об относительной стабильности существующих технологий, а также постоянно растущем спросе на товары и услуги, при котором потребитель не имеет широкого выбора и довольствуется уже самим наличием продукции. Поэтому наиболее эффективной оказалась иерархическая пирамидальная структура компаний, организованных по функциональному признаку. Управление построено на административно-командных принципах. При этом клиентам отводится самый нижний уровень иерархии, где они представлены безликим «массовым потребителем». Однако с развитием современных технологий исчезла стабильность, а с ростом конкуренции изменилась и роль потребителя. Соревнование между производителями привело к дроблению массового рынка на относительно небольшие ниши, где уже потребитель диктует свои условия производителям, а не наоборот.

Решение проблемы повышения эффективности бизнеса – в смене основных принципов организации и управления и в переходе к ориентации не на функции, а на процессы.

Чтобы пояснить, каким образом проведение реинжиниринга бизнес-процессов повышает эффективность работы компании, рассмотрим, **как реинжиниринг изменяет реконструируемые бизнес-процессы.**

1. Несколько рабочих процедур объединяются в одну. Для перепроектированных процессов наиболее характерно отсутствие технологии "сборочного конвейера", в рамках которой на каждом рабочем месте выполняются простые задания, или рабочие процедуры. Выполнявшиеся различными сотрудниками, теперь они интегрируются в одну - *происходит горизонтальное сжатие процесса*. Кроме того, при традиционной организации трудно, а иногда и невозможно определить ответственного за быстрое и качественное выполнение работы. По имеющимся оценкам, горизонтальное сжатие ускоряет выполнение процесса примерно в 10 раз.

2. Исполнители принимают самостоятельные решения. В ходе реинжиниринга компании осуществляют не только горизонтальное, но и *вертикальное сжатие процессов*. Это происходит за счет самостоятельного принятия решения исполнителем, в тех случаях, когда при традиционной организации работ он должен был обращаться к управленческой иерархии.

При традиционной организации работ, ориентированной на выпуск массовой продукции, исходили из предположения, что исполнители не имеют ни времени, ни знаний, необходимых для принятия решений. Реинжиниринг отвергает эти предположения, что вполне естественно при отказе от массового производства и современном уровне образования. Наделение сотрудников большими полномочиями и увеличение роли каждого из них в работе компании приводит к значительному повышению их отдачи.

3. Шаги процесса выполняются в естественном порядке. Реинжиниринг процессов освобождает от линейного упорядочивания рабочих процедур, свойственного традиционному подходу, позволяя распараллеливать процессы там, где это возможно.

4. Процессы имеют различные варианты исполнения. Традиционный процесс ориентирован на производство массовой продукции для массового рынка, поэтому он должен исполняться единообразно, независимо от исходных условий при всех возможных входах процесса. В наше время высокая динамичность рынка приводит к тому, что процесс должен иметь различные версии исполнения в зависимости от конкретной ситуации, состояния рынка и т.д.

5. Работа выполняется в том месте, где это целесообразно. В традиционных компаниях она организуется по функциональным подразделениям: отдел заказов, транспортный отдел и т.п., и если, например, конструкторскому отделу требуется новый карандаш, то он обращается с заявкой в отдел заказов. Тот находит производителя, договаривается о цене, размещает заказ, осматривает товар, оплачивает его и передает конструкторам - все это достаточно расточительно и медленно. Проведенный в одной из компаний США анализ показал, что при традиционном распределении работ внутренние затраты компании на приобретение батарейки стоимостью 3 долл. составили 100 долл. Установлено, что 35% всех заказов составляют заказы стоимостью менее 500 долл. После проведения реинжиниринга отделы перешли к самостоятельному заказу дешевых товаров.

6. Уменьшается количество проверок и управляющих воздействий. Проверки и управляющие воздействия непосредственно не производят материальных ценностей, поэтому задача реинжиниринга – сократить их до экономически целесообразного уровня. Традиционные процессы насыщены подобными шагами, единственное назначение которых – контроль за соблюдением исполнителями предписанных правил. К сожалению, на практике довольно часто оказывается, что стоимость проверок и управляющих воздействий превосходит стоимость заказа требуемого продукта. Реинжиниринг предлагает более сбалансированный подход. Вместо проверки каждого из выполняемых заданий перепроектированный процесс часто агрегирует эти задания и осуществляет проверки и управляющие воздействия в отложенном режиме, что заметно сокращает время и стоимость процессов.

7. Минимизируется количество согласований. Еще один вид работ, не производящих непосредственных ценностей для заказчика, - это согласования. Задача реинжиниринга состоит в минимизации согласований путем сокращения внешних точек контакта. Как и в п.5, речь идет о стирании граней между функциональными подразделениями.

Итак, понятие реинжиниринга бизнес-процессов легло в основу процессного подхода для описания деятельности компании.

Процессный подход предполагает смещение акцентов от управления отдельными структурными элементами на управление сквозными бизнес-процессами, связывающими деятельность всех структурных элементов. Каждый деловой процесс проходит через ряд подразделений, т. е. в его выполнении участвуют специалисты различных отделов компании. Чаще всего приходится сталкиваться с ситуацией, когда собственно процессами никто не управляет, а управляют лишь подразделениями. Более того, структура компаний строится без учета возможностей оптимизации деловых процессов, обеспечивающих необходимые функции. *Процессный подход* позволяет устранить фрагментарность в работе, организационные и информационные разрывы, дублирование, нерациональное использование финансовых, материальных и кадровых ресурсов.

Процессный подход к организации деятельности предприятия предполагает:

- широкое делегирование полномочий и ответственности исполнителям;
- сокращение уровней принятия решений;
- сочетание принципа целевого управления с групповой организацией труда;
- повышенное внимание к вопросам обеспечения качества;
- автоматизацию технологий выполнения бизнес-процессов.

Согласно стандарту "Основные Положения и Словарь — ИСО/ОПМС 9000:2000" (п. 2.4) понятие "**Процессный подход**" определяется как: "Любая деятельность, в которой используются ресурсы для преобразования входов в выходы, может рассматриваться как процесс. Чтобы результативно функционировать, организации должны определять и управлять многочисленными взаимосвязанными и взаимодействующими процессами. Часто выход одного процесса образует непосредственно вход следующего. Систематическая идентификация и менеджмент применяемых организацией процессов, и особенно взаимодействия таких процессов, могут считаться "процессным подходом".

Основной принцип процессного подхода определяет структурирование бизнес-системы в соответствии с деятельностью и бизнес-процессами предприятия, а не в соответствии с его организационно-штатной структурой. Именно бизнес-процессы, обеспечивающие значимый для потребителя результат, представляют ценность и для специалистов, проектирующих ИС.


Процессная модель экономического объекта должна строиться с учетом следующих положений:

Верхний уровень модели должен отражать только контекст деятельности (т.е., контекстная диаграмма должна отражать взаимодействие моделируемого предприятия с внешним миром).

На втором уровне должны быть отражены тематически сгруппированные бизнес-процессы предприятия и их взаимосвязи в виде основных направлений деятельности.

Каждое из направлений деятельности должно быть детализировано на бизнес-процессы.

Детализация бизнес-процессов осуществляется посредством бизнес-функций.



Бизнес-функции описываются последовательностью элементарных технологических операций.

Описание элементарной операции осуществляется с помощью миниспецификации.

Процессный подход требует комплексного изучения различных сторон жизни организации — правовых основ и правил деятельности, организационной структуры, функций и показателей результатов их исполнения, интерфейсов, ресурсного обеспечения, организационной культуры. В результате анализа создается модель деятельности "как есть". Обработка этой модели с помощью различных аналитических методов позволяет проверить, насколько деловые процессы рациональны, а также определить, является ли та или иная операция ориентированной на общественно значимый конечный результат или излишней бюрократической процедурой.

В ходе анализа деловых процессов детально исследуются сферы ответственности подразделений ведомства, его руководителей и сотрудников. Это позволяет установить адреса владельцев деловых процессов, в результате чего процессы перестают быть бесхозными, создаются условия для разработки и внедрения систем стимулирования и ответственности за конечные результаты, определяются моменты и процедуры передачи ответственности. Анализ и оценка деловых процессов позволяют подойти к обоснованию стандартов их выполнения, допустимых рисков и диапазонов свободы принятия решений исполнителями, предельных нормативов затрат ресурсов на единицу эффекта.

Однако чисто "процессная компания" является скорее иллюстрацией правильной организации работ. В действительности все бизнес-процессы компании протекают в рамках организационной структуры предприятия, описывающей функциональные компетентности и отношения.


Процесный подход в описании предметной области

- **Реинжиниринг бизнес-процессов** (М. Хаммер и Дж. Чампи) - предусматривает радикальное перепроектирование бизнес-процессов предприятий для достижения резких, скачкообразных улучшений показателей их деятельности: стоимости, качества, сервиса, темпов развития на базе новых информационных технологий.
- **Процесный подход** (ИСО/ОПМС 9000:2000) - "Любая деятельность, в которой используются ресурсы для преобразования входов в выходы, может рассматриваться как процесс. Чтобы результативно функционировать, организации должны определять и управлять многочисленными взаимосвязанными и взаимодействующими процессами. Часто выход одного процесса образует непосредственно вход следующего. Систематическая идентификация и менеджмент применяемых организацией процессов, и особенно взаимодействия таких процессов, могут считаться "процесным подходом".


Модель процесса




Процесс включает в себя




Владельца процесса – должностное лицо, имеющее в своем распоряжении ресурсы процесса, с определенными правами, зоной ответственности и полномочиями.




Технологии процесса – порядок выполнения деятельности по преобразованию входов в выходы.



Системы показателей процессов – показатели продукта, показатели эффективности процесса, показатели удовлетворенности потребителей.



Управление процессом – деятельность владельца процесса по анализу данных процесса и принятию управленческих решений.



Ресурсы процесса – информация и материальные средства, которые владелец распределяет в ходе планирования работ по процессу и учитывает при расчете эффективности процесса как соотношение затраченных ресурсов на полученный результат процесса.

Формирование модели процесса. Документирование бизнес-процесса

№	Описание модели процесса
1	Входы/поставщики процесса
2	Выходы/клиенты процесса
3	Ресурсы: персонал, оборудование, информация, среда
4	Технология выполнения процесса
5	Все этапы цикла управления (планирование, выполнение, контроль, анализ, принятие решений)
6	Контрольные точки для измерения показателей
7	Возможные отклонения от нормального хода процесса
8	Показатели процесса, продукта и данные удовлетворенности клиентов
9	Участие руководителя (управление процессом)

Понятие реинжиниринга

Реинжиниринг – кардинальная перестройка деловых процессов для достижения радикального скачкообразного существенного улучшения деятельности фирмы

Направление управленческой мысли, возникшее на стыке менеджмента и информатизации

Инструмент конструирования и перестройки хозяйственной деятельности на основе процессного подхода

Методика фундаментальной реорганизации бизнеса

Революция в бизнесе, превратившая его конструирование в инженерную задачу

Цель реинжиниринга

Целью реинжиниринга бизнес-процессов (РБП) (BPR - Business process reengineering) является целостное и системное моделирование и реорганизация материальных, финансовых и информационных потоков, направленная на упрощение организационной структуры, перераспределение и минимизацию использования различных ресурсов, сокращение сроков реализации потребностей клиентов, повышение качества их обслуживания.

Сущность реинжиниринга

Реинжиниринг – «фундаментальное переосмысление и радикальное перепроектирование бизнес-процессов для достижения существенных улучшений в таких ключевых для современного бизнеса показателях результативности, как затраты, качество, уровень обслуживания и оперативность» - Майкл Хаммер, Джеймс Чампи «Реинжиниринг корпорации: манифест для революции в бизнесе»

Ключевые характеристики реинжиниринга

1. Фундаментальность	В процессе реинжиниринга определяется, ЧТО, ПОЧЕМУ , а затем КАК следует делать – стратегия (фундамент) бизнеса
2. Радикальность	Реинжиниринг предполагает отказ от существующих структур и процедур, создание бизнеса заново
3. Существенность	В результате реинжиниринга достигается существенное (в десятки и сотни раз) улучшение результатов деятельности
4. Резкость изменений	Процедура реинжиниринга длится около 9-15 месяцев
5. Бизнес-процесс	Происходит переход к использованию в модели управления процессного подхода, отражающегося в орг. структуре

Задачи реинжиниринга

включают объединение информационных ресурсов структурных подразделений компании и создание интегрированной корпоративной информационной системы управления, функционирующей в реальном масштабе времени, базирующейся на объективных данных о финансовых и материальных потоках по всем сферам хозяйственной деятельности фирмы, обеспечивающей общее снижение затрат и имеющей возможность гибкого реагирования на изменения рыночной ситуации

Ошибочные мнения относительно реинжиниринга

**Реинжинирингом
не является**

Внедрение информационных технологий для автоматизации бизнес-процессов

Изменение программного обеспечения устаревших информационных систем

Реструктуризация и уменьшение размеров бизнеса

Дебюрократизация организационной структуры управления

Внедрение глобального управления качеством

Виды реинжиниринга



Вид реинжиниринга	Ситуация для применения
1. Кризисный реинжиниринг (перепроектирование бизнес-процессов)	1. Состояние глубокого кризиса (потеря конкурентоспособности, отказ потребителей от товаров и т. п.)
2. Реинжиниринг развития (совершенствование бизнес-процессов)	2. Удовлетворительное текущее положение при нежелательных тенденциях и неблагоприятных прогнозах
	3. Благополучная ситуация при желании ускорить и увеличить отрыв от конкурентов

Главные факторы успеха реинжиниринга

Факторы успеха реинжиниринга

```
graph LR; A[Факторы успеха реинжиниринга] --> B[Стремительность преобразований]; A --> C[Настроенность персонала на быстрое изменение характера собственной работы (круг работ, ответственность, работа в команде)]; A --> D[Формирование у каждого работника единого для всех понимания предпочтительного будущего организации и своего личного вклада в его достижение]; A --> E[Создание среды и инфраструктуры для обучения, профессионального роста и развития творческих способностей персонала];
```

Стремительность преобразований

Настроенность персонала на быстрое изменение характера собственной работы (круг работ, ответственность, работа в команде)

Формирование у каждого работника единого для всех понимания предпочтительного будущего организации и своего личного вклада в его достижение

Создание среды и инфраструктуры для обучения, профессионального роста и развития творческих способностей персонала

Базовые категории реинжиниринга



Деловая процедура	- функция, задача, цепь событий, происходящих в течение определенного времени и обладающих распознаваемым результатом
Бизнес-процесс	- горизонтальная иерархия внутренних и зависимых между собой функциональных действий, конечной целью которых является выпуск продукции или отдельных ее компонентов
Бизнес-система	- связанное множество бизнес-процессов, конечной целью которого является выпуск продукции в виде товаров, услуг, документов

Типичные бизнес-процессы, перепроектируемые и совершенствуемые в ходе реинжинириговой деятельности

**Типичные для
реинжиниринга
бизнес-процессы в
компаниях**

```
graph TD; A[Типичные для реинжиниринга бизнес-процессы в компаниях] --> B[Выработка стратегии]; A --> C[Разработка нового товара]; A --> D[Выполнение заказов];
```

**Выработка
стратегии**

**Разработка
нового товара**

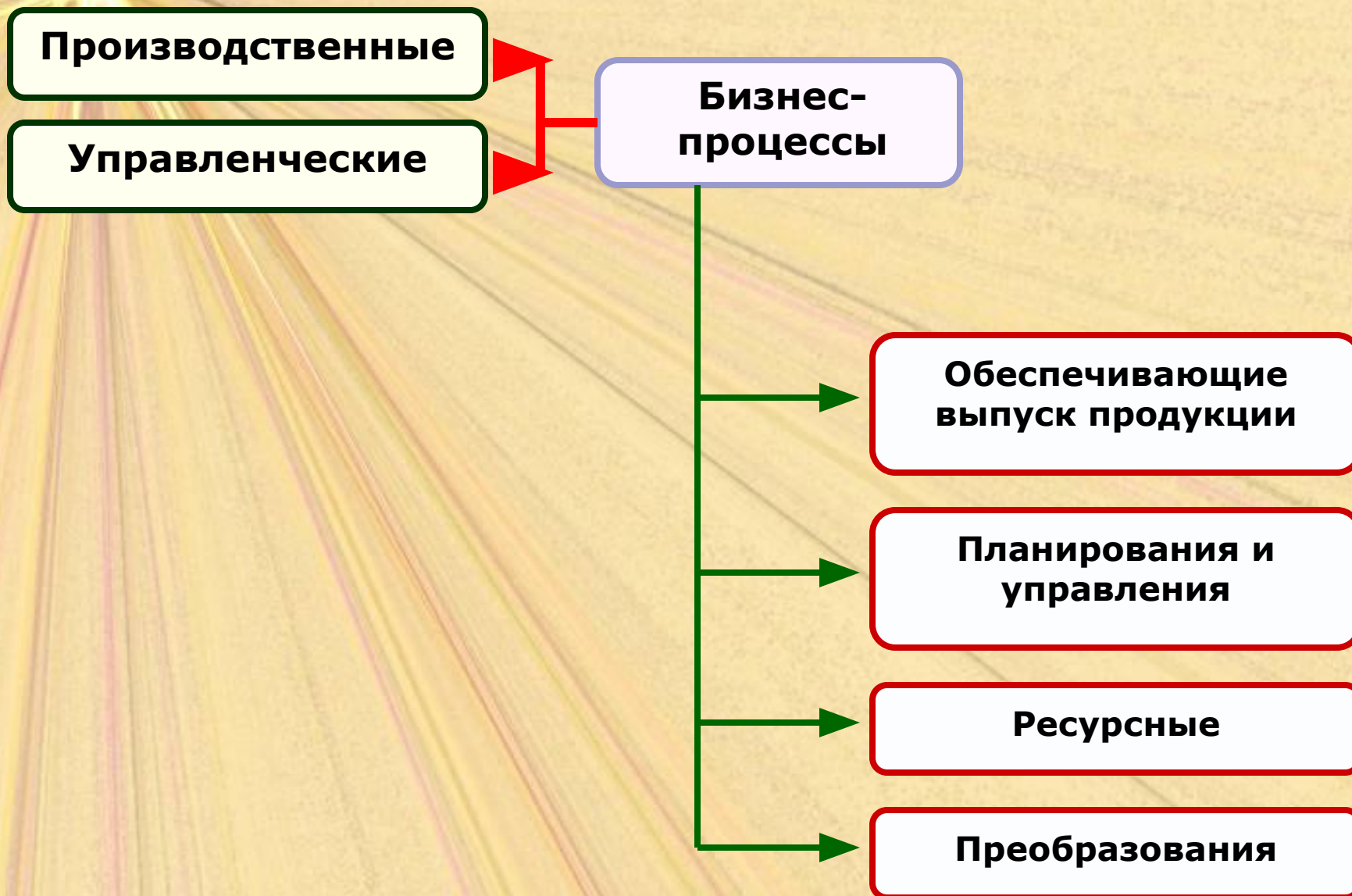
**Выполнение
заказов**

Типичные бизнес-процессы, перепроектируемые и совершенствуемые в ходе реинжиниринговой деятельности

Особенности бизнес-процессов, для которых проводится реинжиниринг:

- Диверсификация товаров и услуг (ориентация на различные сегменты рынка), вызывающая многообразие бизнес-процессов.
- Работа по индивидуальным заказам, требующая высокую степень адаптации базового бизнес-процесса к потребностям клиента.
- Внедрение новых технологий (инновационных проектов), затрагивающих все основные бизнес-процессы предприятия.
- Многообразие кооперативных связей с партнерами предприятия и поставщиками материалов, обуславливающих альтернативность построения бизнес-процесса.
- Нерациональность организационной структуры, запутанность документооборота, вызывающая дублирование операций бизнес-процесса.

Классификация бизнес-процессов



Показатели эффективности бизнес-процессов

**Показатели
эффективности
бизнес-процессов**

```
graph LR; A[Показатели эффективности бизнес-процессов] --> B[Количество продукции заданного качества, оплаченное за определенное время]; A --> C[Численность потребителей продукции]; A --> D[Количество типовых операций по производству продукции за определенное время]; A --> E[Издержки производства продукции]; A --> F[Длительность выполнения типовых операций]; A --> G[Капиталовложения в производство продукции];
```

Количество продукции заданного качества, оплаченное за определенное время

Численность потребителей продукции

Количество типовых операций по производству продукции за определенное время

Издержки производства продукции

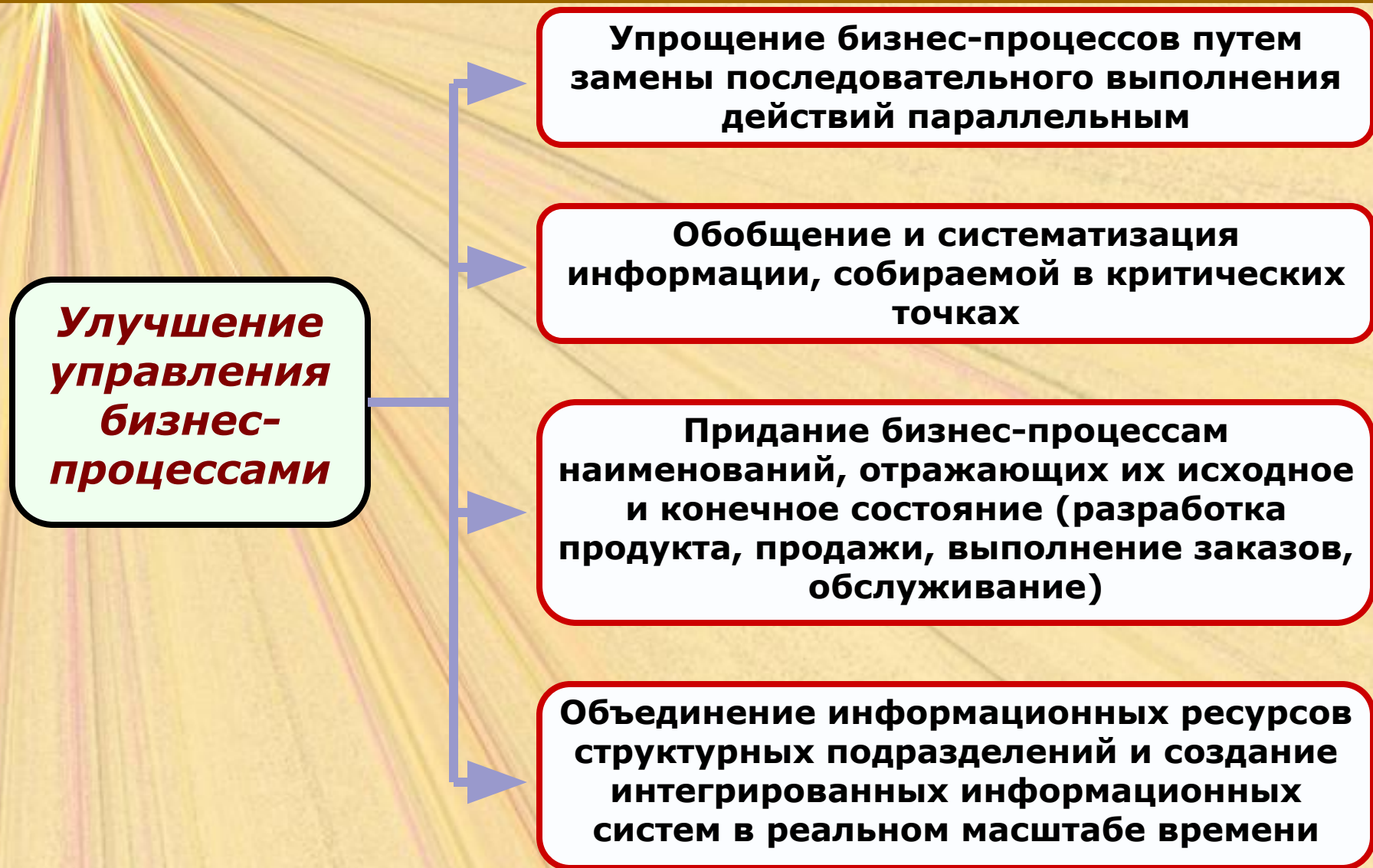
Длительность выполнения типовых операций

Капиталовложения в производство продукции

Участники реинжиниринговой деятельности и их функции

Участники	Функции
1. Лидер проекта – один из высших менеджеров фирмы	Возглавляет деятельность реинжиниринга, отвечает за идеологическое обоснование проекта, создает общий дух новаторства и ответственности
2. Управляющий комитет – члены высшего руководства, лидер проекта, менеджеры процессов	Осуществляет наблюдение, согласует цели и стратегии, интересы рабочих команд, разрешает конфликты
3. Менеджеры оперативного руководства	Разрабатывают методики и инструменты реинжиниринга, проводят обучение, координируют, помогают в формировании команд
4. Менеджеры процессов	Отвечают за обновление отдельных процессов, формируют команды, обеспечивают условия для их работы, осуществляют наблюдение и контроль
5. Рабочие команды – работники фирмы и внешние консультанты и разработчики	Осуществляют непосредственную работу по реинжинирингу

Пути улучшения управления бизнес-процессами



**Улучшение
управления
бизнес-
процессами**

Упрощение бизнес-процессов путем замены последовательного выполнения действий параллельным

Обобщение и систематизация информации, собираемой в критических точках

Придание бизнес-процессам наименований, отражающих их исходное и конечное состояние (разработка продукта, продажи, выполнение заказов, обслуживание)

Объединение информационных ресурсов структурных подразделений и создание интегрированных информационных систем в реальном масштабе времени

Основные этапы реинжиниринга



- 1. Формирование желаемого образа фирмы**
- 2. Создание модели существующего бизнеса (действия, работы, документация)**
- 3. Разработка модели нового бизнеса**
- 4. Перепроектирование выбранных хозяйственных процессов (рабочие процедуры, задания, технологии, способы)**
- 5. Формирование новых функций персонала (должн. инструкции, система мотивации, создание команд, программа подготовки и переподготовки)**
- 6. Создание информационных систем (оборудование, программное обеспечение, информационная система)**
- 7. Тестирование новой модели**
- 8. Внедрение новой модели**

Базовые принципы реинжиниринга



1. Несколько рабочих процедур объединяются в одну (горизонтальное сжатие процесса)

2. Исполнители принимают самостоятельные решения (вертикальное сжатие процесса)

3. Шаги процесса выполняются в естественном порядке (распараллеливание)

4. Процессы имеют различные варианты исполнения в зависимости от ситуации

5. Работа выполняется там (в том подразделении), где целесообразно

6. Уменьшается количество проверок и управленческих воздействий

7. Минимизируется количество согласований

8. Уполномоченный менеджер обеспечивает единую точку контакта (является буфером между сложным процессом и заказчиком)

9. Преобладает смешанный централизованно-децентрализованный подход (автономные подразделения используют централизованные данные)

Методы моделирования бизнес-процессов



Основные инструменты реинжиниринга

**Инструменты
реинжиниринга**

```
graph TD; A[Инструменты реинжиниринга] --> B[Средства интерактивной графики]; A --> C[Имитационное моделирование процессов в реальном времени]; A --> D[Моделирование бизнес-процессов с помощью диаграмм];
```

**Средства
интерактивной
графики**

**Имитационное
моделирование
процессов в
реальном времени**

**Моделирование
бизнес-процессов с
помощью диаграмм**

Основные инструменты реинжиниринга

Бизнес-процессы
упрощены
е. отражены
давать
для управления

Зачем: стратегическая модель, в которой

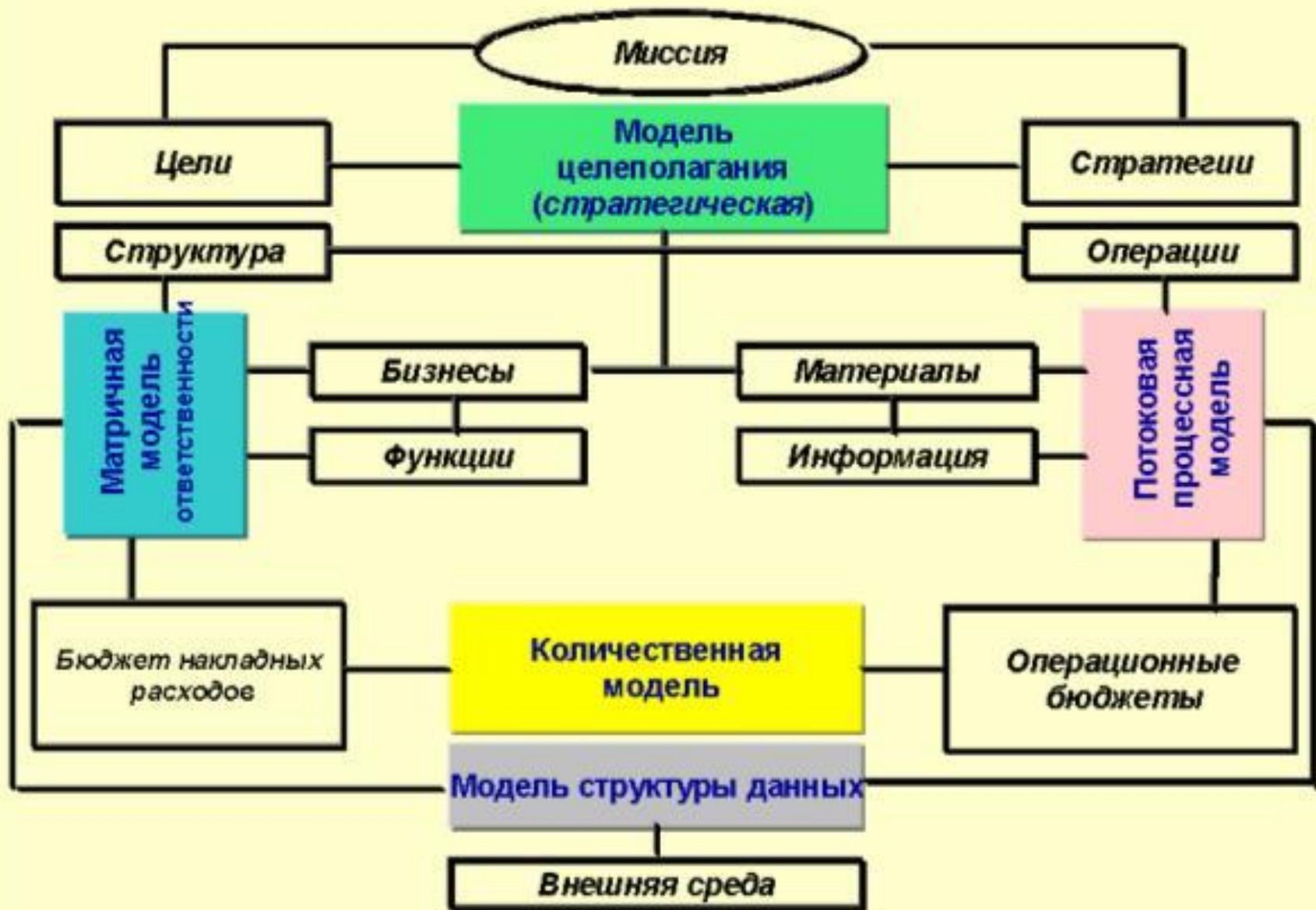
Что – Где – Кто: организационно-функциональная

определенная
В каком виде: модель, определяющая структуру разделения регистров, отчетности, представляемая информация

Сколько: финансовая модель, которая позволяет перейти к количественной оценке ресурсов, потребляемых предприятием в ходе своей деятельности

Зачем								
Зачем	Что							
Зачем	Что							
Зачем	Что	Где						
Зачем	Что	Где						
Зачем	Что	Где						
Зачем	Что	Где	Кто	Как	Когда	Кому		
Зачем	Что	Где	Кто	Как	Когда	Кому	В каком виде	
Зачем	Что	Где	Кто	Как	Когда	Кому	В каком виде	Сколько

Полная бизнес-модель компании



Последствия реинжиниринга

- **Переход от функциональных подразделений к командам процессов.**
- **Работа исполнителя изменяется от простой к многоплановой.**
- **Требования к работникам изменяются: от контролируемого исполнителя предписанных заданий к принятию самостоятельных решений.**
- **Изменяются требования к подготовке сотрудников: от курсов обучения к образованию.**
- **Изменяется оценка эффективности работы и оплаты труда: от оценки деятельности к оценке результата.**
- **Критерий продвижения в должности изменяется: от эффективности выполнения работы к способности выполнять работу.**
- **Изменяется цель исполнителя: от удовлетворения потребностей начальника к удовлетворению потребностей клиентов.**
- **Функции менеджеров изменяются от контролирующих к тренерским.**
- **Организационная структура меняется от иерархической к более «плоской».**
- **Административные функции изменяются от секретарских к лидирующим.**

10-11.4. Методология структурного моделирования и проектирования SADT

Методология SADT (Structured Analysis and Design Technique - технология структурного анализа и проектирования) разработана Дугласом Т. Россом в 1969-1973 годах. Технология изначально создавалась для проектирования систем более общего назначения по сравнению с другими структурными методами, выросшими из проектирования программного обеспечения. SADT - одна из самых известных и широко используемых методик проектирования.

Методология SADT представляет структурный подход к моделированию систем. Структурный подход основан на следующих принципах:

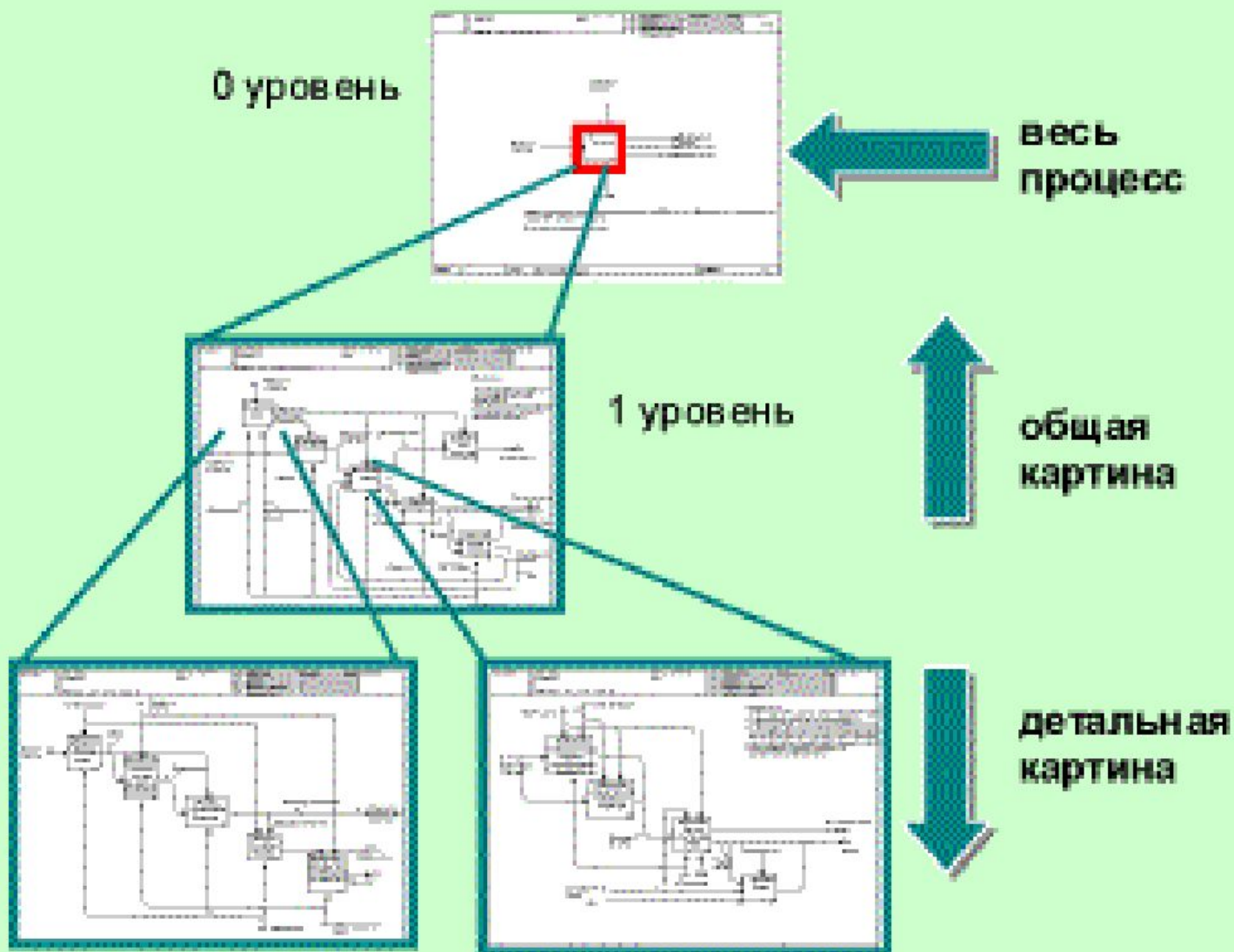
- в процессе моделирования система разделяется (декомпозируется) на составляющие ее функциональные подсистемы;
- декомпозиция проводится до нужной степени детализации, пока содержание каждой составляющей не станет совершенно понятно;
- подсистемы, составляющие модель, иерархически упорядочиваются.

Таким образом, базовыми принципами структурного анализа являются:

- принцип «разделяй и властвуй»;
- принцип иерархического упорядочивания.

Методология SADT успешно используется для моделирования широкого круга систем, как для новых систем, которые только планируется создать, так и для уже существующих. В первом случае SADT используется, чтобы определить требования к будущей системе и описать ее функции, чтобы потом можно было разработать систему, которая удовлетворяет этим требованиям и реализует эти функции. Во втором случае, для уже существующих систем, SADT используется для проведения анализа функций, выполняемых системой, и описания механизмов, посредством которых они осуществляются.

Моделирование бизнес-процессов



Методология SADT может быть направлена как для описания функций, выполняемых системой, так и на описание объектов, составляющих систему, их свойств и связей между ними. В первом случае методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели системы, т.е. отображает производимые системой действия и связи между этими действиями. Во втором случае методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения модели данных.

SADT реализуется в следующих нотациях:

Метод IDEF0 - функциональные модели и соответствующие диаграммы. SADT-модель, представляющая систему в виде иерархии взаимосвязанных функций, которые выполняет система, называется функциональной моделью. Функциональная модель показывает, какие функции выполняет исследуемая система, как эти функции связаны между собой и как они упорядочены по степени важности или по порядку исполнения. Каждая функция, представленная в модели, может быть детализирована с любой степенью подробности, то есть разложена на составляющие ее функции, каждая из которых также может быть разложена на составляющие и т.к., пока не будет достигнута необходимая степень точности ответа на вопросы, поставленные относительно системы.

Функциональная модель строится с помощью графического языка диаграмм. Каждая функция в модели может быть детально описана в виде отдельной диаграммы.

Как разновидность SADT-моделирования функциональное моделирование обозначилось под названием стандарт IDEF0.

Метод DFD (Data Flow Diagrams) - диаграммы потоков данных. Моделирует движение информации в системе. Может использоваться для описания документооборота.

Метод IDEF1X или ERD (Entity-Relationship Diagrams) - диаграммы "сущность-связь". SADT-модель, которая ориентирована на объекты входящие в исследуемую систему, их свойства и связи между ними, называется моделью данных. Обычно, это не что иное, как реляционная модель данных исследуемой системы, которая состоит из сущностей, описываемых наборов атрибутов, и связей между ними. Типы связей определяют характер сущностей. Модель данных может быть положена в основу информационной модели исследуемой системы, создаваемой с помощью различных реляционных СУБД.

Метод IDEF3 – диаграммы процессов. Графически описывает процессы, протекающие в системе.

Процесс моделирования в SADT включает сбор информации об исследуемой области, документирование полученной информации и представление ее в виде модели и уточнение модели. Кроме того, этот процесс подсказывает вполне определенный путь выполнения согласованной и достоверной структурной декомпозиции, что является ключевым моментом в квалифицированном анализе системы. SADT уникальна в своей способности обеспечить как графический язык, так и процесс создания непротиворечивой и полезной системы описаний.

На современном рынке средств разработки ИС достаточно много систем, в той или иной степени удовлетворяющих перечисленным требованиям. CASE-средство BPwin поддерживает методологию SADT. В состав этой методологии входят:

- технология функционального моделирования IDEF0;
- технология динамического моделирования IDEF3 (WorkFlow Diagram);
- технология моделирования потоков данных DFD (DataFlow Diagram).

Функциональная модель предназначена для описания существующих бизнес-процессов на предприятии (так называемая модель AS-IS) и идеального положения вещей – того, к чему нужно стремиться (модель TO-BE). Следовательно, сначала нужно определиться с понятием бизнес-процесса.


Основная цель бизнес-процесса – преобразование входящего массива данных (информации, документов) и ресурсов (материальные, финансовые, людские), необходимых для реализации процесса, в результат (продукцию) процесса. Основными составляющими элементами бизнес-процесса является совокупность подпроцессов, работ, операций, осуществляемых над входами для получения выходов.

Существуют **первичные и вторичные входы и выходы** процесса. Первичные входы поступают на начало процесса. Вторичные входы появляются в ходе реализации процесса на составляющих его подпроцессах. Первичный выход – это прямой, запланированный результат реализации процесса. Вторичный выход – это побочный продукт процесса, не являющийся его главной целью

Процесс осуществляется с помощью определенного механизма и производится для того, кто потребляет результат процесса, т.е., является клиентом процесса.

Таким образом, бизнес-процесс обладает следующими основными **характеристиками:**

- Входящий массив данных (информация, документы и т.п.) и ресурсов (материальные и нематериальные активы);
- Результат бизнес-процесса;
- «Владелец» бизнес-процесса: объект (компания, подразделение, сотрудник), отвечающий за данный бизнес-процесс;
- Механизм реализации.



Методология IDEF0 предписывает построение иерархической системы диаграмм – единичных описаний фрагментов системы. Сначала проводится описание системы в целом и ее взаимодействия с окружающим миром (контекстная диаграмма), после чего проводится функциональная декомпозиция - система разбивается на подсистемы и каждая подсистема описывается отдельно (диаграммы декомпозиции). Затем каждая подсистема разбивается на более мелкие и так далее до достижения нужной степени подробности. После каждого сеанса декомпозиции проводится сеанс экспертизы: каждая диаграмма проверяется экспертами предметной области, представителями заказчика, людьми, непосредственно участвующими в бизнес-процессе. Такая технология создания модели позволяет построить модель, адекватную предметной области на всех уровнях абстрагирования.

Если в процессе моделирования нужно осветить специфические стороны технологии предприятия, VPwin позволяет переключиться на любой ветви модели на нотацию IDEF3 или DFD и создать смешанную модель. Нотация DFD включает такие понятия, как внешняя ссылка и хранилище данных, что делает ее более удобной (по сравнению с IDEF0) для моделирования документооборота. Методология IDEF3 включает элемент «перекресток», что позволяет описать логику взаимодействия компонентов системы.

Функциональное моделирование является основой методологии SADT и предполагает построение древовидной функциональной структуры рассматриваемого процесса.


10-11.5. Метод функционального моделирования IDEF0

В технологии функционального моделирования IDEF0 реализован процессный подход, т.е., система представляется как иерархическая совокупность взаимодействующих процессов, подпроцессов, функций и операций. Такая чисто функциональная ориентация является принципиальной – функции системы анализируются независимо от объектов, которыми они оперируют. Это позволяет более четко смоделировать логику взаимодействия бизнес-процессов организации.

В IDEF0 система представляется как совокупность взаимодействующих работ (или функций). Связи между работами определяют технологический процесс или структуру взаимосвязи внутри организации. Модель SADT представляет собой серию диаграмм, разбивающих сложный объект на составные части.

В соответствии с принципами процессного подхода моделирование какой-либо системы в IDEF0 начинается с определения контекста, т.е. наиболее абстрактного уровня описания системы в целом. В контекст входит определение объекта моделирования, цели и точки зрения на модель.

Под объектом понимается сама система, при этом необходимо точно установить, что входит в систему, а что лежит за ее пределами, другими словами, нужно определить, что будет в дальнейшем рассматриваться как компонент системы, а что – как внешнее воздействие.



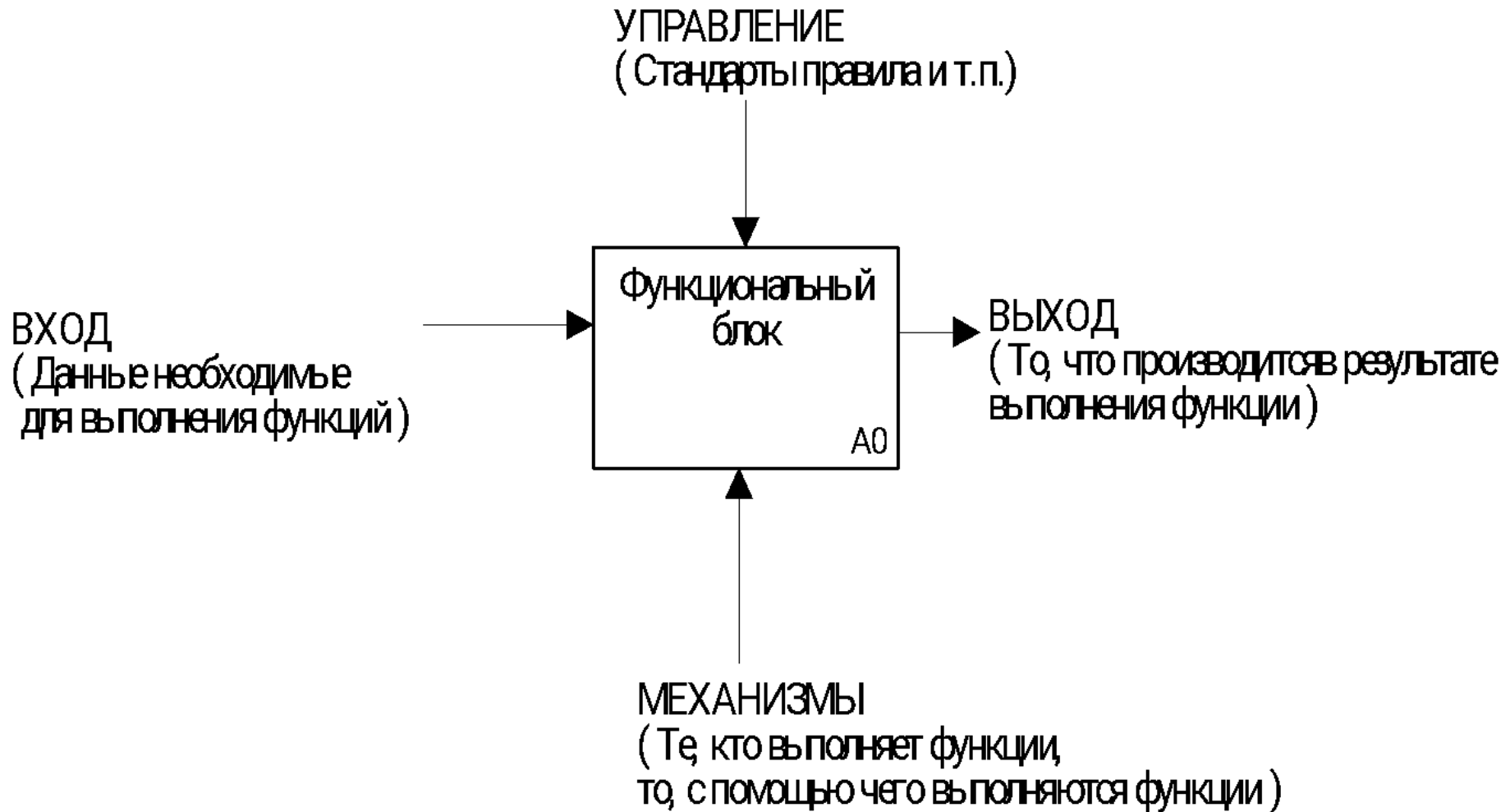
На определение объекта системы будет существенно влиять позиция, с которой рассматривается система, и цель моделирования – вопросы, на которые построенная модель должна дать ответ. Другими словами, первоначально необходимо определить область моделирования. Описание области моделирования как системы в целом, так и ее компонентов является основой построения модели. При формулировании области необходимо учитывать два компонента – широту и глубину. Широта подразумевает определение границ модели. Глубина определяет, на каком уровне детализации модель является завершённой. При определении глубины системы необходимо не забывать об ограничениях времени – трудоемкость построения модели растет в геометрической прогрессии от глубины декомпозиции. После определения границ модели предполагается, что новые объекты не должны вноситься в моделируемую систему; поскольку все объекты модели взаимосвязаны, внесение нового объекта может изменить существующие взаимосвязи. Внесение таких изменений в готовую модель является очень трудоемким процессом.


В основу IDEF0 положен формализованный язык, характеризующийся точными правилами построения функциональной модели. Алфавит этого языка включает совокупность графических символов, из которых строятся функциональные схемы (выражения) в соответствии с правилами построения схем. Для аналитического представления функциональной модели, а также для описания свойств и семантики используются естественные и логико-математические языки.

Основой метода являются следующие понятия:

1. В терминах IDEF0 система представляется в виде комбинации блоков и интерфейсных дуг. Блоки используются для представления функций системы, функции должны иметь имена, выраженные грамматической формой глагола.
2. Интерфейсная дуга - элемент, описывающий данные, управление, механизмы, оказывающие влияние на функцию, изображенную блоком.. В зависимости от того, к какой стороне блока направлена дуга, она, соответственно, носит название "входная", "выходная", "управляющая". Изобразительным элементом, представляющим дугу, является стрелка. Место соединения дуги с блоком определяет тип интерфейса.
3. Управляющие выполнением функции данные входят в блок сверху, информация, которая подвергается воздействию функции, показана с левой стороны блока; результаты выхода показаны с правой стороны. Механизм (человек или информационная система), который выполняет функцию, представляется дугой, входящей в блок снизу.
Стрелки или дуги играют роль интерфейса и означают либо предметы (материальные объекты), либо информационные объекты – данные.

Функциональный блок (Activity Box)





Блоки представляют функции системы, дуги представляют множество объектов (физические объекты, информация или действия, которые образуют связи между функциональными блоками). Взаимодействие работ с внешним миром и между собой описывается в виде стрелок или дуг. С дугами связываются метки на естественном языке, описывающие данные, которые они представляют. Дуги показывают, как функции системы связаны между собой, как они обмениваются данными и осуществляют управление друг другом. Дуги могут разветвляться и соединяться. Ветвление означает множественность (идентичные копии одного объекта) или расщепление (различные части одного объекта). Соединение означает объединение или слияние объектов. Пять типов стрелок допускаются в диаграммах:

Вход (Input) - материал или информация, которые используются работой для получения результата (стрелка, входящая в левую грань).

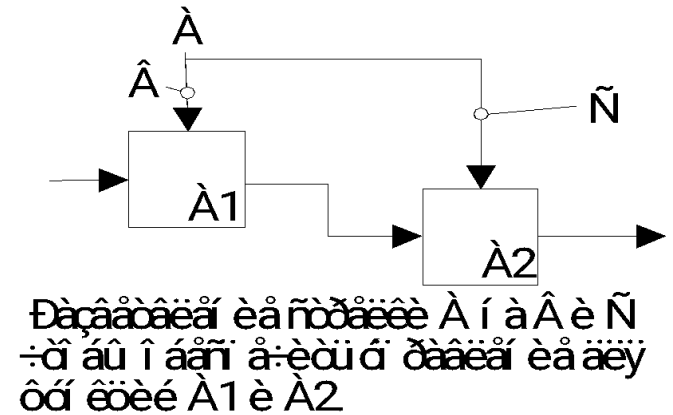
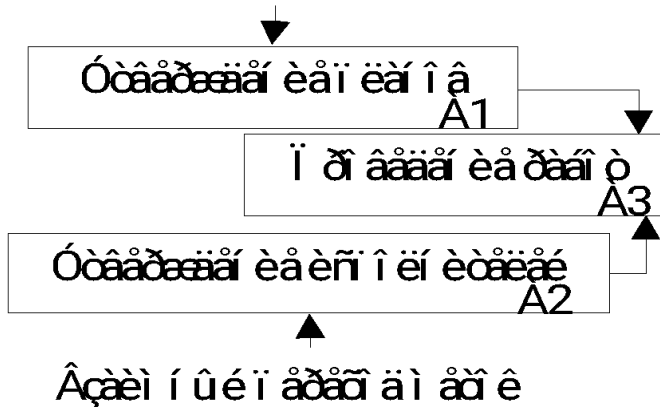
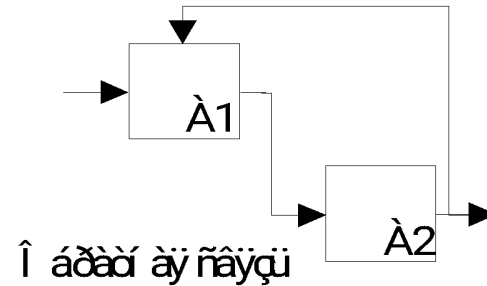
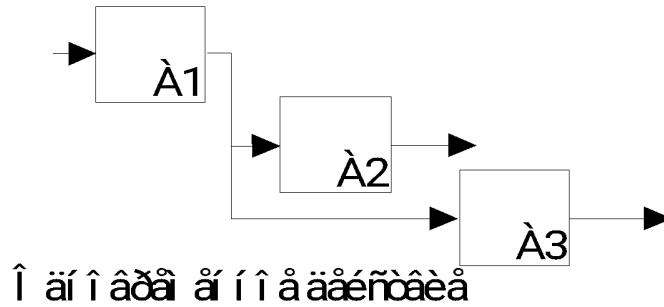
Управление (Control) - правила, стратегии, стандарты, которыми руководствуется работа (стрелка, входящая в верхнюю грань). В отличие от входной информации управление не подлежит изменению.

Выход (Output) - материал или информация, которые производятся работой (стрелка, исходящая из правой грани). Каждая работа должна иметь хотя бы одну стрелку выхода, т.к. работа без результата не имеет смысла и не должна моделироваться.

Механизм (Mechanism) - ресурсы, которые выполняют работу (персонал, станки, устройства - стрелка, входящая в нижнюю грань).

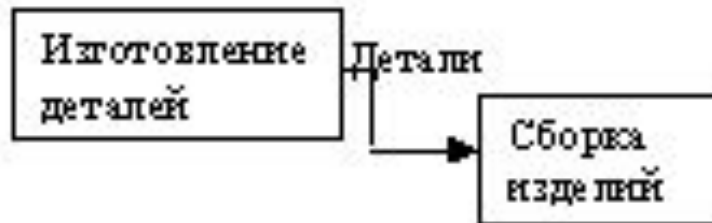
Стрелки помечаются уникальными метками, выраженными грамматической формой существительного и называются ICOM-метками (Input, Control, Output, Mechanism) .

Основные правила соединения блоков (синтаксис)

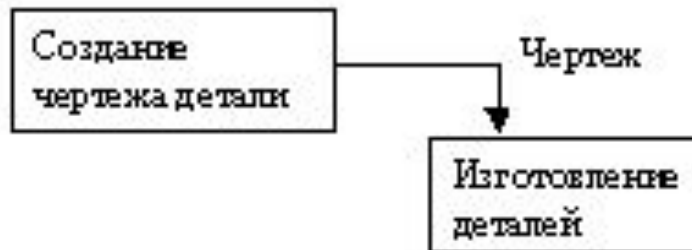


Различают в IDEF0 пять типов связей работ:

Связь по входу (input-output), когда выход вышестоящей работы направляется на вход следующей работы



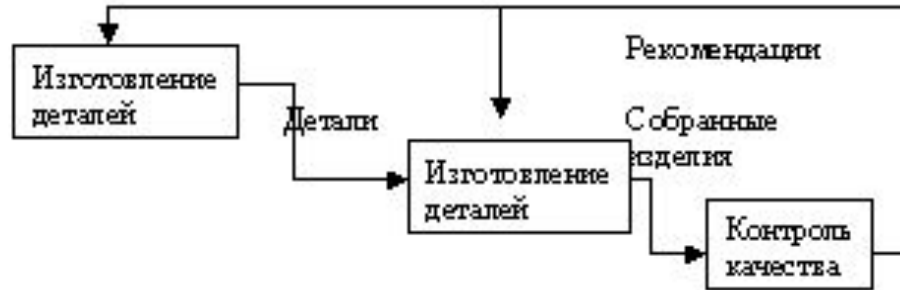
Связь по управлению (output-control), когда выход вышестоящей работы направляется на управление следующей работы. Связь показывает доминирование вышестоящей работы



Обратная связь по входу (output-input feedback), когда выход нижестоящей работы направляется на вход вышестоящей. Используется для описания циклов.



Обратная связь по управлению (output-control feedback), когда выход нижестоящей работы направляется на управление вышестоящей. Является показателем эффективности бизнес-процесса.



Связь выход-механизм (output-mechanism), когда выход одной работы направляется на механизм другой и показывает, что работа подготавливает ресурсы для проведения другой работы.



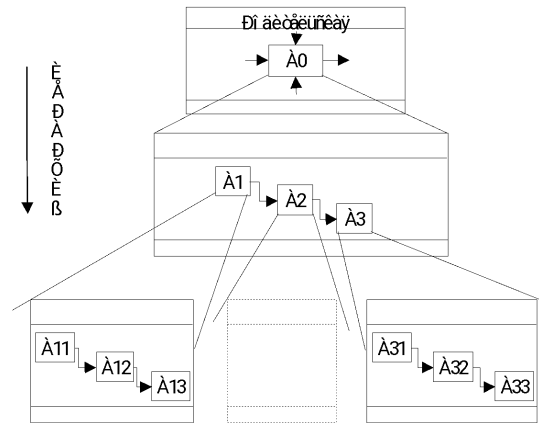
Схему блоков, соединенных по приведенным выше правилам, называют диаграммой соответствующего уровня иерархии.

Модель IDEF0 представляет иерархически образованный набор диаграмм с поддерживающей их семантической информацией, представленной в виде глоссария и в виде гипертекста.

Иерархия диаграмм определяется схемой декомпозиции блоков. Под иерархией понимаются уровни раскрытия блоков функциональной модели в процессе их детализации. Каждый блок может быть декомпозирован на другой диаграмме (рис. 22).

Идентификация диаграмм осуществляется:

- по имени диаграммы;
- по иерархическому (позиционному) коду блоков A_{ijk} , где A_{ijk} является потомком родительского блока A_{ij} , а A_{ij} - потомком блока A_i , $i, j, k \in \mathbb{N}$.



Отметим, что при декомпозиции функциональных блоков набор их формальных свойств (исключая номера ICOM-меток) неизменен относительно уровней.

Функциональная модель, построенная т.о. является адекватным описанием уже существующих процессов (AS-IS). Проанализировав эту модель в нее можно внести изменения для реинжиниринга бизнес-процессов, т.о. будет получена функциональная модель процесса (TO-BE).

Созданный по функциональной модели глоссарий является основой для построения информационной модели. Глоссарий сохраняет преемственность между именами функций функциональной модели и отношениями информационной модели, интерфейсными дугами функциональной модели и сущностями информационной модели.

10-11.6 Метод процессного моделирования IDEF3

Создание процессной модели является очень сложной задачей, но позволяет детально исследовать технологический процесс, т.к. детализирует функциональную модель до отдельных работ.

Методология процессного моделирования IDEF3 позволяет описать логику взаимодействия информационных потоков, взаимоотношения между процессами обработки информации и объектами, являющимися частью этих процессов. Методология IDEF3, называемая также *workflow diagramming* используется в моделировании деловых процессов для анализа завершенности процедур обработки информации. С их помощью описываются сценарии ведения процедур с учетом причинно-следственных связей между объектами системы.

IDEF3 дополняет IDEF0 и строит модели, которые в дальнейшем могут быть использованы для имитационного анализа такими инструментами имитационного моделирования как Arena (фирма System Modeling Corporation).

Единицами описания IDEF3 модели являются диаграммы и единицы работы (Unit of work (UOW)), также называемые работами (activity) и связи между ними.

Диаграмма является основной единицей описания, имеет имя и состоит из работ. Она может быть построена при декомпозиции функции IDEF0 модели или построена отдельно как IDEF3 диаграмма.



Диаграмма IDEF3

Правила определения работ:

1. Работы изображаются прямоугольниками с прямыми углами и имеют имя, выраженное отглагольным существительными, обозначающим процесс действия, одиночным или в составе фразы, и номер (идентификатор); другое имя существительное в составе той же фразы обычно отображает основной выход (результат) работы (например, принятие решения).

2. Имя работы может меняться в процессе моделирования, но ее идентификатор остается неизменным, даже если работа будет удалена, ее идентификатор не будет вновь использоваться для других работ.

3. Каждая работа в IDEF3 должна иметь ассоциированный документ, который включает текстовое описание компонентов работы: объектов (Objects), и фактов (Facts), связанных с работой, ограничений (Constraints), накладываемых на работу, и дополнительное описание работы (Description).

Связи показывают взаимоотношение работ, обозначаются стрелками и могут быть трех типов:

1. Старшая (Precedence)- сплошная линия, связывающая единицы работ (UOW). Показывает, что работа- источник должна заканчиваться прежде, чем начнется работа- цель.

2. Связь отношения (Relational Link) –пунктирная линия, используемая для изображения связей между единицами работ(UOW), а также между единицами работ и объектами ссылок.

3. Потоки объектов (Object Flow)- стрелка с двумя наконечниками, применяется для описания того факта, что объект используется в двух или более единицах работы, например, когда объект порождается в одной работе, а используется в другой.

Окончание одной работы может служить сигналом к началу нескольких работ, или же одна работа для своего запуска может ожидать окончания нескольких работ. Тогда для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые должны произойти используются перекрестки. Все перекрестки на диаграмме нумеруются, каждый номер имеет префикс J. Стрелки могут сливаться и разветвляться только через перекрестки. Различают перекрестки для слияния (Fan- in Junction) и разветвления (Fan- out Junction) стрелок. Перекрестки бывают 5 типов.

Типы перекрестков:

1. Асинхронное «И» (Asynchronous AND). Все предшествующие процессы должны быть завершены, или все следующие запущены.
2. Синхронное «И» (Synchronous AND). Все предшествующие процессы должны быть завершены одновременно, или все следующие одновременно запущены.
3. Асинхронное «ИЛИ» (Asynchronous OR). Один или несколько предшествующих процессов должны быть завершены, или последующих запущены.
4. Синхронное «ИЛИ» (Synchronous OR). Один или несколько предшествующих процессов должны быть завершены одновременно, или последующих одновременно запущены.
5. Исключающее «ИЛИ» (XOR или Exclusive OR). Только один предшествующий процесс завершен или только один следующий запускается.

Правила создания перекрестков:

1. Каждому перекрестку для слияния должен предшествовать перекресток для разветвления
2. Перекресток для слияния «И» не может следовать за перекрестком для разветвления «ИЛИ».
3. Перекресток для слияния типа исключающего «ИЛИ» не может следовать за перекрестком для разветвления типа «И».
4. Перекресток, имеющий одну стрелку на одной стороне, должен иметь более одной стрелки на другой.

В IDEF3 декомпозиция используется для детализации работ и может быть применена многократно, т.е. работа может иметь множество дочерних работ. Это позволяет в одной модели описать альтернативные потоки. Декомпозиция может быть сценарием или описанием. Описание включает все возможные пути процесса. Сценарий является частным случаем описания и иллюстрирует только один путь развития процесса.


Номер каждой работы состоит из номера родительской работы, номера декомпозиции и собственного номера работы на текущей диаграмме.


Основой модели IDEF3 служит так называемый сценарий процесса, который выделяет последовательность действий и подпроцессов анализируемой системы.


Как и в методе IDEF0, основной единицей модели IDEF3 является диаграмма. Другой важный компонент модели — действие, или в терминах IDEF3 "единица работы" (Unit of Work). Диаграммы IDEF3 отображают действие в виде прямоугольника. Действия именуются с использованием глаголов или отглагольных существительных, каждому из действий присваивается уникальный идентификационный номер. Этот номер не используется вновь даже в том случае, если в процессе построения модели действие удаляется. В диаграммах IDEF3 номер действия обычно предваряется номером его родителя.




Существенные взаимоотношения между действиями изображаются с помощью связей. Все связи в IDEF3 являются однонаправленными, и хотя стрелка может начинаться или заканчиваться на любой стороне блока, обозначающего действие, диаграммы IDEF3 обычно организуются слева направо таким образом, что стрелки начинаются на правой и заканчиваются на левой стороне блока.

Связь типа **"временное предшествование"** ; что исходное действие должно полностью завершиться, прежде чем начнется выполнение конечного действия.

Связь типа **"объектный поток"**  используется в том случае, когда некоторый объект, являющийся результатом выполнения исходного действия, необходим для выполнения конечного действия. Наименования потоковых связей должны четко идентифицировать объект, который передается с их помощью. Исходное действие должно завершиться, прежде чем конечное действие может начать выполняться.

Связь типа **"нечеткое отношение"**  используется для выделения отношений между действиями, которые не являются отношениями предшествования или объектных связей. Одно из применений нечетких отношений — отображение взаимоотношений между параллельно выполняющимися действиями.




Завершение одного действия может инициировать начало выполнения сразу нескольких других действий или, наоборот, определенное действие может требовать завершения нескольких других действий до начала своего выполнения. Такие ситуации описываются с помощью **перекрестков (junction)**. Перекрестки (или соединения) используются для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы. Перекрестки разбивают или соединяют внутренние потоки и используются для изображения ветвления процесса:

разворачивающие соединения используются для разбиения потока. Завершение одного действия вызывает начало выполнения нескольких других;

сворачивающие соединения объединяют потоки. Завершение одного или нескольких действий вызывает начало выполнения другого действия.

Соединения **«И»** инициируют выполнение конечных действий. Все действия, присоединенные к сворачивающему соединению "и", должны завершиться, прежде чем начнется выполнение следующего действия. Например, после обнаружения пожара инициируются включение пожарной сигнализации, вызов пожарной охраны, и начинается тушение пожара. Запись в журнал производится только тогда, когда все три перечисленных действия завершены.






Соединение **"исключающее или"** означает, что вне зависимости от количества действий, связанных со сворачивающим или разворачивающим соединением, инициировано будет только одно из них, и поэтому только оно будет завершено перед тем, как любое действие, следующее за сворачивающим соединением, сможет начаться. Если правила активации соединения известны, они обязательно должны быть документированы либо в его описании, либо пометкой стрелок, исходящих из разворачивающего соединения. На рис. соединение "исключающее или" используется для отображения того факта, что студент не может одновременно быть направлен на лекции по двум разным курсам.

Соединение **"ИЛИ"** предназначено для описания ситуаций, которые не могут быть описаны двумя предыдущими типами соединений. Аналогично связи нечеткого отношения соединение "или" в основном определяется и описывается непосредственно аналитиком. На рис. соединение J2 может активизировать проверку данных чека и/или проверку суммы наличных. Проверка чека инициируется, если покупатель желает расплатиться чеком, проверка суммы наличных — при оплате наличными. И то, и другое действие инициируются при частичной оплате, как чеком, так и наличными.

В рассмотренных примерах все действия выполнялись асинхронно, т.е. они не инициировались одновременно. Однако существуют случаи, когда время начала или окончания параллельно выполняемых действий должно быть одинаковым, т.е. действия должны выполняться синхронно. Для моделирования такого поведения системы используются различные виды синхронных соединений, которые обозначаются двумя двойными вертикальными линиями внутри прямоугольника.

5 типов перекрестков

Наименование	Сворачивающий перекресток	Разворачивающий перекресток
Асинхронный «И»	Все предшествующие процессы должны быть завершены	Все следующие процессы должны быть запущены
Синхронный «И»	Все предшествующие процессы завершены одновременно	Все следующие процессы запускаются одновременно
Асинхронный «ИЛИ»	Один или несколько предшествующих процессов должны быть завершены	Один или нескольких следующих процессов должны быть запущены
Синхронный «ИЛИ»	Один или несколько предшествующих процессов завершены одновременно	Один или нескольких следующих процессов запускаются одновременно
Исключающее «ИЛИ»	Только один предшествующий процесс завершен	Только один следующий процесс запускается



Все соединения на диаграммах должны быть парными, из чего следует, что любое разворачивающее соединение имеет парное себе сворачивающее. Однако типы соединений не обязательно должны совпадать.

Соединения могут комбинироваться для создания более сложных ветвлений. Комбинации соединений следует использовать с осторожностью, поскольку перегруженные ветвлением диаграммы могут оказаться сложными для восприятия.

Действия в IDEF3 могут быть декомпозированы или разложены на составляющие для более детального анализа. Метод IDEF3 позволяет декомпозировать действие несколько раз, что обеспечивает документирование альтернативных потоков процесса в одной модели.

Еще одним элементом диаграммы IDEF3 является **указатель**. Указатель выражает некую идею, концепцию или данные, которые нельзя связать со стрелкой, перекрестком или работой. Указатели должны быть связаны с единицами работ или перекрестками пунктирными линиями. Типы и назначение указателей представлены в таблице.

Типы указателей

Тип указателя	Назначение
Объект (OBJECT)	Описывает участие в работе важного объекта, не рассматриваемого в качестве основного объекта в модели в целом.
Ссылка (GOTO)	Инструмент циклического перехода (в повторяющейся последовательности работ), возможно на текущей диаграмме, но не обязательно. Если все работы цикла присутствуют на текущей диаграмме, цикл может также изображаться стрелкой, возвращающейся на стартовую работу. Может ссылаться на перекресток.
Единица действия (UOB-Unit of behavior)	Применяется, когда необходимо подчеркнуть множественное использование работы, но без зацикливания. Обычно этот тип ссылки не используется для моделирования автоматически запускающихся работ. Например, если действие «Подсчет наличных» запускается несколько раз, в первый раз оно создается как действие, а последующие его появления на диаграмме оформляются указателями UOB.
Заметка (NOTE)	Используется для документирования важной информации, относящейся к графическим объектам на диаграмме.
Уточнение (ELAB - Elaboration)	Используется для усовершенствования графиков или их более детального описания. Обычно употребляется для детального описания разветвления и слияния стрелок на перекрестках.

10-11.7. Моделирование потоков данных

Диаграммы потоков данных (Data Flow Diagrams — DFD) предназначены для демонстрации того, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами.

Диаграммы потоков данных используются для описания движения документов и обработки информации как дополнение к IDEF0. В отличие от IDEF0, где система рассматривается как взаимосвязанные работы и стрелки представляют собой жесткие взаимосвязи, стрелки в DFD показывают лишь то, как объекты (включая данные) движутся от одной работы к другой. DFD отражает функциональные зависимости значений, вычисляемых в системе, включая входные значения, выходные значения и внутренние хранилища данных. DFD - это граф, на котором показано движение значений данных от их источников через преобразующие их процессы к их потребителям в других объектах.

Основными компонентами диаграмм потоков данных являются:

- внешние сущности;
- функциональные блоки;
- потоки данных;
- хранилища данных.

Внешняя сущность представляет собой материальный объект или физическое лицо, являющиеся 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 системы.

Графическое представление внешней сущности



Функциональный блок моделирует некоторую функцию или процесс, который преобразует входные потоки данных в выходные в соответствии с определенным алгоритмом. Физически процесс может быть реализован различными способами: это может быть подразделение организации, выполняющее обработку входных документов, или программа, аппаратно реализованное логическое устройство и т.д.



Графическое изображение процесса в виде функционального блока.

Функциональные блоки нотации DFD имеют входы и выходы, но не имеют управления и механизма исполнения. В некоторых интерпретациях нотации DFD механизмы исполнения моделируются как ресурсы и изображаются в нижней части функционального блока.

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. Каждый поток данных имеет имя, отражающее его содержание. В DFD используются также двунаправленные стрелки, которые нужны для отображения взаимодействия между блоками по типу «запрос – ответ».



Хранилище данных — это абстрактное устройство для хранения информации, которую можно в любой момент поместить в хранилище и через некоторое время извлечь, причем способы помещения и извлечения могут быть любыми (рис. 31). В отличие от потоков данных, описывающих объекты в движении, хранилища данных изображают объекты в покое. В материальных системах хранилища данных изображаются там, где объекты ожидают обработки, например, в очереди. В системах обработки информации хранилища данных являются механизмом, который позволяет сохранить данные для последующих процессов. Хранилище данных может быть реализовано физически в виде микрофиши, ящика в картотеке, таблицы в оперативной памяти, файла на магнитном носителе и т.д. Хранилище данных в общем случае является прообразом будущей базы данных, и описание хранящихся в нем данных должно соответствовать модели данных.

Графическое изображение хранилища данных



10-11.8. Построение DFD-диаграмм

Первым шагом при построении иерархии DFD является построение контекстных диаграмм. Обычно при проектировании относительно простых систем 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. Перед построением контекстной DFD необходимо проанализировать внешние события (внешние сущности), оказывающие влияние на функционирование системы. Количество потоков на контекстной диаграмме должно быть по возможности небольшим, поскольку каждый из них может быть в дальнейшем разбит на несколько потоков на следующих уровнях диаграммы.

Для проверки контекстной диаграммы можно составить список событий. Список событий должен состоять из описаний действий внешних сущностей (событий) и соответствующих реакций системы на события. Каждое событие должно соответствовать одному или более потокам данных: входные потоки интерпретируются как воздействия, а выходные потоки — как реакции системы на входные потоки.

Для сложных систем (десять и более сущностей) 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.

Структуры потоков данных и определения их компонентов хранятся и анализируются в словаре данных. Каждая логическая функция (процесс) может быть детализирована с помощью DFD нижнего уровня; когда дальнейшая детализация перестает быть полезной, переходят к выражению логики функции при помощи спецификации процесса (мини-спецификации). Содержимое каждого хранилища также сохраняют в словаре данных, модель данных хранилища раскрывается с помощью ER-диаграмм.

В частности, в DFD не показываются процессы, которые управляют собственно потоком данных и не приводятся различия между допустимыми и недопустимыми путями. DFD содержат множество полезной информации, а, кроме того:

- позволяют представить систему с точки зрения данных;
- иллюстрируют внешние механизмы подачи данных, которые потребуют наличия специальных интерфейсов;
- позволяют представить как автоматизированные, так и ручные процессы системы;
- выполняют ориентированное на данные секционирование всей системы.

Потоки данных используются для моделирования передачи информации (или даже физических компонентов) из одной части системы в другую. Потоки на диаграммах изображаются именованными стрелками, стрелки указывают направление движения информации. Иногда информация может двигаться в одном направлении, обрабатываться и возвращаться в ее источник. Такая ситуация может моделироваться либо двумя различными потоками, либо одним двунаправленным.

Процесс преобразует входной поток данных в выходной в соответствии с действием, задаваемым именем процесса. Каждый процесс должен иметь уникальный номер для ссылок на него внутри диаграммы. Этот номер может использоваться совместно с номером диаграммы для получения уникального индекса процесса во всей модели.

Хранилище данных (data storage) позволяет на ряде участков определять данные, которые будут сохраняться в памяти между процессами. Фактически хранилище представляет "срезы" потоков данных во времени. Информацию, которую оно содержит, можно использовать в любое время после ее определения, при этом данные могут выбираться в произвольном порядке. Имя хранилища должно идентифицировать его содержимое. В случае, когда поток данных входит (выходит) в (из) хранилище и его структура соответствует структуре хранилища, он должен иметь то же самое имя, которое нет необходимости отражать на диаграмме.

Внешняя сущность (терминатор) представляет сущность вне контекста системы, являющуюся источником или приемником системных данных. Ее имя должно содержать существительное, например "Клиент". Предполагается, что объекты, представленные такими узлами, не должны участвовать ни в какой обработке.

Правила построения:

1. Все потоки данных должны начинаться или заканчиваться процессом. Данные не могут протекать непосредственно от источника до потребителя или между источником / потребителем и хранилищем данных, если они не проходят через промежуточный процесс.
2. Многочисленные потоки данных между двумя компонентами можно показывать двумя линиями потока данных или двунаправленной стрелкой.
3. Название процесса состоит из глагола, следующего за существительным. В соответствии с соглашением, названия источников, получателей и хранилищ данных использует заглавные буквы, в то время как названиям процесса и потоки данных показываются произвольно.
4. Процессы в уровне 1 диаграмма потока данных перечисляется 1, 2, 3, и так далее. Подпроцессам в декомпозированной диаграмме потока данных назначают номера, начинающиеся с номера родительского процесса.
5. Символы могут быть повторены для облегчения чтения диаграммы.

Основные принципы: принцип сохранения данных

Любые данные, которые входят в процесс, должны использоваться или воспроизводиться этим процессом. Любые выходные данные процесса должны быть введены или созданы алгоритмом в пределах процесса. Любые данные, используемые алгоритмом в пределах процесса должны быть сначала введены в процесс. Любые данные, созданные алгоритмом должны или использоваться другим алгоритмом в пределах того же самого процесса или выведены процессом.

принцип итераций

Процессы высокого уровня декомпозируются в процессы низшего уровня. На самом низком уровне - примитивные процессы, которые исполняют единственную функцию (или алгоритм).

Контекстная диаграмма (уровень 0) определяет границы системы, выдвигая на первый план источники и получатели. Выделение границы системы при изображении контекстной диаграммы помогает аналитику, пользователю и ответственным менеджерам рассматривать альтернативные логические проекты системы высокого уровня.

Уровень 1 диаграммы потока данных показывает важнейшие процессы системы, хранилища данных, источники и получатели, связанные потоками данных. Процесс уровня 1 является сложным и может включать программы, руководства, ручные процедуры, аппаратные средства ЭВМ, процедуры и другие действия.

Каждый процесс уровня 1 состоит из нескольких подпроцессов, которые внесены в список описаний процессов. Чтобы разбить диаграмму потока данных, аналитик создает независимый **уровень 2 диаграммы** потока данных для каждого процесса уровня 1.

Функциональный примитив - процесс, который не требует никакого дальнейшего разложения. Отдельные физические компоненты системы находятся на один шаг ниже функциональных примитивов.

Документирование. Элементы данных документируются в словаре данных. В процессе разработки элементы данных, которые занимают то же самое место, хранят или разделяют поток данных, формируют сложные вычисления или структуры данных также документируются в словаре данных.

Каждый процесс определен в описании процесса, которое обращает внимания на вход и элементы данных выхода и кратко описывает задачи или действия, которые он выполняет. Описания процессов иногда документируются в словаре данных.

Проверка модели включает следующие этапы:

- Проверка синтаксиса
- Проверка элементов данных
- Взаимные ссылки
- Проверка целей

10-11.9. Особенности применения функциональных и объектно-ориентированных методологий моделирования предметной области


Существуют различные методы структурного моделирования предметной области, среди которых следует выделить **функционально-ориентированные и объектно-ориентированные методологии**.

Объектные методика рассматривают моделируемую организацию как набор взаимодействующих объектов – производственных единиц. Объект определяется как осязаемая реальность – предмет или явление, имеющая четко определяемое поведение. Целью применения данной методики является выделение объектов, составляющих организацию, и распределение между ними ответственностей за выполняемые действия.

Функциональные методика рассматривают организацию как набор функций, преобразующий поступающий поток информации в выходной поток. Процесс преобразования информации потребляет определенные ресурсы.

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

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



Перечисленные недостатки функциональных моделей снимаются в **объектно-ориентированных моделях**, в которых главным структурообразующим компонентом выступает класс объектов с набором функций, которые могут обращаться к атрибутам этого класса.

С точки зрения бизнес-моделирования каждый из представленных подходов обладает своими преимуществами. При выборе методики моделирования предметной области обычно в качестве критерия выступает степень ее динамичности. Объектный подход позволяет построить более устойчивую к изменениям систему, лучше соответствует существующим структурам организации. Функциональное моделирование хорошо показывает себя в тех случаях, когда организационная структура находится в процессе изменения или вообще слабо оформлена. Подход от выполняемых функций интуитивно лучше понимается исполнителями при получении от них информации об их текущей работе.

Однако в рамках одной и той же ИС для различных классов задач могут требоваться различные виды моделей, описывающих одну и ту же предметную область. В таком случае должны использоваться комбинированные модели предметной области.