

Тема 1. Инжиниринг.

Вопрос 5. Методологии IDEF, DFD.

Методология IDEF0.

Методология SADT легла в основу не менее известных американских стандартов семейства IDEF. IDEF0 - методология функционального моделирования. С помощью наглядного графического языка IDEF0 изучаемая система предстает перед разработчиками и аналитиками в виде набора взаимосвязанных функций (функциональных блоков - в терминах IDEF0). Как правило, моделирование средствами IDEF0 является первым этапом изучения любой системы.

Исторически IDEF0 как стандарт был разработан в 1981 г. в рамках программы автоматизации промышленных предприятий, которая носила обозначение ICAM (Integrated Computer Aided Manufacturing) и была предложена департаментом Военно-Воздушных Сил США. Собственно семейство стандартов IDEF унаследовало свое обозначение от названия этой программы (IDEF=ICAM DEFinition). Участники программы ICAM столкнулись с необходимостью разработки новых методов анализа, позволяющих обеспечить групповую работу над созданием модели с непосредственным участием всех аналитиков и специалистов, занятых в рамках проекта. В результате поиска соответствующих решений родилась методология функционального моделирования IDEF0. С 1981 г. стандарт IDEF0 претерпел несколько незначительных изменений, в основном ограничивающего характера, и последняя его редакция была выпущена в декабре 1993 г. Национальным институтом по стандартам и технологиям США (NIST).

В основе методологии лежат четыре основных понятия.

Первым из них является понятие функционального блока (Activity Box). Функциональный блок графически изображается в виде прямоугольника (рис. 1) и олицетворяет собой некоторую конкретную функцию в рамках рассматриваемой системы.

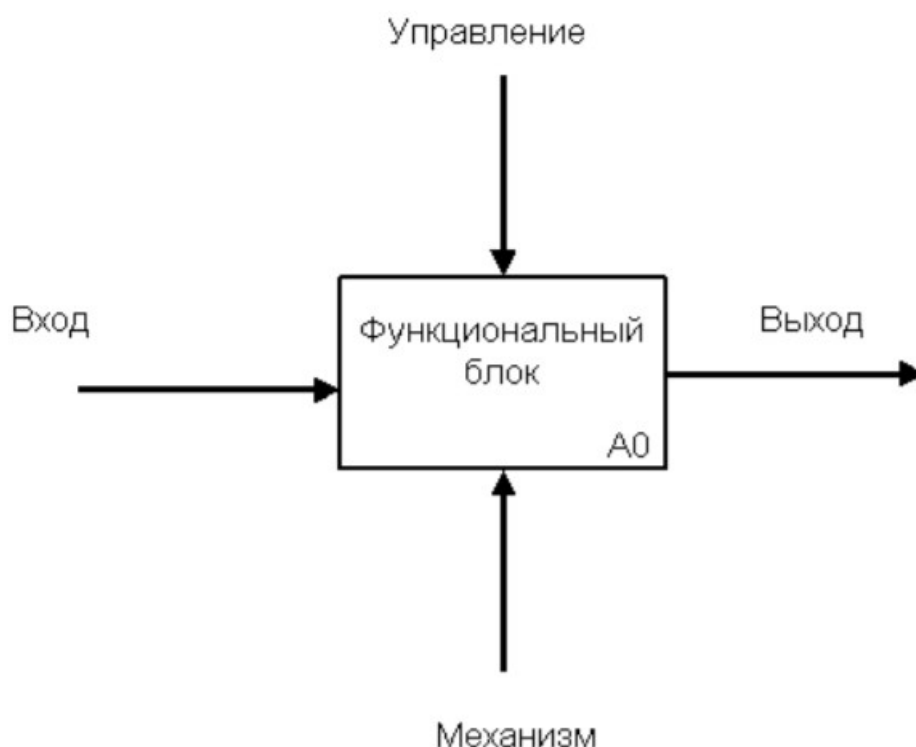


Рис. 1. Функциональный блок

В соответствии со стандартом название каждого функционального блока должно быть сформулировано в глагольном наклонении (например, «производить услуги», а не

«производство услуг»).

Каждая из четырех сторон функционального блока имеет своё определенное значение:

- верхняя сторона имеет значение «Управление» (Control);

- левая сторона имеет значение «Вход» (Input);

- правая сторона имеет значение «Выход» (Output);

- нижняя сторона имеет значение «Механизм» (Mechanism). Каждый функциональный блок в рамках единой рассматриваемой

системы должен иметь свой уникальный идентификационный номер.

Вторым основным понятием методологии IDEF0 является понятие интерфейсной дуги (Arrow), которую часто называют потоком или стрелкой. Интерфейсная дуга отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, отображенную данным функциональным блоком. Графическим отображением интерфейсной дуги является однонаправленная стрелка. Каждая интерфейсная дуга должна иметь свое уникальное наименование (Arrow Label). В соответствии со стандартом, наименование должно быть оборотом существительного.

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

В зависимости от того, к какой из сторон подходит данная интерфейсная дуга, она носит название «входящей», «исходящей» или «управляющей». Кроме того, «источником» (началом) и «приемником» (концом) каждой функциональной дуги могут быть только функциональные блоки, при этом «источником» может быть только выходная сторона блока, а «приемником» любая из трех оставшихся. При этом любой функциональный блок должен иметь, по крайней мере, одну управляющую интерфейсную дугу и одну исходящую. Это означает, что каждый процесс должен происходить по каким-то правилам (отображаемым управляющей дугой) и должен выдавать некоторый результат (выходящая дуга), иначе его рассмотрение не имеет никакого смысла.

При построении IDEF0-диаграмм важно правильно отделять входящие интерфейсные дуги от управляющих, что часто бывает непросто. Для систем одного класса всегда есть определенные разграничения. Например, в случае рассмотрения предприятий и организаций существуют пять основных видов объектов: материальные потоки (детали, товары, сырье и т.д.), финансовые потоки (наличные и безналичные, инвестиции и т.д.), потоки документов (коммерческие, финансовые и организационные документы), потоки информации (информация, данные о намерениях, устные распоряжения и т.д.) и ресурсы (сотрудники, станки, машины и т.д.). При этом в различных случаях входящими и исходящими интерфейсными дугами могут отображаться все виды объектов, управляющими - только относящиеся к потокам документов и информации, а дугами - механизмами - только ресурсы.

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

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

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

Модель IDEF0 всегда начинается с представления системы как единого целого - одного

функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области. Такая диаграмма с одним функциональным блоком называется контекстной диаграммой и обозначается идентификатором «А-0».

В пояснительном тексте к контекстной диаграмме должна быть указана цель (Purpose) построения диаграммы в виде краткого описания и зафиксирована точка зрения (Viewpoint) относительно основного направления развития модели и уровня необходимой детализации.

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

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

Принципы ограничения сложности IDEF0-моделей:

- ограничение количества функциональных блоков на диаграмме тремя-шестью. Верхний предел (шесть) заставляет разработчика использовать иерархии при описании сложных предметов, а нижний предел (три) гарантирует, что на соответствующей диаграмме достаточно деталей, чтобы оправдать ее создание;
- ограничение количества подходящих к одному функциональному блоку (выходящих из одного функционального блока) интерфейсных дуг четырьмя.

Организация работы по разработке IDEF0-модели.

Обычно процесс разработки является итеративным и состоит из следующих условных этапов:

1. Создание группы специалистов, относящихся к различным сферам деятельности предприятия, для разработки IDEF0-модели. Эта группа в терминах IDEF0 называется авторами (Authors). Построение первоначальной модели является динамическим процессом, в течение которого авторы опрашивают компетентных лиц о структуре различных процессов.

На основе имеющихся положений, документов и результатов опросов создается черновик (Model Draft) модели.

2. Распространение черновика для рассмотрения, согласований и комментариев. На этой стадии происходит обсуждение черновика модели с широким кругом компетентных лиц (в терминах IDEF0-читателей) на предприятии.

3. Официальное утверждение модели. Согласованная модель утверждается руководителем рабочей группы в том случае, если у авторов модели и читателей отсутствуют разногласия по поводу ее адекватности. Окончательная модель представляет собой согласованное представление о предприятии (системе) с заданной точки зрения и для заданной цели.

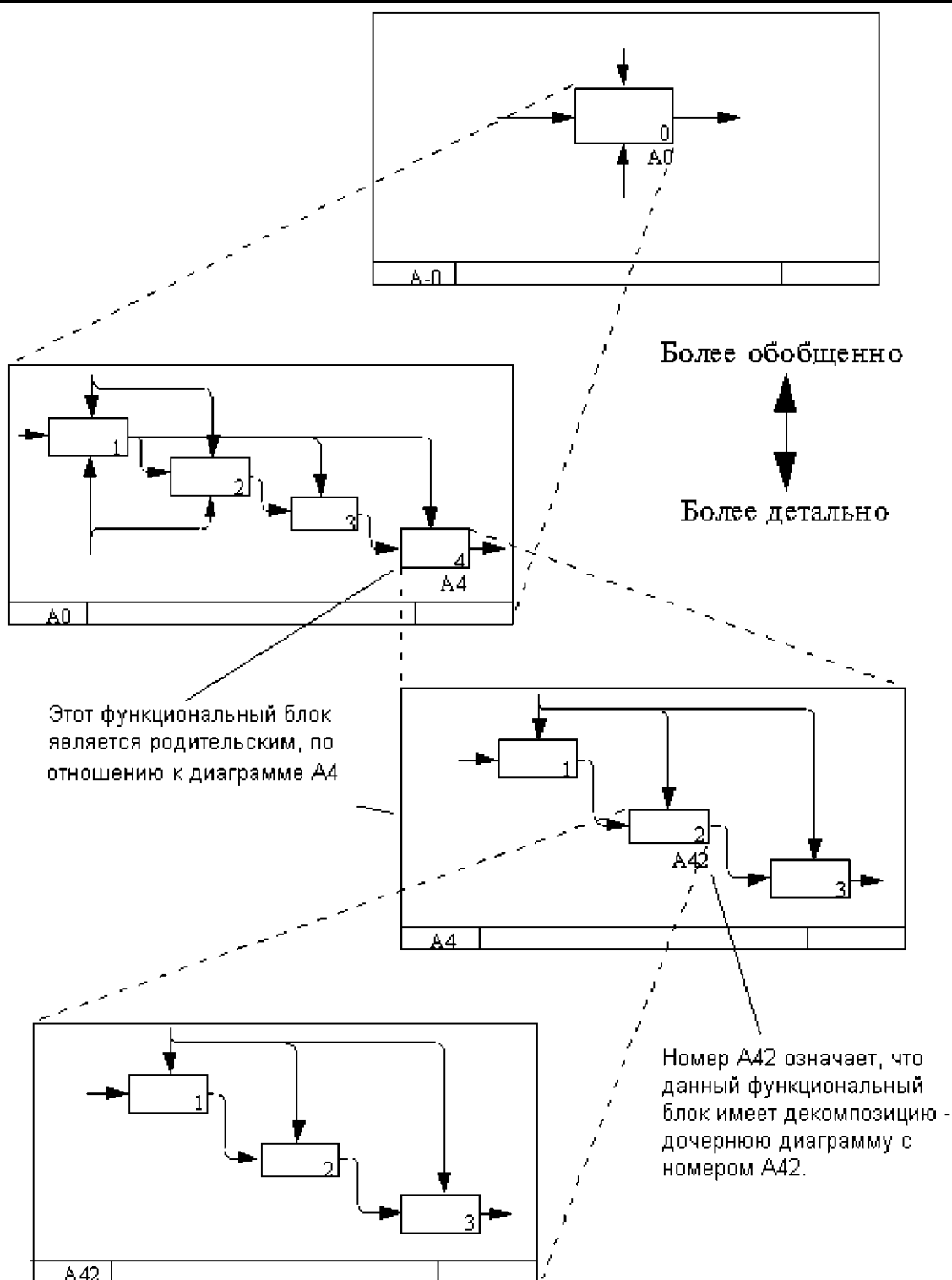


Рис. 2. Декомпозиция функциональных блоков

Объекты и связи в IDEF0.

Основным объектом диаграммы процессов в нотации IDEF0 является объект Activity. Графически он представляет собой четырехугольник, изображающий функции, выполняемые в организации. Второй основной составляющей стандарта IDEF0 являются стрелки (рис. 3). На диаграмме процесса в IDEF0 стрелки, входящие в функцию слева, служат для описания потоков материальных ресурсов или потоков информации, документов.



Рис. 3. Формирование модели бизнес-процесса. Шаг 1.

Входящие ресурсы преобразуются функцией (работой, процессом). Результатом этого преобразования являются материальные выходы или информация, которые показываются в виде стрелок, выходящих из правой стороны четырехугольника.

Для выполнения любой реальной работы необходимы основные средства, инструменты, персонал, программные продукты и т.д. Все эти ресурсы отображаются на диаграмме стрелками, входящими в четырехугольник снизу.

Что еще необходимо показать на диаграмме для того, чтобы можно было описать реальный процесс организации? Следует отобразить управляющие воздействия, которые определяют порядок выполнения работы, управляют работой. Такими воздействиями могут быть, например, устное распоряжение руководителя, нормативный документ, государственный, отраслевой стандарт, технические условия и т.д. Управляющие воздействия показываются на диаграмме стрелками сверху. Любое управляющее воздействие существует в виде определенной информации, поэтому стрелки сверху в нотации IDEF0 обозначают управляющие информационные потоки.

Следует подчеркнуть, что порядок отображения стрелок должен строго соблюдаться при формировании моделей. Каждая сторона четырехугольника определяет тип стрелки.

Нарушать эти правила нельзя. В противном случае создаваемые модели не только не соответствуют стандарту, но их невозможно будет читать.

Все стрелки начинаются от края диаграммы и подходят к функциям. Таким образом, край диаграммы в IDEF0 имеет глубокий смысл.

Моделирование процессов в нотации IDEF0 начинается с создания так называемой контекстной диаграммы, описывает деятельность организации или процесса в целом. На контекстной диаграмме отображаются важнейшие входы и выходы, механизмы, необходимые для работы, а также управляющие воздействия.

Обратные связи по управлению и информации - возможность отражения реального процесса.

Начнем описание процесса с того, что поместим на диаграмму три функции, как показано на рис. 4.

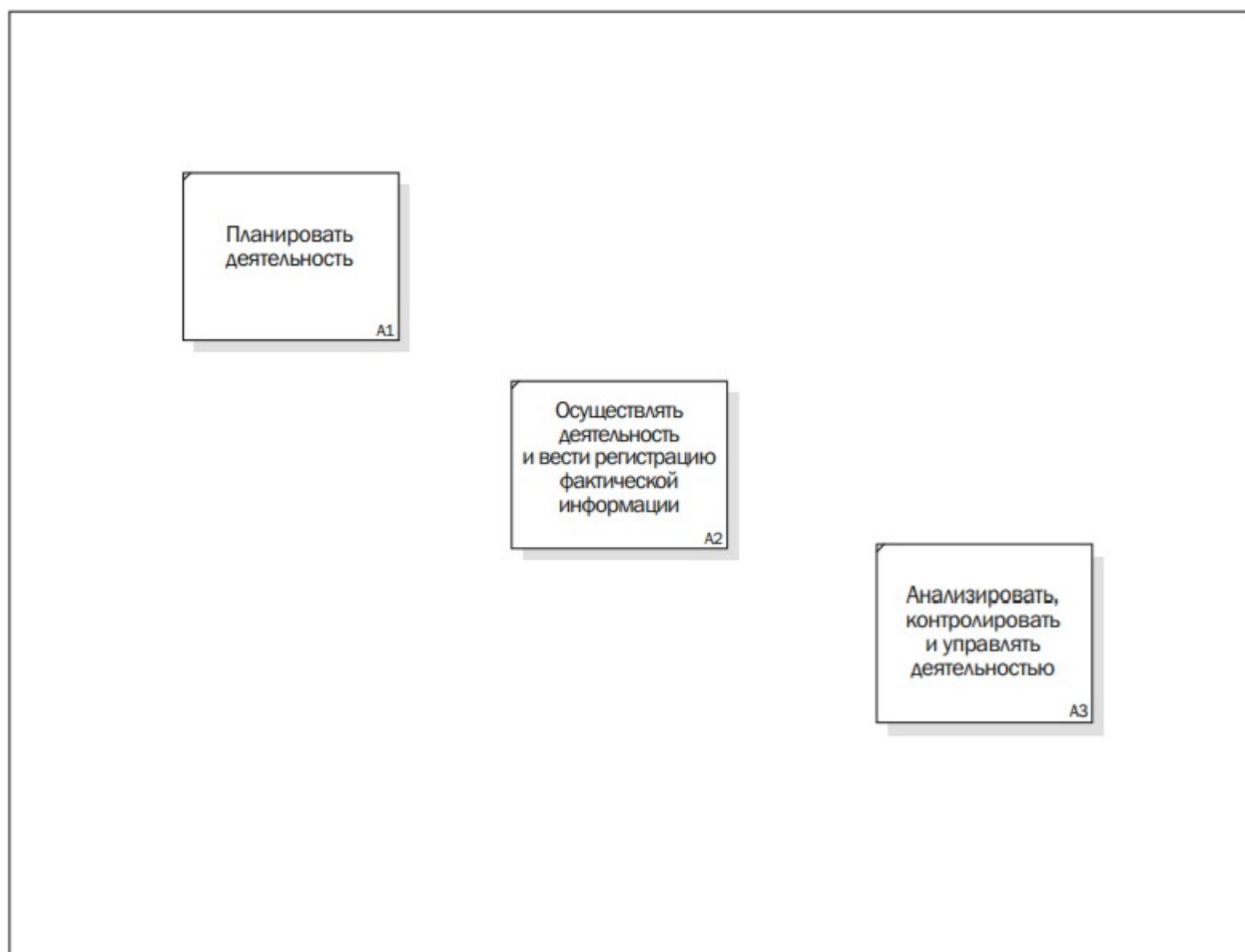


Рис. 4. Формирование модели бизнес-процесса. Шаг 2.

Первую функцию назовем «Планировать деятельность», вторую - «Осуществлять деятельность и вести регистрацию фактической информации», третью - «Анализировать, контролировать и управлять деятельностью». Обратим внимание, что для наименования функций могут быть использованы только глаголы или отглагольные существительные. Это одно из базовых требований нотации. Было бы, например, неправильно называть объект «Начальник коммерческого отдела» или «Отдел закупок».

Важнейшими требованиями нотации являются количество объектов на диаграмме и количество стрелок, входящих в каждую сторону четырехугольника. В стандарте

рекомендовано располагать на одной диаграмме не более шести и не менее двух функций. С каждой стороны в четырехугольник может входить не более шести стрелок одновременно. Оба этих требования ограничивают количество объектов на диаграмме процесса и заставляют аналитика более тщательно продумывать схему создаваемого процесса.

Объекты на диаграмме расположены в шахматном или так называемом порядке доминирования.

Рассмотрим рис. 5.

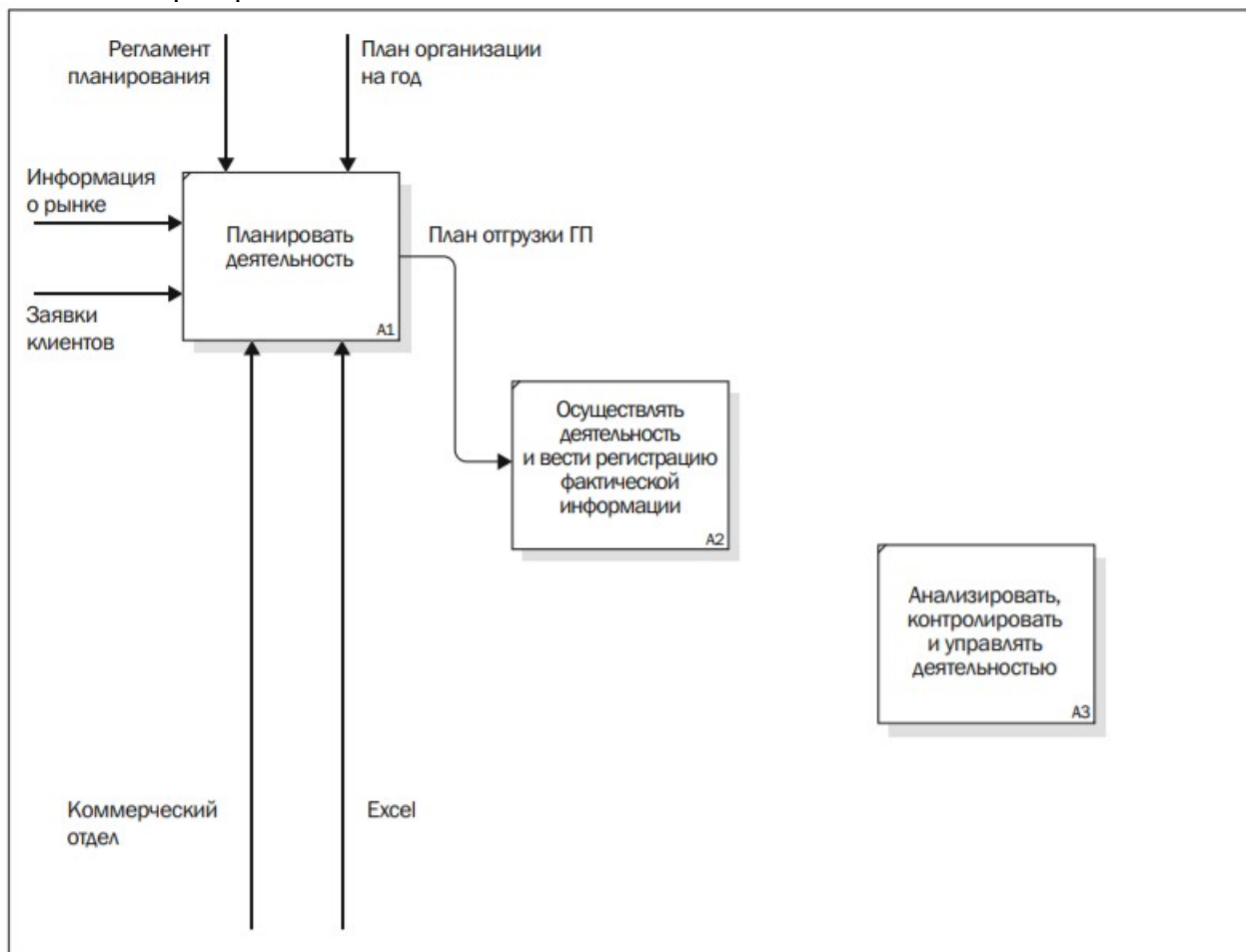


Рис. 5. Формирование модели бизнес-процесса. Шаг 3.

Представим себе, что функцию планирования выполняет Коммерческий отдел (КО), который использует при этом средство автоматизации MS Excel. Для планирования КО использует информацию о рынке (прайс-листы и т.п.) и заявки клиентов. Регламентируется деятельность КО «Регламентом планирования», «Планом организации на год». Результатом работы КО является «План отгрузки ГП (готовой продукции)». Посмотрим, как эта информация будет отображена на диаграмме.

Рассмотрим функцию «Осуществлять деятельность...». Ее выполняет Производственный отдел (ПрО) и Цех. Для выполнения работ требуется сырье и материалы. Работы регламентируются нормативами на расход сырья государственными, отраслевыми стандартами, техническими условиями, требованиями клиента. Для работы ПрО требуется АСУ ТП собственной разработки. Для производства готовой продукции Цеху необходимы станки и прочее оборудование, т.е. основные средства (ОС).

Результат работы ПрО и Цеха - готовая продукция, которая является выходом функции «Осуществлять деятельность и вести регистрацию фактической информации». Кроме того,

выходом этой функции является также фактическая информация о выполнении плана производства и отгрузки. На рис. 6 показаны все перечисленные выше ресурсы и информация.

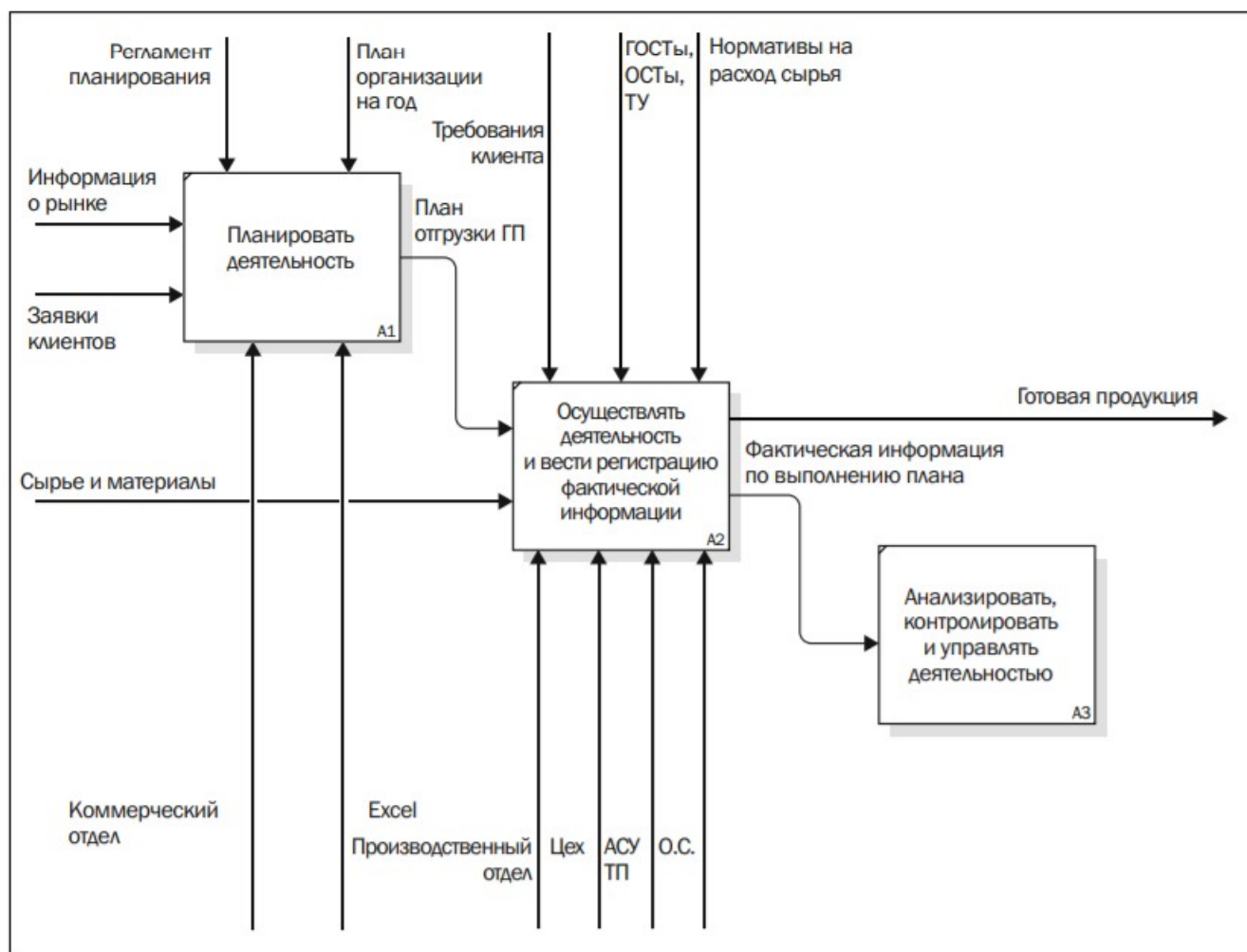


Рис. 6. Формирование модели бизнес-процесса Шаг 4

Нам осталось показать входы и выходы функции «Анализировать, контролировать и управлять деятельностью». Кто должен ее выполнять? Для нашего примера будем считать, что контролирует работу тот, кто ее планирует, т.е. КО.

В своей работе по анализу и контролю КО руководствуется регламентом анализа и контроля. Не стоит забывать и годовой план работы организации в целом. Для работы КО использует MS Excel.

Результатом работы КО является отчет для руководства организации «План/ факт», как показано на рис. 7.

Вы заметили, что стрелка, изображающая КО (как и MS Excel), не повторяется на диаграмме дважды? Она ветвится. Ветвление стрелок - прекрасный инструмент, позволяющий сделать диаграмму процесса более наглядной.

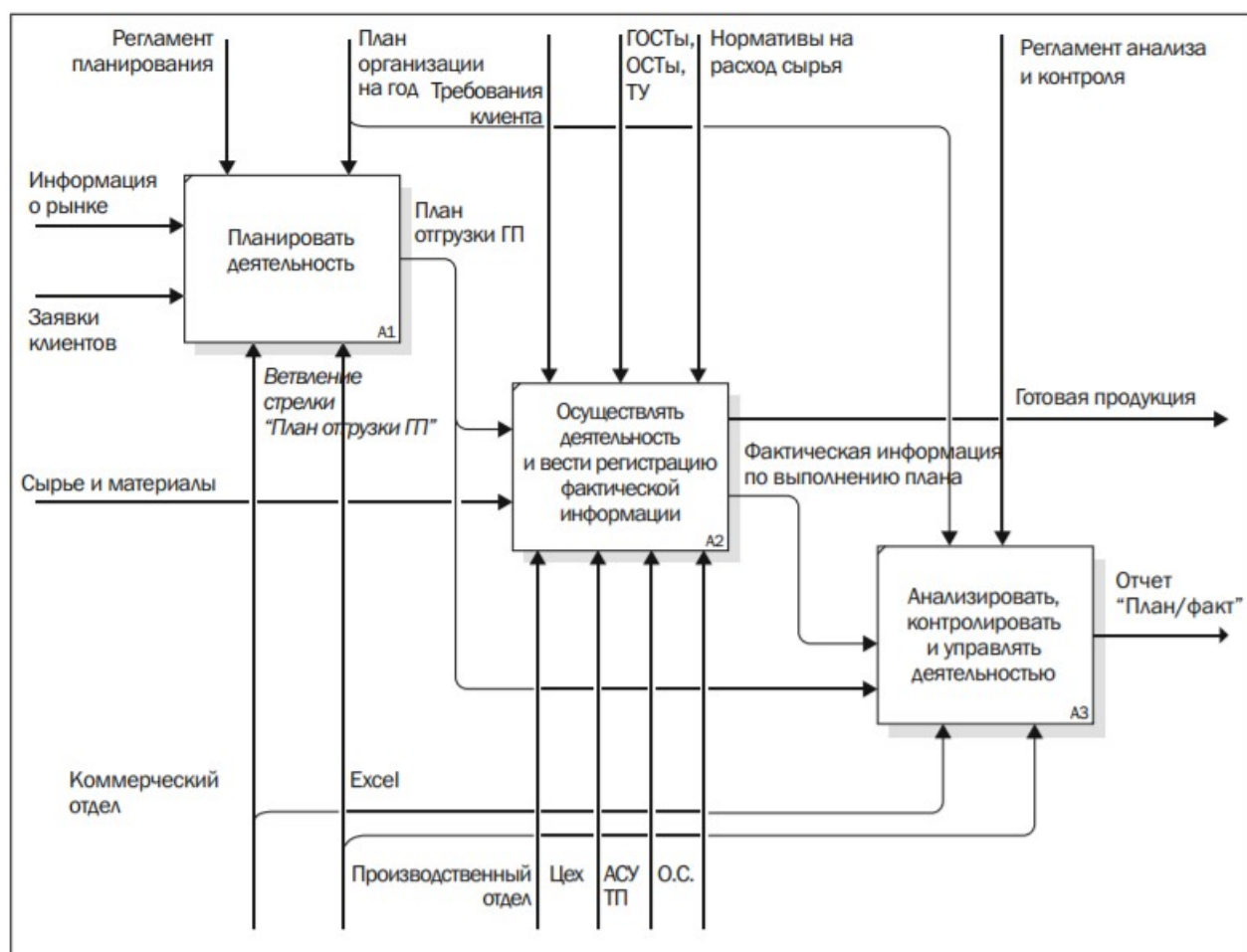


Рис. 7. Формирование модели бизнес-процесса Шаг 5

Итак, диаграмма готова. Что же забыли на ней указать? Каким образом осуществляется управление этим циклическим процессом? Очевидно, что необходимо отобразить на схеме процесса, по крайней мере, два типа обратных связей по информации и управлению (рис. 10).

Первым типом обратной связи в диаграммах IDEF0 являются обратные связи по информации. Они показываются в виде стрелок, выходящих из правой стороны четырехугольника и входящих в левую сторону другого четырехугольника. Обратные связи этого типа на диаграмме процесса обязательно отображаются снизу, т.е. обходят функции снизу. В нашем примере покажем обратную связь по информации «Информация для корректировки плана». Стрелка, отображающая эту обратную связь, выходит из правой стороны четырехугольника «Анализировать, контролировать и управлять деятельностью» и входит в левую сторону четырехугольника «Планировать деятельность». Таким образом, мы отобразили на диаграмме процесса тот факт, что КО регулярно анализирует выполнение плана и, в случае отклонений от него, формирует информацию, необходимую для корректировки плана на следующий период.

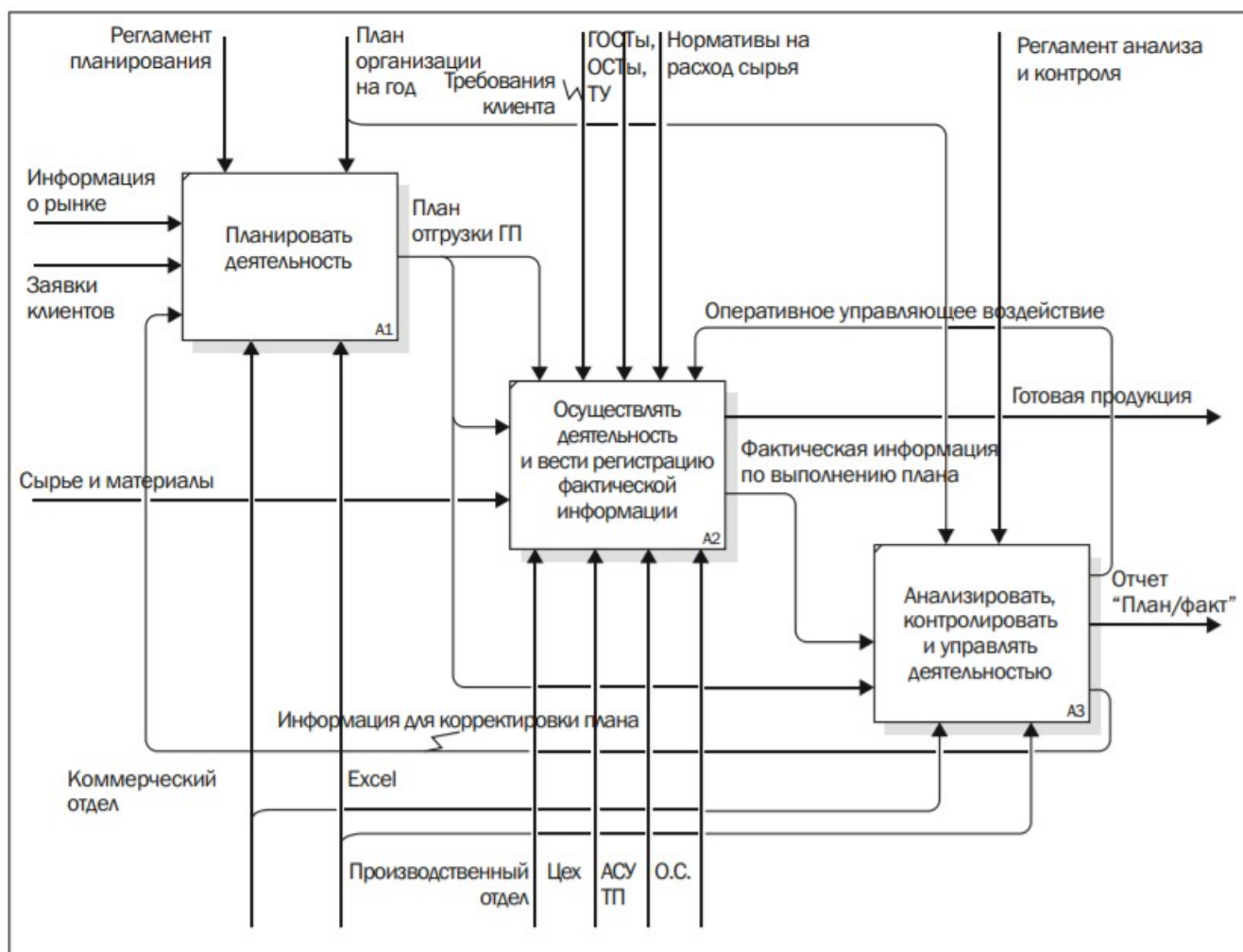


Рис. 8. Формирование модели бизнес-процесса Шаг 6

Итак, обратные связи по информации позволяют отобразить на диаграммах информационные потоки, необходимые для корректировки действий, выполняемых по ходу бизнес-процесса.

Вторым видом обратной связи является обратная связь по управлению. Возможность отображения этих обратных связей - важнейшее преимущество нотации IDFE0. Обратная связь по управлению отличается от обратной связи по информации тем, что стрелка, изображающая ее на диаграмме, обходит ее сверху функций и входит в верхнюю сторону четырехугольника.

В нашем примере покажем обратную связь по управлению «Оперативное управляющее воздействие» в виде стрелки, выходящей из правой стороны четырехугольника «Анализировать, контролировать и управлять деятельностью» и входит в верхнюю сторону четырехугольника «Осуществлять деятельность...». Эта обратная связь означает, что при анализе и контроле выполнения плана, КО принимает оперативные управленческие решения, регулирующие выполнение работ ПрО и Цеха по производству готовой продукции.

Рассмотренный пример показывает, что при формировании моделей процессов в IDEF0 можно (и нужно!) эффективно использовать стрелки, отображающие обратные связи по информации и управлению.

Некоторые правила ветвления и слияния стрелок.

Помимо ветвления стрелки могут также сливаться. Подробно правила ветвления и слияния стрелок описаны в стандарте IDEF0. Здесь же мы приведем несколько важных примеров использования этих правил.

На рис. 9 показаны ситуации правильного и неправильного наименования стрелок при ветвлении и слиянии.

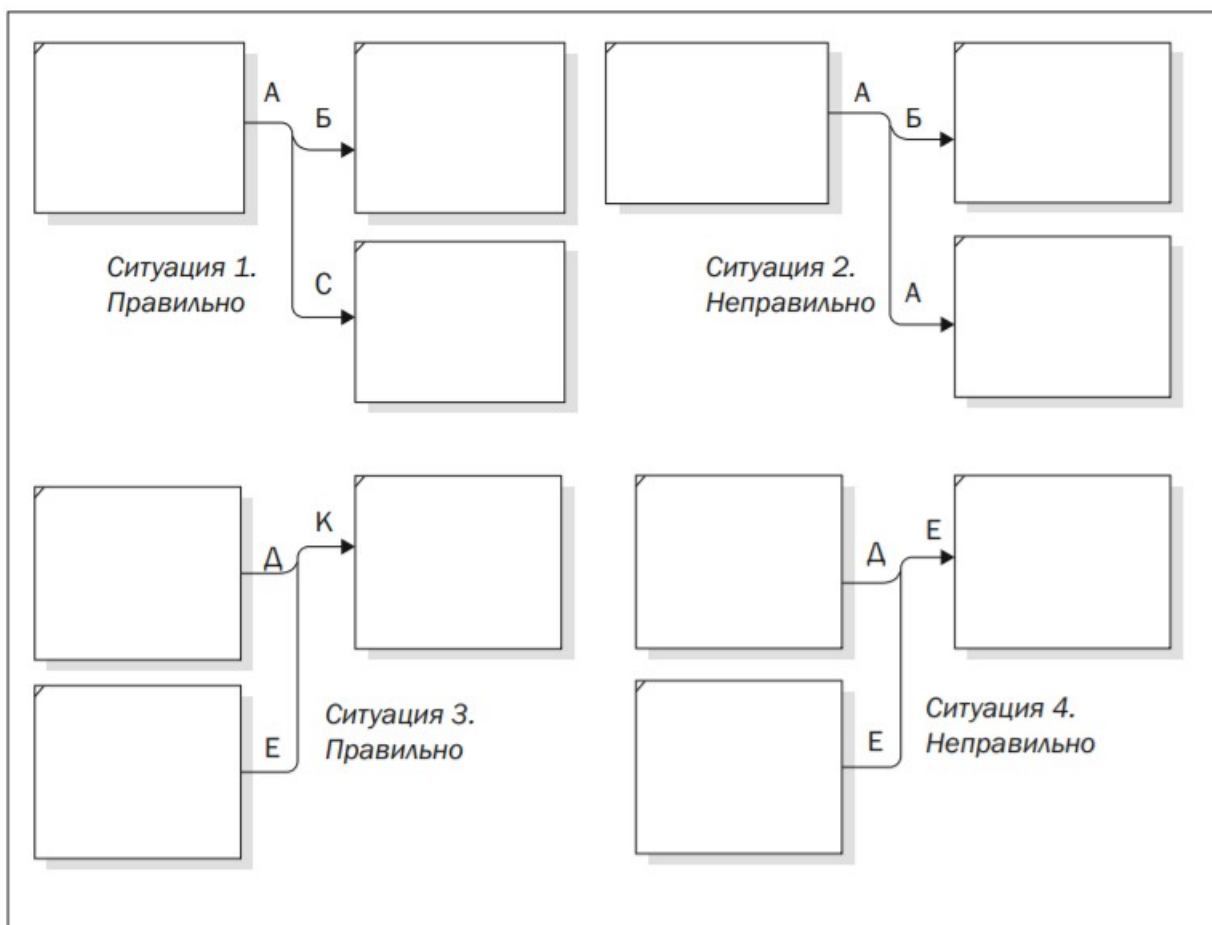


Рис.

9. Правила ветвления и слияния стрелок

Ветвление стрелок в ситуации 1 означает, что поток ресурсов А содержит в себе потоки Б и В. Например, план продаж может включать в себя план по отгрузке в натуральном выражении и план отгрузки в стоимостном выражении.

Ветвление стрелок в ситуации 2 недопустимо, так как оно означало бы, что поток А содержит в себе одновременно и А и Б, что некорректно.

Аналогично можно рассмотреть ситуации 3 и 4 слияния стрелок.

На рис. 10 показано, как можно пользоваться механизмом ветвления и слияния стрелок при построении диаграмм процессов в IDEF0. Стрелка, входящая на диаграмму процесса, ветвится на несколько других, отражающих более детально поток ресурсов или информации. Исходящие стрелки сливаются, показывая, как формируется результат выполнения процесса в целом. Сказанное справедливо также для стрелок сверху - управляющих воздействий и стрелок снизу - механизмов (персонал, инфраструктура).

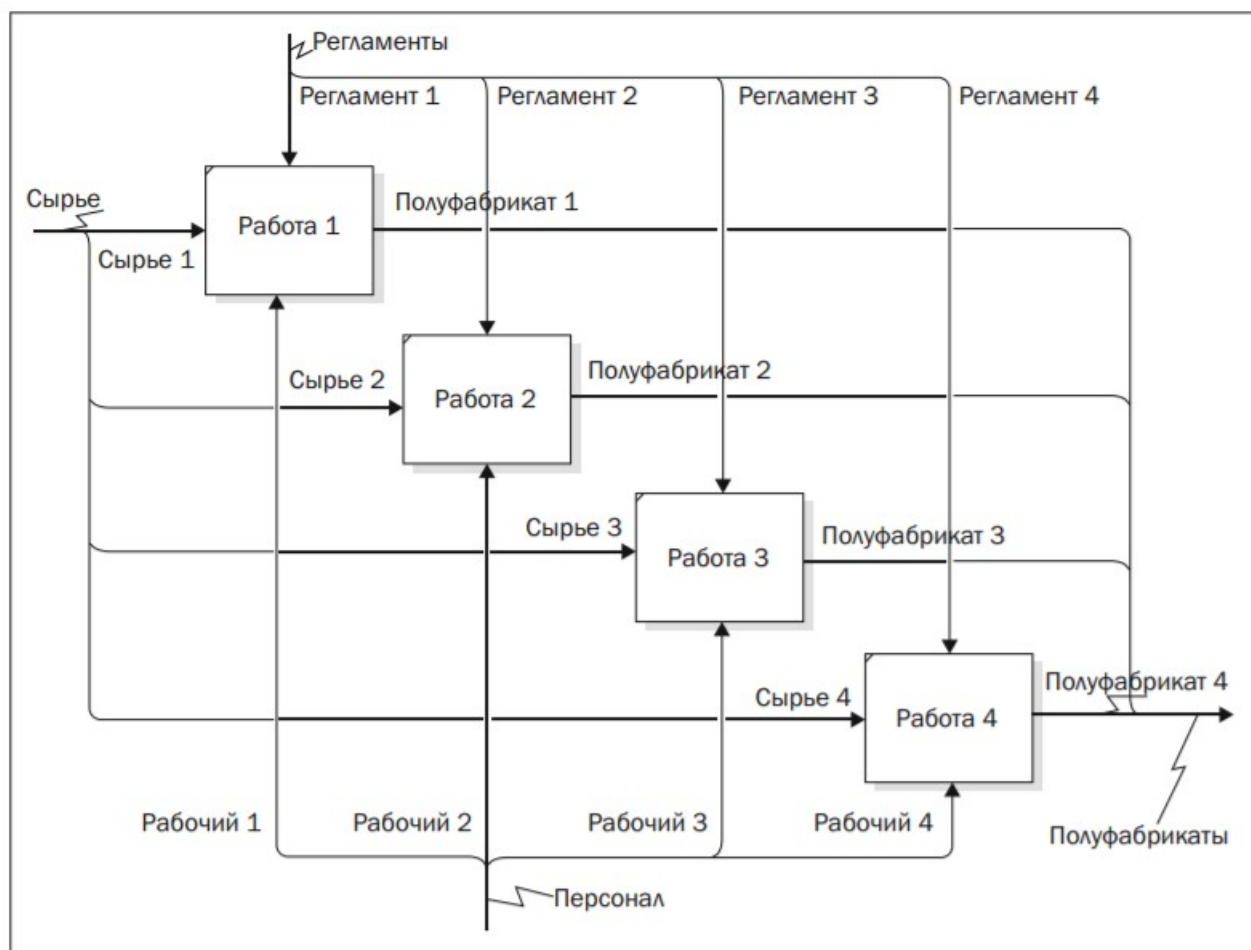


Рис. 10. Пример ветвления и слияния стрелок.

Таким образом, ветвление и слияние стрелок позволяет показывать потоки ресурсов и информации сначала укрупненно, что важно для описания процессов на верхнем уровне, а затем более детально - для диаграмм процессов нижнего уровня. Указанный механизм эффективно используется при построении диаграмм IDEF0 при декомпозиции моделей бизнес-процессов.

Ветвление и слияние стрелок является важнейшим инструментом для создания моделей в IDEF0. Особенно наглядным этот факт становится при осуществлении декомпозиции моделей процессов с верхнего уровня на нижний.

Миграция и «туннелирование» стрелок. Принципы декомпозиции в IDEF0.

Важнейшим понятием нотации IDEF0 является понятие «туннелирования» стрелок. Выполним декомпозицию функции «Осуществлять деятельность» (рис. 11). На более детальном уровне эта функция включает в себя следующие функции (работы):

1. «Разрабатывать график производства»;
2. «Выполнять подготовку производства»;
3. «Изготавливать готовую продукцию»;
4. «Хранить готовую продукцию на складе»;
5. «Отгружать готовую продукцию клиенту».

При первом шаге декомпозиции мы получим схему процесса, на которой показаны стрелки, которые не будут входить ни в один четырехугольник. Стрелки «мигрировали» на уровень вниз. Теперь необходимо «подвязать» их к конкретным функциям, при этом можно

использовать механизм ветвления и слияния стрелок. Обратим внимание, что все стрелки, показанные на верхнем уровне, будут показаны и на нижнем уровне. Таким образом, сохраняется связность при моделировании бизнес-процесса - детальные процессы оказываются однозначно связанными с процессами верхнего уровня и наоборот.

Теперь необходимо подвести каждую из показанных на рис. 13 стрелок к соответствующему объекту - функции.

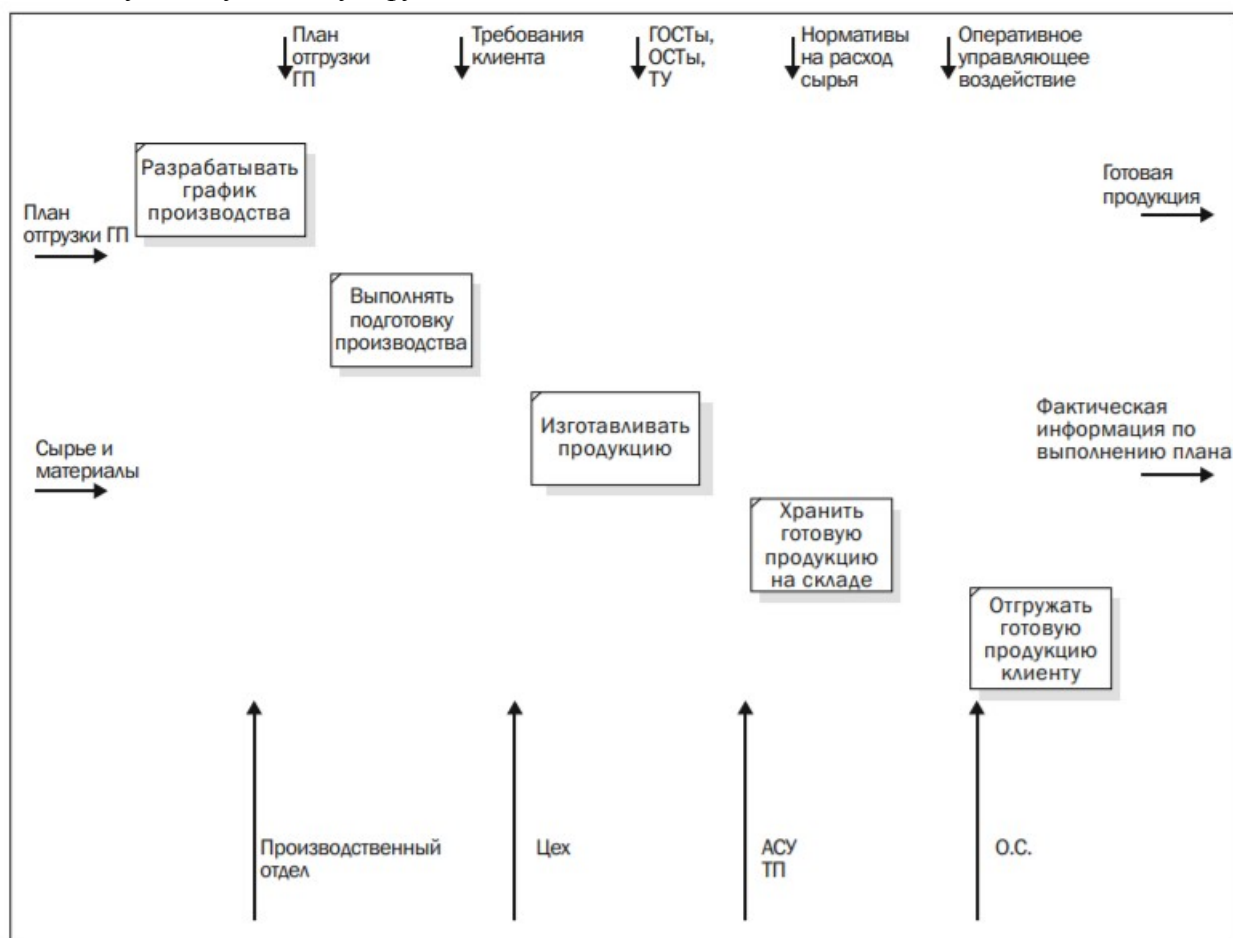


Рис. 11. Формирование модели бизнес-процесса. Шаг 7.

Функцию «План отгрузки ГП» подводим к функции «Разрабатывать график производства». К этой функции сверху также будут подведены «Требования клиента» и «План отгрузки ГП», но уже в виде управляющего воздействия. Выходом первой функции являются управляющее воздействие «График производства» и информационный поток «Данные графика производства».

Входящая стрелка «Сырье и материалы» ветвится на две стрелки: «Вспомогательное сырье» и «Основное сырье и материалы».

Выходом второй функции процесса «Выполнять подготовку производства» являются «Данные по готовности оборудования».

Третья функция процесса «Изготавливать готовую продукцию» использует входящие материальные ресурсы «Основное сырье и материалы» и информацию «Данные графика производства» и «Данные по готовности оборудования». Выходами третьей функции являются «Данные по производству ГП», «ГП на складе» (готовая продукция, отгружаемая на склад) и «Брак». Обратим внимание, что выход «Брак» (стрелка и наименование выделены жирным шрифтом) не был показан на диаграмме верхнего уровня, а появился только сейчас, при подробном описании (рис. 12). Почему это могло произойти? Занимаясь описанием

процесса на верхнем уровне, мы вполне могли забыть некоторый из выходов, либо, посчитав его малозначимым, просто опустить. На диаграмме процесса более низкого уровня этот выход должен быть отражен.

Четвертая функция процесса «Хранить готовую продукцию на складе» формирует выходы: «Данные по запасам ГП» и «ГП на складе». При ее описании, однако, пришлось дополнительно ввести в рассмотрение и отобразить в виде стрелок исполнителя «Склад ГП» и управляющий вход «Условия хранения ГП на складе».

Все четыре новых входа, которые отсутствовали на диаграмме верхнего уровня и появились на рис. 12, показаны жирным шрифтом.

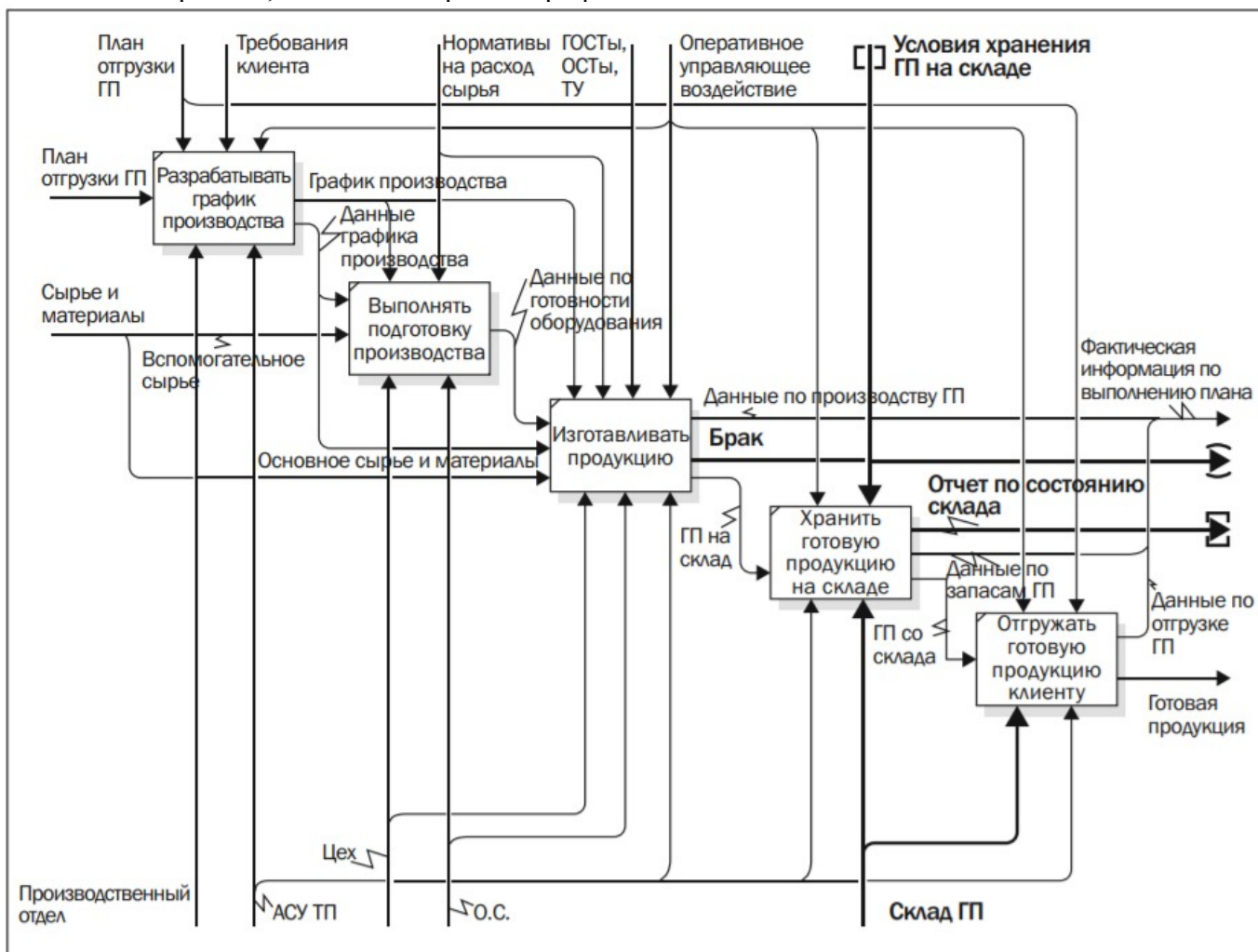


Рис. 12. Формирование модели бизнес-процесса. Шаг 8.

Начало стрелки «Условия хранения ГП на складе» заключено в квадратные скобки. Это условное обозначение появляется, когда мы показываем на диаграмме новую более низкого уровня стрелку, которой нет на диаграмме верхнего уровня. Для стрелок, входящих в диаграмму процесса, квадратные скобки указывают в начале стрелки. Для новых стрелок, являющихся исходящими, квадратные скобки указывают в конце, как, например, для стрелки «Отчет по состоянию склада».

Квадратные скобки означают, что нарушена нотация описания процесса. Чтобы устранить возникшее противоречие с нотацией, необходимо либо сделать стрелку «туннельной», либо разрешить ее миграцию на диаграмму верхнего уровня. Так, например, стрелка «Брак» сделана «туннельной». Ее не отображают на диаграмме верхнего уровня, она

будет видна только на текущей диаграмме. «Туннельные» стрелки заключают в круглые скобки.

В случае со стрелкой «Склад ГП» - другая ситуация: мы разрешили противоречие с нотацией, устранив квадратные скобки и обеспечив миграцию стрелки «Склад ГП» на диаграмму верхнего уровня.

Таким образом, механизм туннелирования стрелок может быть эффективно использован при проведении декомпозиции бизнес-процессов. На диаграммах процесса верхнего уровня мы отображаем потоки ресурсов и информации укрупненно. При декомпозиции каждый раз мы можем отображать все более детальные потоки, при этом схема процесса усложняется за счет большего количества стрелок.

Нумерация объектов на диаграммах.

Каждый объект (функция, работа) на диаграмме процесса в нотации IDEF0 может быть пронумерован. Существует несколько способов нумерации. Мы рассмотрим наиболее простой и часто применяемый способ - дерево функций процесса (рис. 13).



Рис. 13. Диаграмма дерева функций.

Как видно из рис. 13, нумерация диаграмм идет сверху вниз - от диаграммы верхнего уровня к диаграммам нижнего уровня. Каждая диаграмма нижнего уровня получает свой номер на основе номера родительской диаграммы верхнего уровня. Например, функции «Осуществлять деятельность...» присвоен номер A2, а функции процесса более низкого уровня - номера A21 - A25. Если мы декомпозируем функциям A22, то функции более детального процесса будут присвоены номера A221 - A22N. Использование рассмотренного механизма нумерации делает отслеживание функций процессов достаточно наглядным. Напомним, что количество функций на одной диаграмме не должно составлять более шести (иногда допускается восемь). В этом случае по номеру функции всегда можно однозначно определить уровень процесса.

Преимущества и недостатки использования IDEF0 для описания бизнес-процессов.

Методология моделирования бизнес-процессов IDEF0, на наш взгляд, предназначена для описания процессов верхнего уровня. Описывая такие процессы, аналитик уделяет огромное внимание управлению процессами, обратным связям по управлению и информации. Приведем основные преимущества и недостатки методологии IDEF0:

Важнейшей характерной чертой IDEF0 является полнота описания бизнес-процесса, которая достигается за счет наличия средств, отображающих управляющие воздействия, обратные связи по управлению и информации. Методология IDEF0 представляет аналитику возможность не заботиться о комплексности декомпозиции путем использования механизмов мигрирования и туннелирования стрелок. Такой механизм обеспечивает связность создаваемых диаграмм между собой. Кроме того, она делает модель процесса наглядной. Использование возможности разделения и слияния стрелок также способствует созданию более наглядных и проработанных моделей. Резюмируя, можно сказать, что жесткие требования по формированию моделей в IDEF0 в сочетании с гибкими средствами представления потоков информации и ресурсов, обеспечивают создание IDEF0-моделей стандартного вида.

Основным преимуществом методологии IDEF0 является также соответствие формата представления процесса в IDEF0 определению процесса МС ИСО 9000:2005, что позволяет выбирать IDEF0 в качестве внутреннего стандарта организации, регламентирующего описание бизнес-процессов.

К недостаткам IDEF0 можно отнести сложность восприятия схем процессов сотрудниками организации, особенно руководителями. Следует отметить, однако, что эффективное применение любой нотации предполагает обучение как сотрудников, так и руководителей умению читать и анализировать схемы процессов.

Кроме того, применяя IDEF0, сложно увязывать между собой модели нескольких процессов (например, сбыт и производство) при необходимости создания отдельных моделей для каждого из этих процессов. Однако недостаток является, скорее, техническим и может быть устранен путем предварительных договоренностей о правилах моделирования.

Таблица 1. Преимущества и недостатки методологии IDEF0

Преимущества	Недостатки
<ul style="list-style-type: none"> – Полнота описания бизнес-процесса (управление, информационные и материальные потоки, обратные связи). – Комплексность при декомпозиции (мигрирование и туннелирование стрелок). – Возможность агрегирования и детализации потоков данных и информации (разделение и слияние стрелок). – Наличие жестких требований методологии, обеспечивающих получение моделей процессов стандартного вида. – Простота документирования процессов. – Соответствие подхода к описанию процессов в IDEF0 стандартам ИСО 9000:2005 	<ul style="list-style-type: none"> – Сложность восприятия (большое количество стрелок). – Большое количество уровней декомпозиции. – Трудность увязки нескольких процессов представленных в различных моделях одной и той же организации

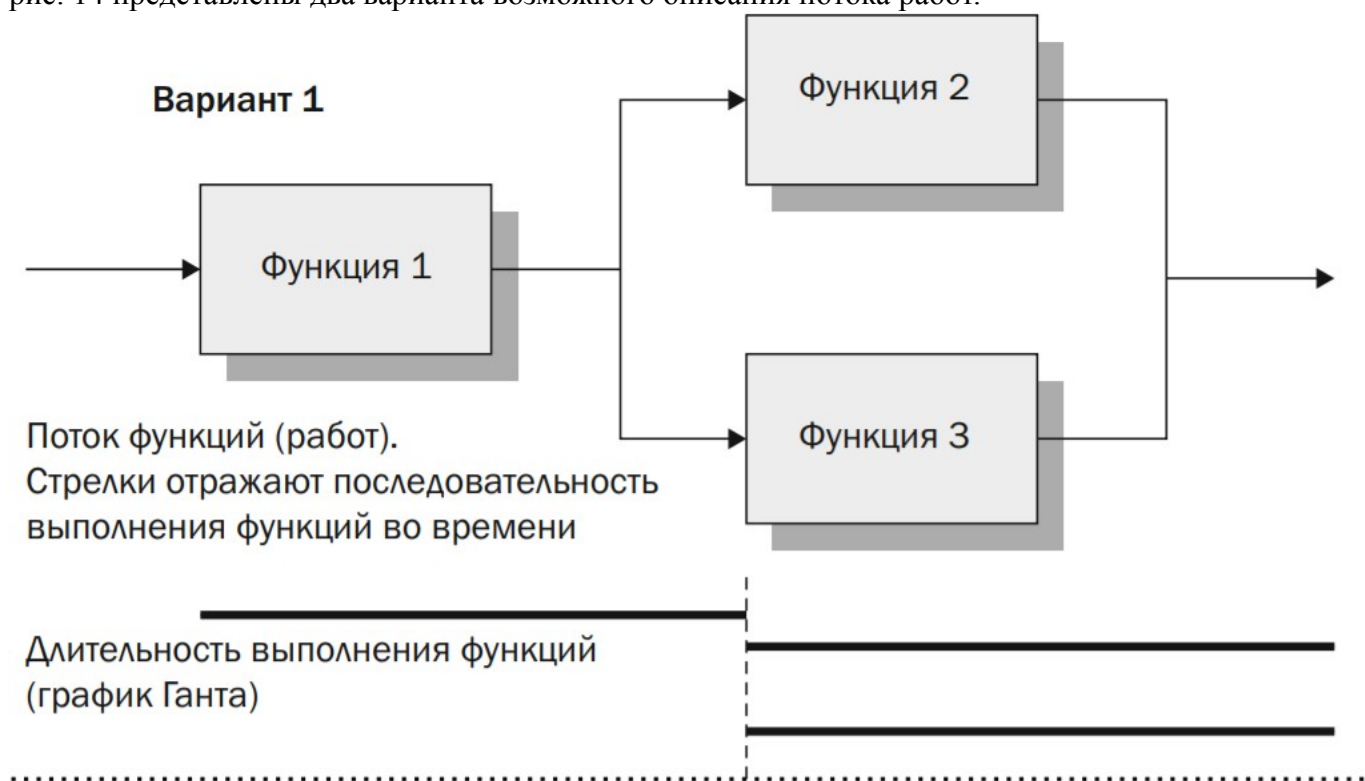
На практике часто встречаются ситуации, когда модели IDEF0 используют для

описания последовательно выполняемых работ. В таких моделях, как правило, слабо отражено управление процессом, не указаны руководители, почти нет обратных связей. На наш взгляд, использовать IDEF0 для описания последовательно выполняемых работ некорректно.

Методология IDEF3.

Нотация IDEF3 является второй важнейшей нотацией (после IDEF0) и предназначена для описания потоков работ (Work Flow Modeling). IDEF3 широко используется для создания моделей бизнес-процессов организации на нижнем уровне - при описании работ, выполняемых в подразделениях и на рабочих местах. Следует отметить, что нотация IDEF3 была взята за основу при создании методики описания процессов ARIS eEPC - «расширенной цепочки процесса, управляемого событиями».

Основными графическими объектами модели, используемыми в IDEF3, являются четырехугольники и стрелки. Первые служат для описания функций (работ, процессов), вторые - для отражения в модели последовательности выполнения функций во времени либо последовательности выполнения функций, обусловленной потоком материальных ресурсов. Прежде чем перейти непосредственно к нотации IDEF3, рассмотрим следующий пример. На рис. 14 представлены два варианта возможного описания потока работ.



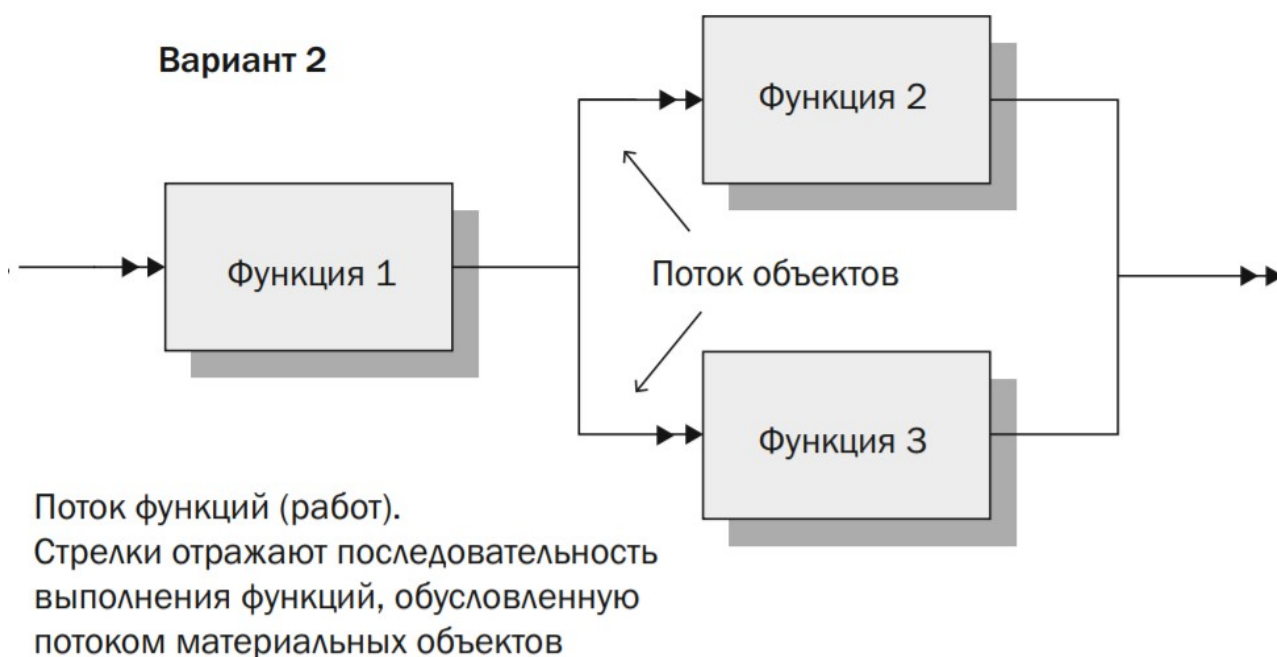


Рис. 14. Описание потоков работ

На рис. 14 (вариант 1) показано, что вначале выполняется функция 1. После завершения выполнения этой функции начинают одновременно выполнять функции 2 и 3. Стрелки в этом случае показывают нам, каким образом завершение выполнения одной функции влияет на начало выполнения другой.

Процесс варианта 2 построен по-другому. Начало выполнения функций здесь обусловлено поступлением на вход некоторых материальных ресурсов (вход функции 1), окончание - выходом материальных ресурсов (выход функции 1). Потоки ресурсов определяют начало выполнения следующих функций процесса (функций 2 и 3) и т.д.

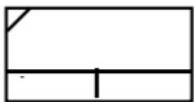
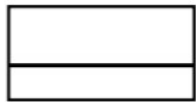






Чем плохи способы описания процессов, представленные на рис. 14? Дело в том, что построенные таким образом схемы процессов невозможно однозначно понять (прочитать). Функции 2 и 3 могут выполняться не одновременно. Например, может сложиться ситуация, когда потребуется выполнение либо функции 2, либо функции 3 процесса. Очевидно, что в этом случае выбранный нами способ описания процесса не позволит сделать вывод, какой же вариант развития событий реализуется на самом деле. Вернемся к нотации IDEF3.

Для того чтобы избежать неоднозначности описания потоков работ, в нотации IDFE3 определены дополнительные объекты, служащие для отображения возможных вариантов ветвления и слияния потоков работ, реализующихся при определенных условиях. Указанные объекты являются логическими символами трех видов:

- логический оператор «И»;
- логический оператор «ИЛИ»;
- логический оператор - исключающее «ИЛИ».

Виды объектов нотации IDEF3 и их назначение представлены в табл. 2.

Таблица 2. Виды объектов нотации IDEF3 и их назначение

1	Модель работы (UOW)	Объект служит для описания функций (процедур, работ), выполняемых подразделениями/сотрудниками предприятия	
2	Объект ссылки (Referent)	Объект, используемый для описания ссылок на другие диаграммы модели, циклические переходы в рамках одной модели, различные комментарии к функциям и перекресткам	
3	Логический оператор «И»	Оператор, позволяющий описать ветвление и слияние процесса. Оператор показывает, что после выполнения функции начинается выполнение всех последующих функций	
4	Логический оператор «ИЛИ»	Оператор, позволяющий описать ветвление и слияние процесса. Оператор показывает, что после выполнения функции начинается выполнение какой-то одной или всех последующих функций	
5	Логический оператор исключающее «ИЛИ»	Оператор, позволяющий описать ветвление и слияние процесса. Оператор показывает, что после выполнения функции начинает выполняться только одна из всех последующих функций	
6	Стрелка предшествования	Соединяет последовательно выполняемые функции	
7	Стрелка отношения	Используется для привязки объектов-комментариев к функциям	
8	Стрелка потока объектов	Показывает поток объектов от одной функции к другой	

Моделирование процессов в нотации DFD.

Одним из важнейших способов описания процесса являются диаграммы потоков данных (информации) DFD (Data Flow Diagram). Подобные диаграммы содержат, как правило, два типа графических объектов: четырехугольники и стрелки. Первые описывают функции (работы, процессы), вторые - потоки данных между этими функциями. Простейшая схема процесса в формате DFD показана на рис. 15.

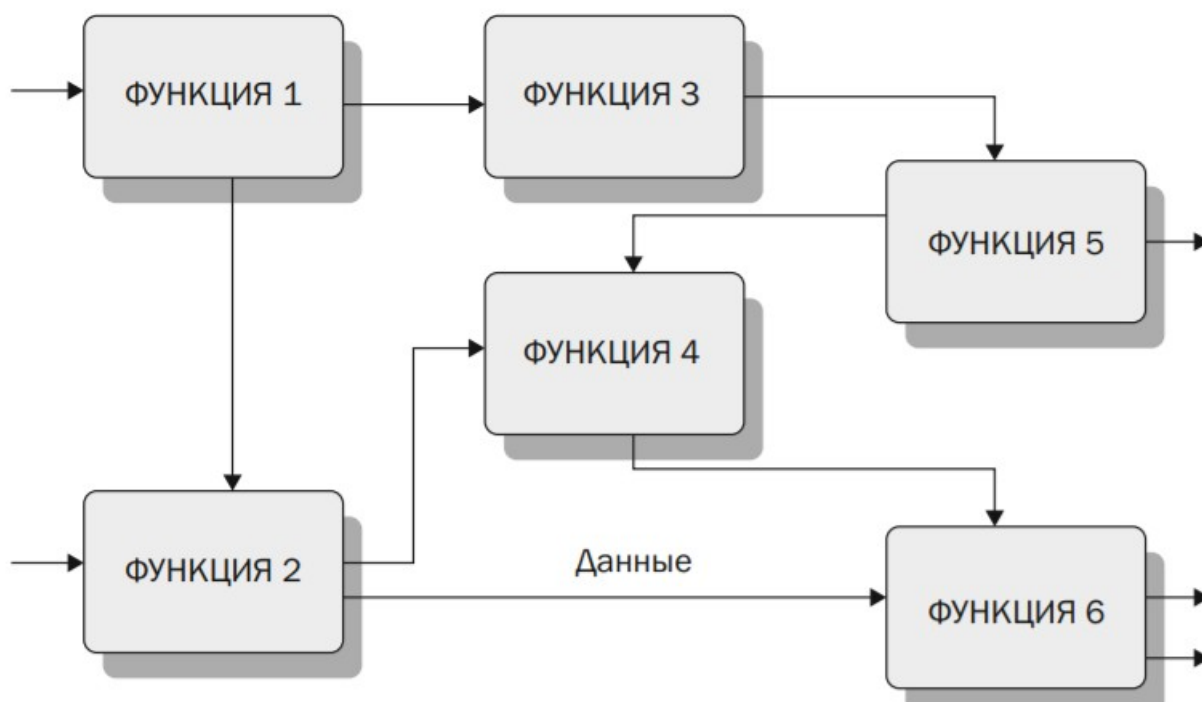


Рис. 15. Пример простейшей модели потоков данных

На диаграмме DFD функции обычно располагаются слева направо в порядке, соответствующем последовательности их выполнения во времени, хотя это не является обязательным. Если придерживаться указанного требования, то полученная схема - это описание процесса, которое схоже с описанием процесса в нотации IDEF3. Процесс, представленный на рис. 15, имеет два входящих и три исходящих потока данных. На верхнем уровне рассмотрения этот процесс выглядел бы в виде одной функции с двумя входами и тремя выходами. Таким образом, к описанию процессов в DFD применимы типовые правила декомпозиции. Что касается сторон четырехугольников, то в нотации DFD они не имеют того значения, как в IDEF0. Следует отметить, что существует несколько подходов к формированию моделей потоков данных.

Часто нотацию DFD путают с простым описанием потоков информации между подразделениями. Это далеко не одно и то же. На рис. 16 представлена модель, отражающая потоки данных между подразделениями, но не являющаяся моделью процесса.

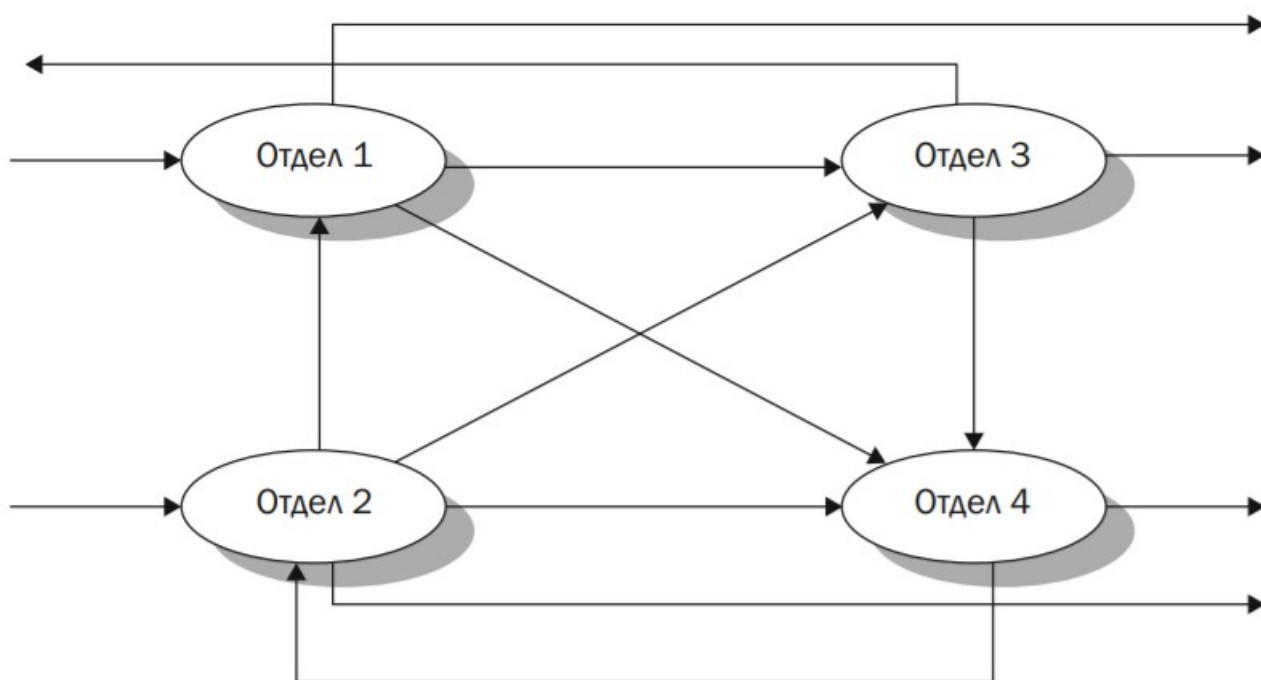


Рис. 16. Пример модели потоков данных между подразделениями организации

В чем здесь дело? Почему нельзя рассматривать простое описание потоков между подразделениями организации как схему процесса? Ответ достаточно прост. В каждом большом подразделении (например, отдел сбыта крупного предприятия) выполняются различные бизнес-процессы. Часто у этих процессов существуют различные внутренние и внешние клиенты. Именно поэтому схема, приведенная на рис. 16, описывает только потоки данных, пересекающие границы подразделений, но не содержит информации о реально выполняемых бизнес-процессах как на уровне подразделений, так и на уровне организации в целом. Кстати, рассмотренный на рис. 16 формат представления потоков данных является практически важным и широко используемым.

Пример описания процесса в DFD можно усложнить, используя понятие «хранилище данных». Под этим понимается любой носитель информации, например, бумажный документ, электронный файл, промышленная база данных на сервере организации и т.д. При построении модели процесса с использованием хранилищ данных, необходимо помнить, что данные (информация) не могут перемещаться между функциями процесса сами по себе. Их можно передавать только через определенных посредников - носителей информации или, что то же самое, хранилищ данных. На рис. 17 представлена модель процесса в нотации DFD, построенная с использованием понятия «хранилище данных».

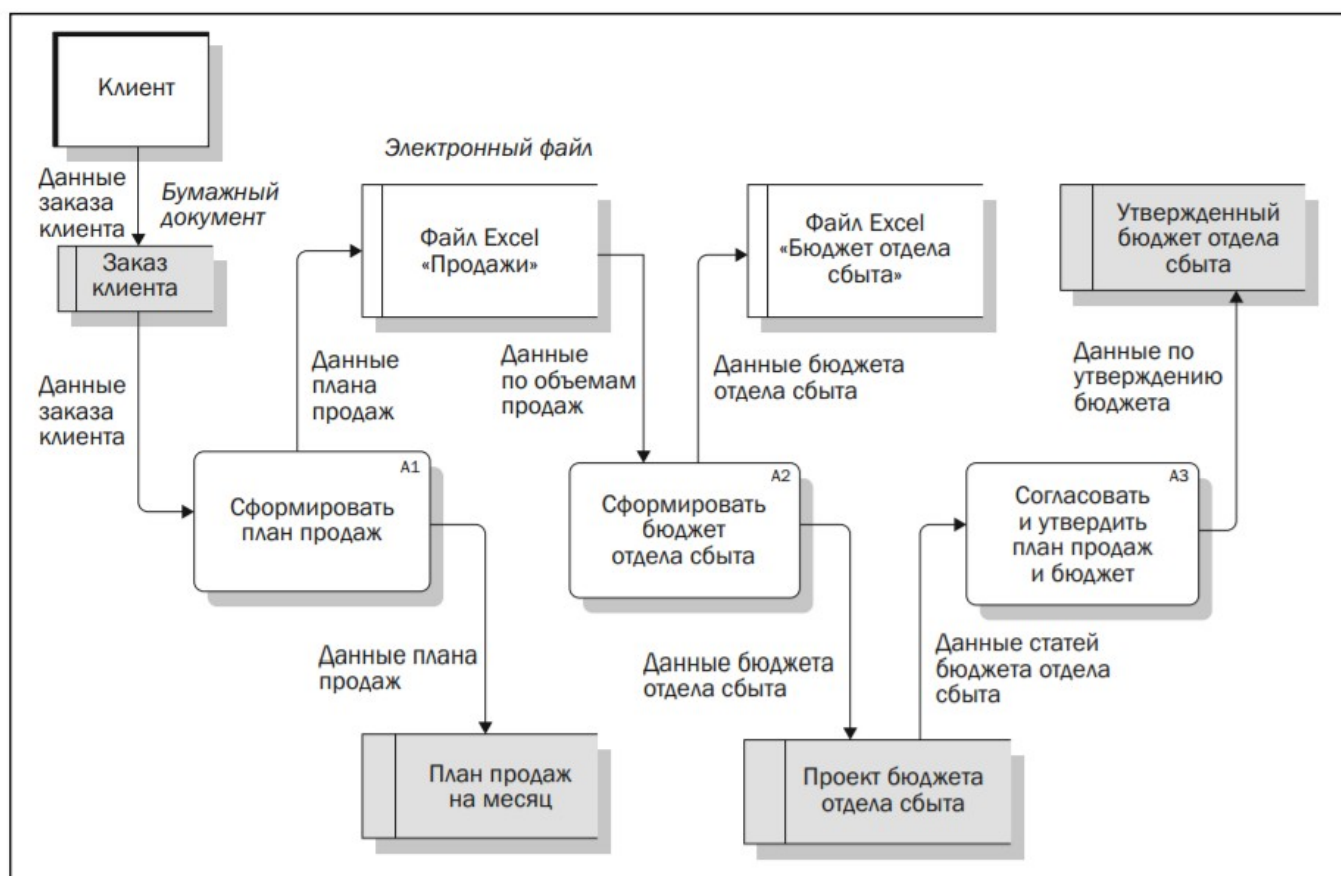


Рис. 17. Модель процесса в нотации DFD

Для чего служат нотации DFD? В первую очередь они нужны для описания реально существующих в организации потоков данных. Описания могут создаваться как по процессному, так и по функциональному признаку. В первом случае мы получаем модели бизнес-процессов в формате DFD, во втором - схему обмена данными между подразделениями. Созданные модели потоков Данных организации могут быть использованы при решении таких задач, как:

- определение существующих хранилищ данных (текстовые документы, файлы, Система управления базой данных - СУБД);
- определение и анализ данных, необходимых для выполнения каждой функции процесса;
- подготовка к созданию модели структуры данных организации, так называемая ERD-модель (IDEF1X);
- выделение основных и вспомогательных бизнес-процессов организации. Следует отметить что нотация DFD может быть эффективно применена для описания потоков документов или потоков материальных ресурсов. На рис. 18 показан пример применения нотации DFD для этих целей.

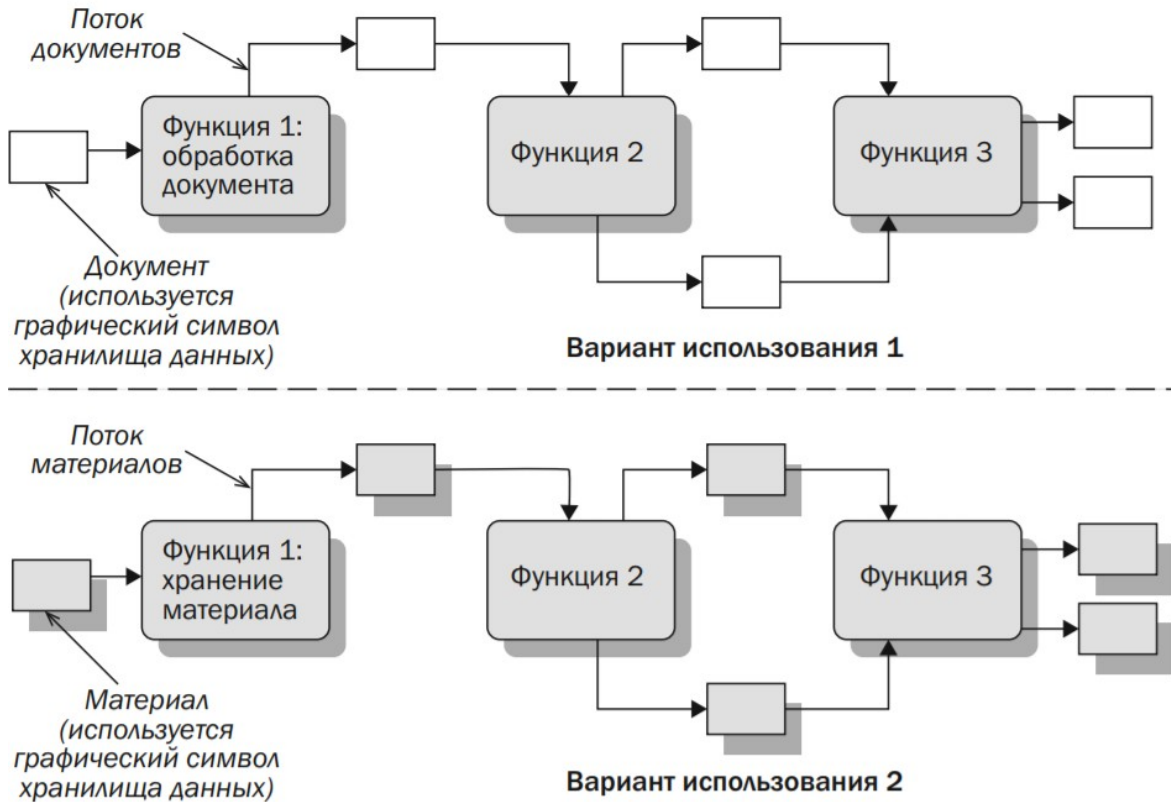


Рис. 18. Описание потоков документов (Вариант 1) или потоков материальных ресурсов (Вариант 2).

Более того, нотация DFD может быть несколько модернизирована таким образом, чтобы на одной диаграмме можно было бы показать как потоки данных, так и потоки материальных ресурсов (рис. 19).

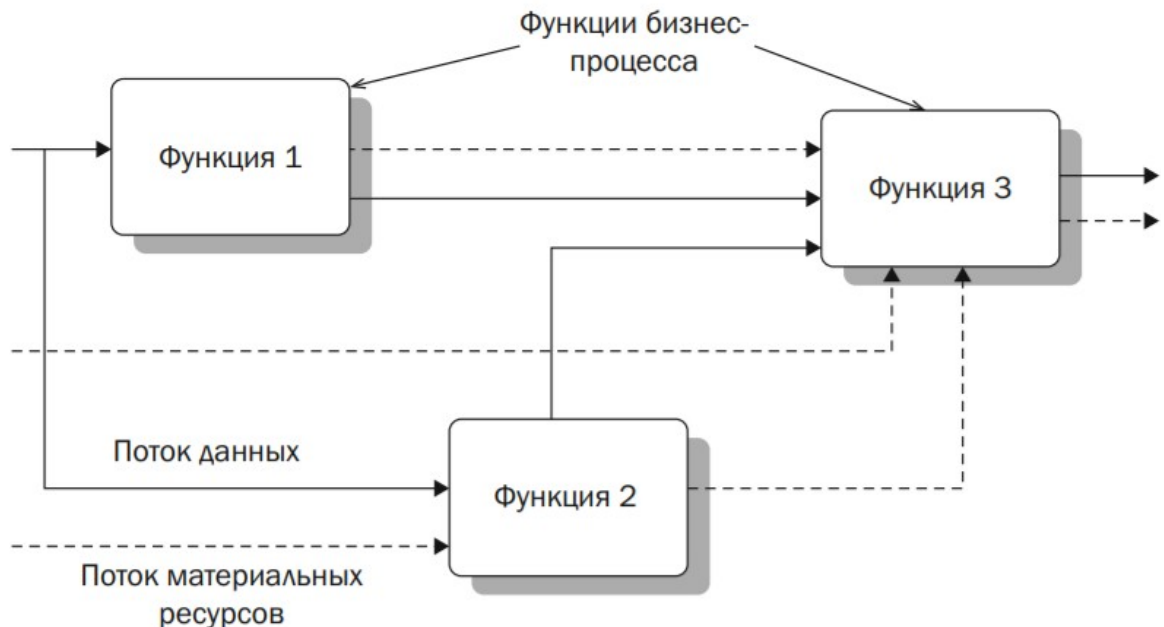


Рис. 19. Совмещение различных типов стрелок на одной модели DFD.¹

¹ Репин В.В., Елиферов В.Г. Процессный подход к управлению. Моделирование бизнес-процессов. - М.: РИА «Стандарты и качество», 2004. - 408 с.

На практике при создании моделей процессов часто бывает полезно использовать несколько способов описания. Сначала, например, мы создаем модель в нотации IDEF0, выявляем функции, входящие в процесс. Затем проводим декомпозицию процесса. При достижении некоторого уровня детализации (3-4) становится целесообразно сформировать для каждого детального процесса несколько схем в различных форматах: управление - IDEF0, а потоки данные и материалов - в DFD.