

## Тема: МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ IDEF0: РУКОВОДЯЩИЙ ДОКУМЕНТ

Введение.

Постоянное усложнение производственно-технических и организационно-экономических систем – фирм, предприятий, производств, и др. субъектов производственно-хозяйственной деятельности - и необходимость их анализа с целью совершенствования функционирования и повышения эффективности обуславливают необходимость применения специальных средств описания и анализа таких систем. Эта проблема приобретает особую актуальность в связи с появлением интегрированных компьютеризированных производств и автоматизированных предприятий.

В США это обстоятельство было осознано еще в конце 70-ых годов, когда ВВС США предложили и реализовали Программу интегрированной компьютеризации производства ICAM (ICAM - Integrated Computer Aided Manufacturing), направленную на увеличение эффективности промышленных предприятий посредством широкого внедрения компьютерных (информационных) технологий.

Реализация программы ICAM потребовала создания адекватных методов анализа и проектирования производственных систем и способов обмена информацией между специалистами, занимающимися такими проблемами. Для удовлетворения этой потребности в рамках программы ICAM была разработана методология IDEF (ICAM Definition), позволяющая исследовать структуру, параметры и характеристики производственно-технических и организационно-экономических систем (в дальнейшем, там, где это не вызывает недоразумений – систем). Общая методология IDEF состоит из трех частных методологий моделирования, основанных на графическом представлении систем:

- IDEF0 используется для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающие эти функции.
- IDEF1 применяется для построения информационной модели, отображающей структуру и содержание информационных потоков, необходимых для поддержки функций системы;
- IDEF2 позволяет построить динамическую модель меняющихся во времени поведения функций, информации и ресурсов системы.

К настоящему времени наибольшее распространение и применение имеют методологии IDEF0 и IDEF1 (IDEF1X), получившие в США статус федеральных стандартов. [1,2]. Методология IDEF0 основана на подходе, разработанном Дугласом Т. Россом в начале 70-ых годов и получившем название SADT (Structured Analysis & Design Technique - метод структурного анализа и проектирования). Основу подхода и, как следствие, методологии IDEF0, составляет графический язык описания (моделирования) систем, обладающий следующими свойствами.

- Графический язык - полное и выразительное средство, способное наглядно представлять широкий спектр деловых, производственных и других процессов и операций предприятия на любом уровне детализации.
- Язык обеспечивает точное и лаконичное описание моделируемых объектов, удобство использования и интерпретации этого описания.
- Язык облегчает взаимодействие и взаимопонимание системных аналитиков, разработчиков и персонала изучаемого объекта (фирмы, предприятия), т.е. служит средством «информационного общения» большого числа специалистов и рабочих групп, занятых в одном проекте, в процессе обсуждения, рецензирования, критики и утверждения результатов.
- Язык прошел многолетнюю проверку и продемонстрировал работоспособность как в проектах ВВС США, так и в других проектах, выполнявшихся государственными и частными промышленными компаниями.

- Язык легок и прост в изучении и освоении.
- Язык может генерироваться рядом инструментальных средств машинной графики; известны коммерческие программные продукты, поддерживающие разработку и анализ моделей - диаграмм IDEF0, например, продукт Design/IDEF 3.7 (и более поздние версии) фирмы Meta Software Corporation (США).

Перечисленные свойства языка предопределили выбор методологии IDEF0 в качестве базового средства анализа и синтеза производственно-технических и организационно-экономических систем, что нашло свое отражение в упомянутых федеральных стандартах США.

В связи с расширяющимся применением информационных технологий и, в частности, CALS-технологий в народном хозяйстве Российской Федерации в настоящем РД приводятся основные сведения о методологии IDEF0 и графическом языке описания моделей, а также некоторые практические рекомендации по разработке таких моделей.

## 2. Концепция IDEF0

Методология IDEF0 основана на следующих концептуальных положениях.

2.1 Модель – искусственный объект, представляющий собой отображение (образ) системы и ее компонентов. Согласно [3],

**М** моделирует **А**, если **М** отвечает на вопросы относительно **А**.

Здесь **М** – модель, **А** – моделируемый объект (оригинал).

Модель разрабатывают для понимания, анализа и принятия решений о реконструкции (реинжиниринге) или замене существующей, либо проектировании новой системы. Система представляет собой совокупность взаимосвязанных и взаимодействующих частей, выполняющих некоторую полезную работу. Частями (элементами) системы могут быть любые комбинации разнообразных сущностей, включающие людей, информацию, программное обеспечение, оборудование, изделия, сырье или энергию (энергоносители). Модель описывает, что происходит в системе, как ею управляют, какие сущности она преобразует, какие средства использует для выполнения своих функций и что производит.

2.2 Блочное моделирование и его графическое представление. Основной концептуальный принцип методологии IDEF – представление любой изучаемой системы в виде набора взаимодействующих и взаимосвязанных блоков, отображающих процессы, операции, действия (определения – см. ниже), происходящие в изучаемой системе. В IDEF0 все, что происходит в системе и ее элементах, принято называть функциями. Каждой функции ставится в соответствие блок. На IDEF0 – диаграмме, основном документе при анализе и проектировании систем, блок представляет собой прямоугольник. Интерфейсы, посредством которых блок взаимодействует с другими блоками или с внешней по отношению к моделируемой системе средой, представляются стрелками), входящими в блок или выходящими из него. Входящие стрелки показывают, какие условия должны быть одновременно выполнены, чтобы функция, описываемая блоком, осуществилась.

2.3 Лаконичность и точность. Документация, описывающая систему, должна быть точной и лаконичной. Многословные характеристики, изложенные в форме традиционных текстов, неудовлетворительны. Графический язык позволяет лаконично, однозначно и точно показать все элементы (блоки) системы и все отношения и связи между ними, выявить ошибочные, лишние или дублирующие связи и т.д..

2.4 Передача информации. Средства IDEF0 облегчают передачу информации от одного участника разработки модели (отдельного разработчика или рабочей группы) к другому. К числу таких средств относятся:

- диаграммы, основанные на простой графике блоков и стрелок, легко читаемые и понимаемые;
- метки на естественном языке для описания блоков и стрелок, а также глоссарий и сопроводительный текст для уточнения смысла элементов диаграммы;
- последовательная декомпозиция диаграмм, строящаяся по иерархическому принципу, при котором на верхнем уровне отображаются основные функции, а затем происходит их детализация и уточнение;
- древовидные схемы иерархии диаграмм и блоков, обеспечивающие обзорность модели в целом и входящих в нее деталей.

2.5 Строгость и формализм. Разработка моделей IDEF0 требует соблюдения ряда строгих формальных правил, обеспечивающих преимущества методологии в отношении однозначности, точности и целостности сложных многоуровневых моделей. Эти правила описываются ниже. Здесь отмечается только основное из них: все стадии и этапы разработки и корректировки модели должны строго, формально документироваться с тем, чтобы при ее эксплуатации не возникало вопросов, связанных с неполнотой или некорректностью документации.

2.6 Итеративное моделирование. Разработка модели в IDEF0 представляет собой пошаговую, итеративную процедуру. На каждом шаге итерации разработчик предлагает вариант модели, который подвергают обсуждению, рецензированию и последующему редактированию, после чего цикл повторяется. Такая организация работы способствует оптимальному использованию знаний системного аналитика, владеющего методологией и техникой IDEF0, и знаний специалистов – экспертов в предметной области, к которой относится объект моделирования.

2.7 Отделение «организации» от «функций». При разработке моделей следует избегать изначальной «привязки» функций исследуемой системы к существующей организационной структуре моделируемого объекта (предприятия, фирмы). Это помогает избежать субъективной точки зрения, навязанной организацией и ее руководством. Организационная структура должна явиться результатом использования (применения) модели. Сравнение результата с существующей структурой позволяет, во-первых, оценить адекватность модели, а во-вторых – предложить решения, направленные на совершенствование этой структуры.

### 3. Основные определения (понятия) методологии и языка IDEF0.

3.1 Блок: прямоугольник, содержащий имя и номер и используемый для описания функции.

3.2 Ветвление: разделение стрелки на два или большее число сегментов. Может означать «развязывание пучка» (см. 3.27).

3.3 Внутренняя стрелка: входная, управляющая или выходная стрелка, концы которой связывают источник и потребителя, являющиеся блоками одной диаграммы. Отличается от граничной стрелки.

3.4 Входная стрелка: класс стрелок, которые отображают вход IDEF0-блока, то есть данные или материальные объекты, которые преобразуются функцией в выход. Входные стрелки связываются с левой стороной блока IDEF0.

3.5 Выходная стрелка: класс стрелок, которые отображают выход IDEF0- блока, то есть данные или материальные объекты, произведенные функцией. Выходные стрелки связываются с правой стороной блока IDEF0.

3.6 Глоссарий: список определений для ключевых слов, фраз и аббревиатур, связанных с узлами, блоками, стрелками или с моделью IDEF0 в целом.

3.7 Граничная стрелка: стрелка, один из концов которой связан с источником или потребителем, а другой не присоединен ни к какому блоку на диаграмме. Отображает связь диаграммы с другими блоками системы и отличается от внутренней стрелки.

3.8 Декомпозиция: разделение моделируемой функции на функции - компоненты.

3.9 Дерево узлов: представление отношений между родительскими и дочерними узлами модели IDEF0 в форме древовидного графа. Имеет то же значение и содержание, что и перечень узлов (см. 3.23).

3.10 Диаграмма А-0: специальный вид (контекстной) диаграммы IDEF0, состоящей из одного блока, описывающего функцию верхнего уровня, ее входы, выходы, управления, и механизмы, вместе с формулировками цели модели и точки зрения, с которой строится модель.

3.11 Диаграмма: часть модели, описывающая декомпозицию блока.

3.12 Диаграмма-иллюстрация (FEO): графическое описание, используемое, для общения специфических фактов о диаграмме IDEF0. При построении диаграмм FEO можно не придерживаться правила IDEF0.

3.13 Дочерний блок: блок на дочерней (порожденной) диаграмме.

3.14 Дочерняя диаграмма: диаграмма, детализирующая родительский (порождающий) блок.

3.15 Имя блока: глагол или глагольный оборот, помещенный внутри блока и описывающий моделируемую функцию.

3.16 Интерфейс: разделяющая граница, через которую проходят данные или материальные объекты; соединение между двумя или большим числом компонентов модели, передающее данные или материальные объекты от одного компонента к другому.

3.17 Код ICOM: аббревиатура( Input - Вход, Control - Управление, Output - Выход, Mechanism – Механизм), код, обеспечивающий соответствие граничных стрелок дочерней диаграммы со стрелками родительского блока; используется для ссылок.

3.18 Контекст: окружающая среда, в которой действует функция (или комплект функций на диаграмме).

3.19 Контекстная диаграмма: диаграмма, имеющая узловой номер А-п ( $p \geq 0$ ), которая представляет контекст модели, Диаграмма А-0, состоящая из одного блока, является необходимой (обязательной) контекстной диаграммой; диаграммы с узловыми номерами А-1, А-2,... - дополнительные контекстные диаграммы.

3.20 Метка стрелки: существительное или оборот существительного, связанные со стрелкой или сегментом стрелки и определяющие их значение.

3.21 Модель IDEF0: графическое описание системы, разработанное с определенной целью (см. 3.46 ) и с выбранной точки зрения (см. 3.39 ). Комплект одной или более диаграмм IDEF0, которые изображают функции системы с помощью графики, текста и глоссария.

3.22 Номер блока: число (0 - 6), помещаемое в правом нижнем углу блока и однозначно идентифицирующее блок на диаграмме.

3.23 Перечень узлов: список, часто ступенчатый, показывающий узлы модели IDEF0 в упорядоченном виде. Имеет то же значение и содержание, что и дерево узлов (см. 3.9 ).

3.24 Примечание к модели: текстовый комментарий, являющийся частью диаграммы IDEF0 и используемый для записи факта, не нашедшего графического изображения.

3.25 Родительская диаграмма: диаграмма, которая содержит родительский блок.

3.26 Родительский блок: блок, который подробно описывается дочерней диаграммой.

3.27 Связывание/развязывание: объединение значений стрелок в составное значение (связывание в «пучок»), или разделение значений стрелок (развязывание «пучка»), выраженные синтаксисом слияния или ветвления стрелок.

3.28 Сегмент стрелки: сегмент линии, который начинается или заканчивается на стороне блока, в точке ветвления или слияния, или на границе (несвязанный конец стрелки).

3.29 Семантика: значение синтаксических компонентов языка.

3.30 Синтаксис: Структурные компоненты или характеристики языка и правила, которые определяют отношения между ними.

3.31 Слияние: объединение двух или большего числа сегментов стрелок в один сегмент. Может означать «развязывание пучка» (см. 3.27 )

3.32 С-номер: номер, создаваемый в хронологическом порядке и используемый для идентификации диаграммы и прослеживания ее истории; может быть использован в качестве ссылочного выражения при определении конкретной версии диаграммы.

3.33 Стрелка: направленная линия, состоящая из одного или нескольких сегментов, которая моделирует открытый канал или канал, передающий данные или материальные объекты от источника (начальная точка стрелки), к потребителю (конечная точка с «наконечником»). Имеется 4 класса стрелок: входная стрелка, выходная стрелка, управляющая стрелка, стрелка механизма (включает стрелку вызова). (См.: сегмент стрелки, граничная стрелка, внутренняя стрелка).

3.34 Стрелка вызова: вид стрелки механизма, который обозначает обращение из блока данной модели (или части модели) к блоку другой модели (или другой части той же модели) и обеспечивает связь между моделями или между разными частями одной модели.

3.35 Стрелка механизма: класс стрелок, которые отображают механизмы IDEF0, то есть средства, используемые для выполнения функции; включает специальный случай стрелки вызова. Стрелки механизмов связываются с нижней стороной блока IDEF0.

3.36 Стрелка, помещенная в туннель (туннельная стрелка): стрелка (со специальной нотацией), не удовлетворяющая обычному требованию, согласно которому каждая стрелка на дочерней диаграмме должна соответствовать стрелкам на родительской диаграмме.

3.37 Текст: любой текстовый (не графический) комментарий к графической диаграмме IDEF0.

3.38 Тильда: небольшая ломаная (волнистая) линия, используемая для соединения метки с конкретным сегментом стрелки или примечания модели с компонентом диаграммы.

3.39 Точка зрения: указание на должностное лицо или подразделение организации, с позиции которого разрабатывается модель

3.40 Узел: блок, порождающий дочерние блоки; родительский блок. (См.: перечень узлов, дерево узлов, узловой номер, узловая ссылка, номер узла диаграммы).

3.41 Узловая ссылка: код, присвоенный диаграмме, для ее идентификации и определения положения в иерархии модели; формируется из сокращенного имени модели и узлового номера диаграммы с дополнительными расширениями.

3.42 Узловой номер диаграммы: часть узловой ссылки диаграммы, которая соответствует номеру родительского блока.

3.43 Узловой номер: код, присвоенный блоку и определяющий его положение в иерархии модели; может быть использован в качестве подробного ссылочного выражения.

3.44 Управляющая стрелка: класс стрелок, которые в IDEF0 отображают управления, то есть условия, при выполнении которых выход блока будет правильным. Данные или объекты, моделируемые как управления, могут преобразовываться функцией, создающей соответствующий выход. Управляющие стрелки связываются с верхней стороной блока IDEF0.

3.45 Функция: деятельность, процесс или преобразование (моделируемые блоком IDEF0), идентифицируемое глаголом или глагольной формой, которая описывает, что должно быть выполнено.

3.46 Цель: краткая формулировка причины создания модели.

#### 4. Синтаксис графического языка IDEF0.

Набор структурных компонентов языка, их характеристики и правила, определяющие связи между компонентами, представляют собой синтаксис языка. Компоненты синтаксиса IDEF0 – блоки, стрелки, диаграммы и правила. Блоки представляют функции, определяемые как деятельность, процесс, операция, действие или преобразование (см. ниже). Стрелки представляют данные или материальные объекты, связанные с функциями. Правила определяют, как следует применять компоненты; диаграммы обеспечивают формат графического и словесного описания моделей. Формат образует основу для управления конфигурацией модели.

#### 4.1 Блок.

Блок описывает функцию. Типичный блок показан на рис. 1. Внутри каждого блока помещается его имя и номер. Имя должно быть активным глаголом или глагольным оборотом, описывающим функцию. Номер блока размещается в правом нижнем углу. Номера блоков используются для их идентификации на диаграмме и в соответствующем тексте.

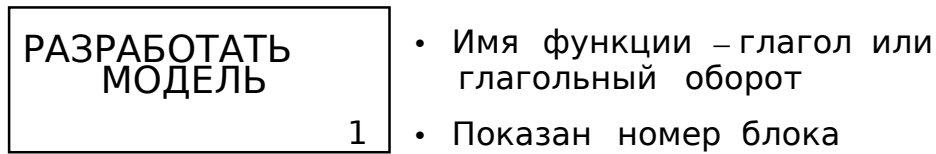


Рис.1. Типичный блок. Содержит имя и номер.

#### 4.2. Стрелка.

Стрелка формируется из одного или более отрезков прямых и наконечника на одном конце. Как показано на рис. 2, сегменты стрелок могут быть прямыми или ломаными; в последнем случае горизонтальные и вертикальные отрезки стрелки сопрягаются дугами, имеющими угол  $90^\circ$ . Стрелки не представляют поток или последовательность событий, как в традиционных блок-схемах потоков или процессов. Они лишь показывают, какие данные или материальные объекты должны поступить на вход функции для того, чтобы эта функция могла выполняться.

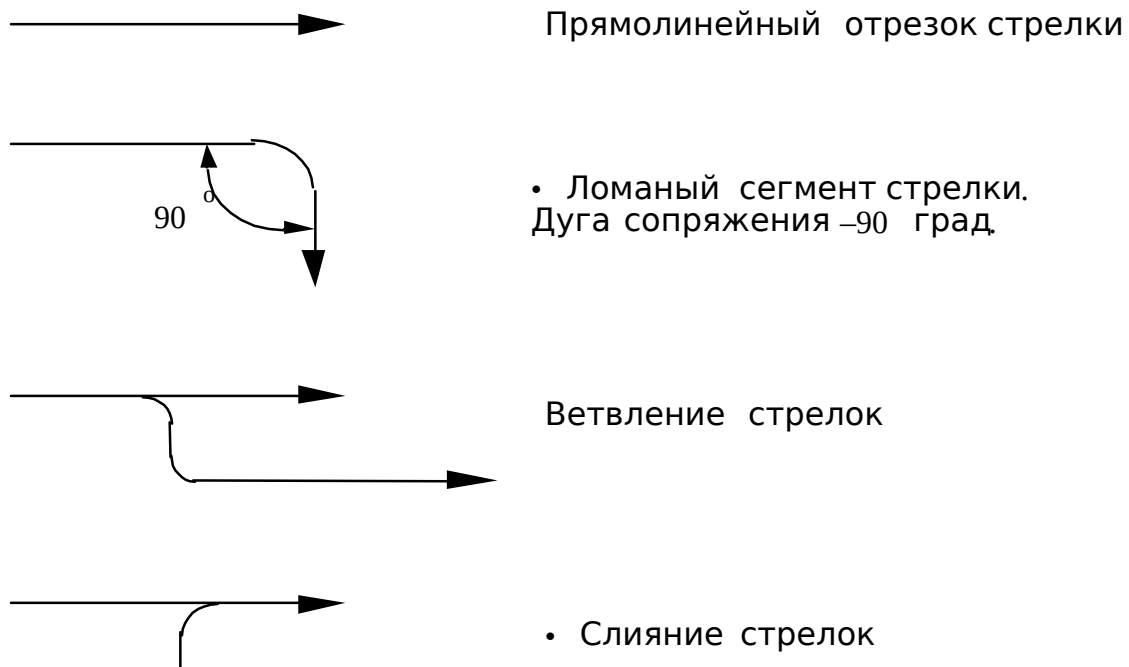


Рис. 2. Синтаксис стрелок.

#### 4.3 Синтаксические правила.

##### 4.3.1 Блоки

1. Размеры блоков должны быть достаточными для того, чтобы включить имя блока.
2. Блоки должны быть прямоугольными, с прямыми углами.
3. Блоки должны быть нарисованы сплошными линиями.

### 4.3.2 Стрелки

1. Ломаные стрелки изменяют направление только под углом 90 град.
2. Стрелки должны быть нарисованы сплошными линиями различной толщины.
3. Стрелки могут состоять только из вертикальных или горизонтальных отрезков; отрезки, направленные по диагонали, не допускаются.
4. Концы стрелок должны касаться внешней границы функционального блока, но не должны пересекать ее.
5. Стрелки должны присоединяться к блоку на его сторонах. Присоединение в углах не допускается.

### 5. Семантика языка IDEF0.

Семантика определяет содержание (значение) синтаксических компонентов языка и способствует правильности их интерпретации. Интерпретация устанавливает соответствие между блоками и стрелками с одной стороны и функциями и их интерфейсами – с другой.

### 5.1 Семантика блоков и стрелок

Поскольку IDEF0 есть методология функционального моделирования, имя блока, описывающее функцию, должно быть глаголом или глагольным оборотом; например, имя блока "Выполнить проверку", означает, что блок с таким именем превращает непроверенные детали в проверенные. После присваивания блоку имени, к соответствующим его сторонам присоединяются входные, выходные и управляющие стрелки, а также стрелки механизма, что и определяет наглядность и выразительность изображения блока IDEF0.

Чтобы гарантировать точность модели, следует использовать стандартную терминологию. Блоки именуются глаголами или глагольными оборотами и эти имена сохраняются при декомпозиции. Стрелки и их сегменты, как отдельные, так и связанные в «пучок», помечаются существительными или оборотами существительного. Метки сегментов позволяют конкретизировать данные или материальные объекты, передаваемые этими сегментами, с соблюдением синтаксиса ветвлений и слияний.

Каждая сторона функционального блока имеет стандартное значение с точки зрения связи блок/стрелки. В свою очередь, сторона блока, к которой присоединена стрелка, однозначно определяет ее роль. Стрелки, входящие в левую сторону блока - входы. Входы преобразуются или расходуются функцией, чтобы создать то, что появится на ее выходе. Стрелки, входящие в блок сверху - управления. Управления определяют условия, необходимые функции, чтобы произвести правильный выход. Стрелки, покидающие блок справа – выходы, т.е. данные или материальные объекты, произведенные функцией.

Стрелки, подключенные к нижней стороне блока, представляют механизмы. Стрелки, направленные вверх, идентифицируют средства, поддерживающие выполнение функции. Другие средства могут наследоваться из родительского блока. Стрелки механизма, направленные вниз, являются стрелками вызова. Стрелки вызова обозначают обращение из данной модели или из данной части модели к блоку, входящему в состав другой модели или другой части модели, обеспечивая их связь, т.е. разные модели или разные части одной и той же модели могут совместно использовать один и тот же элемент (блок). Стандартное расположение стрелок показано на рис. 3.

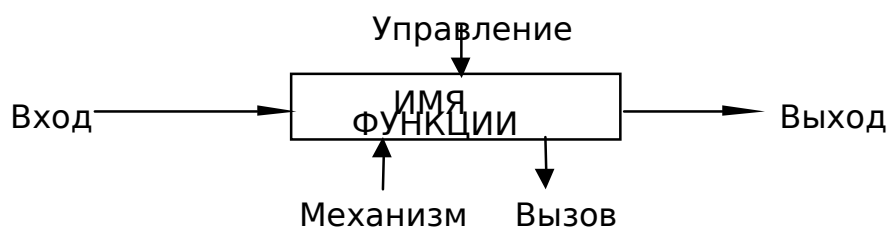


Рис. 3. Стандартное расположение стрелок

## 5.2 Имена и метки.

Как указывалось, имена функций – глаголы или глагольные обороты. Примеры таких имен:

производить детали	планировать ресурсы	наблюдать
наблюдать за выполнением	проектировать систему	эксплуатировать
разработать детальные чертежи	изготовить компонент	проверить деталь

Стрелки идентифицируют данные или материальные объекты, необходимые для выполнения функции или производимые ею. Каждая стрелка должна быть помечена существительным или оборотом существительного, например:

Спецификации	отчет об испытаниях	бюджет
Конструкторские требования	конструкция детали	директива
Инженер-конструктор	плата в сборе	требования

Пример размещения меток стрелок и имени блока показан на рис. 4.

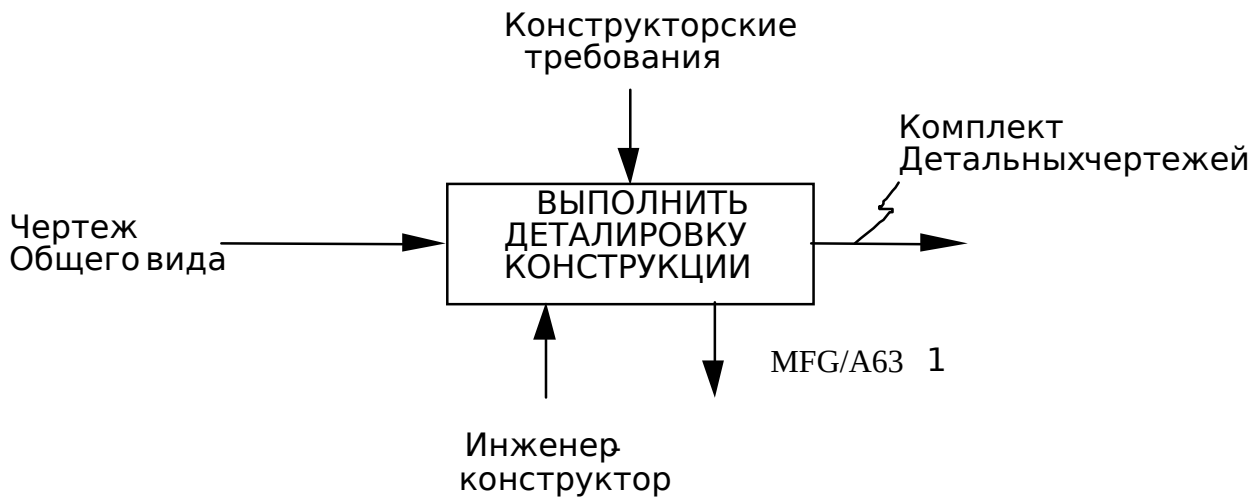


Рис. 4. Пример размещения меток стрелок и имени блока

## 5.3 Семантические правила блоков и стрелок

1. Имя блока должно быть активным глаголом или глагольным оборотом.
2. Каждая сторона функционального блока должна иметь стандартное отношение блок/стрелки:
  - а) входные стрелки должны связываться с левой стороной блока;
  - б) управляющие стрелки должны связываться с верхней стороной блока;
  - в) выходные стрелки должны связываться с правой стороной блока;
  - г) стрелки механизма (кроме стрелок вызова) должны указывать вверх и подключаться к нижней стороне блока.
  - д) стрелки вызова механизма должны указывать вниз, подключаться к нижней стороне блока, и помечаться ссылкой на вызываемый блок.



3. Сегменты стрелок, за исключением стрелок вызова, должны помечаться существительным или оборотом существительного, если только единственная метка стрелки несомненно не относится к стрелке в целом.

4. Чтобы связать стрелку с меткой, следует использовать "тильду" ( ~ ).

5. В метках стрелок не должны использоваться следующие термины: функция, вход, управление, выход, механизм, вызов.

#### 5.4 Диаграммы IDEF0.

IDEF0-модели состоят из трех типов документов: графических диаграмм, текста и глоссария. Эти документы имеют перекрестные ссылки друг на друга. Графическая диаграмма – главный компонент IDEF0-модели, содержащий блоки, стрелки, соединения блоков и стрелок и ассоциированные с ними отношения. Блоки представляют основные функции моделируемого объекта. Эти функции могут быть разбиты (декомпозированы) на составные части и представлены в виде более подробных диаграмм; процесс декомпозиции продолжается до тех пор, пока объект не будет описан на уровне детализации, необходимом для достижения целей конкретного проекта. Диаграмма верхнего уровня обеспечивает наиболее общее или абстрактное описание объекта моделирования. За этой диаграммой следует серия дочерних диаграмм, дающих более детальное представление об объекте.

#### 5.5 Контекстная диаграмма верхнего уровня.

Каждая модель должна иметь контекстную диаграмму верхнего уровня, на которой объект моделирования представлен единственным блоком с граничными стрелками. Эта диаграмма называется А-0 (А минус нуль). Стрелки на этой диаграмме отображают связи объекта моделирования с окружающей средой. Поскольку единственный блок представляет весь объект, его имя – общее для всего проекта. Это же справедливо и для всех стрелок диаграммы, поскольку они представляют полный комплект внешних интерфейсов объекта. Диаграмма А-0 устанавливает область моделирования и ее границу. Пример диаграммы А-0 показан на рис. 5.

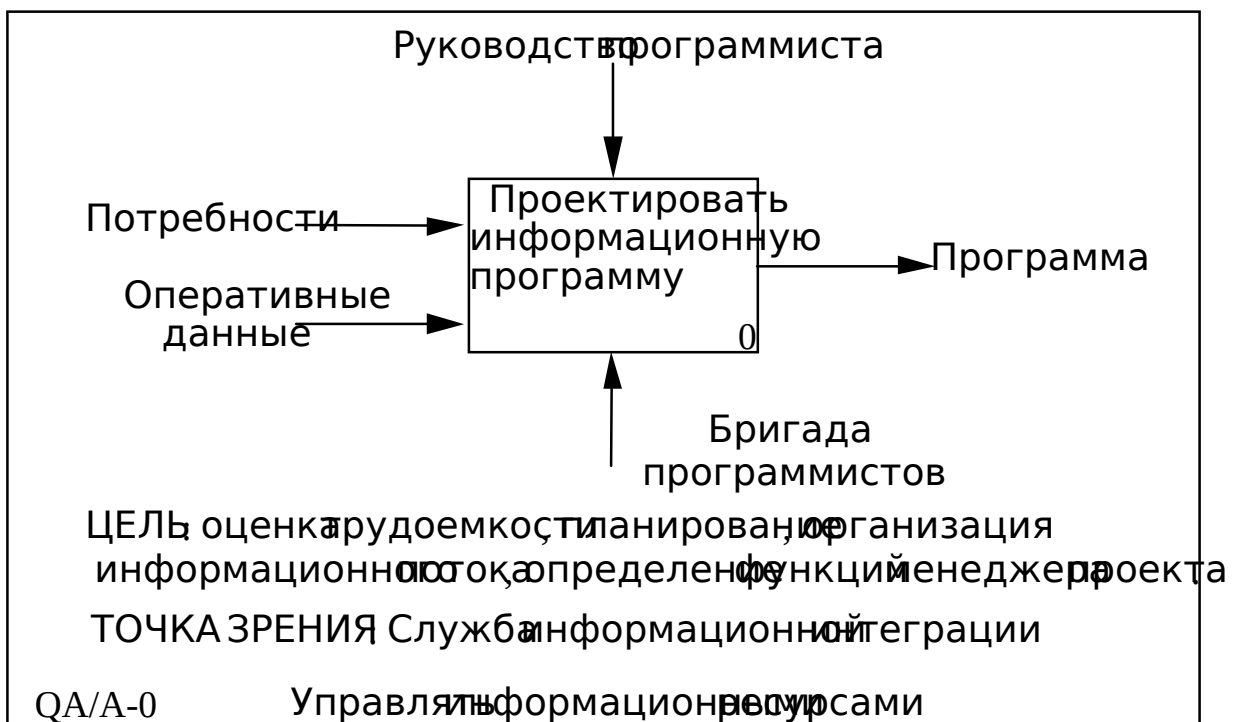


Рис. 5. Пример диаграммы А0

Контекстная диаграмма А-0 также должна содержать краткие утверждения, определяющие точку зрения должностного лица или подразделения, с позиций которого создается модель, и цель, для достижения которой ее разрабатывают. Эти утверждения помогают руководить разработкой модели и ввести этот процесс в определенные рамки. Точка зрения определяет, что и в каком разрезе можно увидеть в пределах контекста модели. Изменение точки зрения, приводит к рассмотрению других аспектов объекта. Аспекты, важные с одной точки зрения, могут не появиться в модели, разрабатываемой с другой точки зрения на тот же самый объект.

Формулировка цели выражает причину создания модели, т.е. содержит перечень вопросов, на которые должна отвечать модель, что в значительной мере определяет ее структуру. Наиболее важные свойства объекта обычно выявляются на верхних уровнях иерархии; по мере декомпозиции функции верхнего уровня и разбиения ее на подфункции, эти свойства уточняются. Каждая подфункция, в свою очередь, декомпозируется на элементы следующего уровня, и так происходит до тех пор, пока не будет получена релевантная структура, позволяющая ответить на вопросы, сформулированные в цели моделирования. Каждая подфункция моделируется отдельным блоком. Каждый родительский блок подробно описывается дочерней диаграммой на более низком уровне. Все дочерние диаграммы должны быть в пределах области контекстной диаграммы верхнего уровня.

#### 5.6 Дочерняя диаграмма.

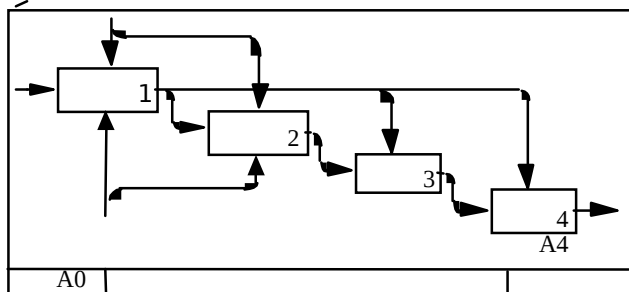
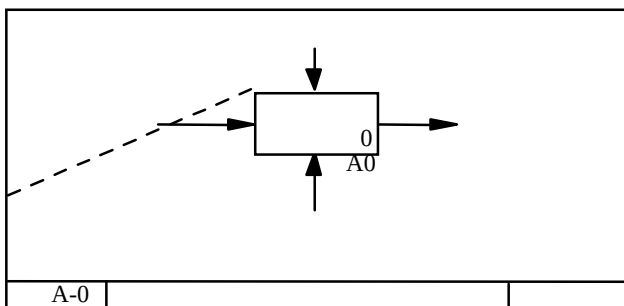
Единственная функция, представленная на контекстной диаграмме верхнего уровня, может быть разложена на основные подфункции посредством создания дочерней диаграммы. В свою очередь, каждая из этих подфункций может быть разложена на составные части посредством создания дочерней диаграммы следующего, более низкого уровня, на которой некоторые или все функции также могут быть разложены на составные части. Каждая дочерняя диаграмма содержит дочерние блоки и стрелки, обеспечивающие дополнительную детализацию родительского блока.

Дочерняя диаграмма, создаваемая при декомпозиции, охватывает ту же область, что и родительский блок, но описывает ее более подробно. Таким образом, дочерняя диаграмма как бы вложена в свой родительский блок. Эта структура иллюстрируется рис. 6.

#### 5.7 Родительская диаграмма

Родительская диаграмма – та, которая содержит один или более родительских блоков. Каждая обычная (не-контекстная) диаграмма является также дочерней диаграммой, поскольку, по определению, она подробно описывает некоторый родительский блок. Таким образом, любая диаграмма может быть как родительской диаграммой (содержать родительские блоки), так и дочерней (подробно описывать собственный родительский блок). Аналогично, блок может быть как родительским (подробно описываться дочерней диаграммой) так и дочерним (появляющимся на дочерней диаграмме). Основное иерархическое отношение существует между родительским блоком и дочерней диаграммой, которая его подробно описывает (рис.6).

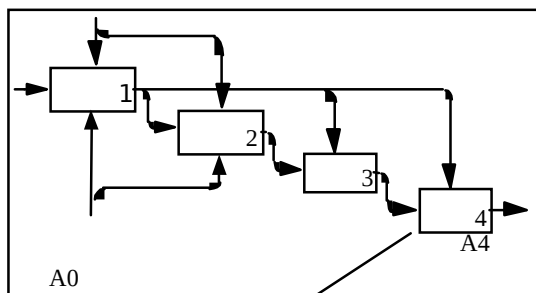
То, что блок является дочерним и раскрывает содержание родительского блока на диаграмме предшествующего уровня, указывается специальным ссылочным кодом, написанным ниже правого нижнего угла блока. Этот ссылочный код может формироваться несколькими способами, из которых самый простой заключается в том, что код, начинающийся с буквы А (по имени диаграммы А-0), содержит цифры, определяемые номерами родительских блоков. Например, показанные на рис. 7 коды означают, что диаграмма является декомпозицией 1-го блока диаграммы, которая, в свою очередь является декомпозицией 6-го блока диаграммы А0, а сами коды образуются присоединением номера блока



Более общее представление

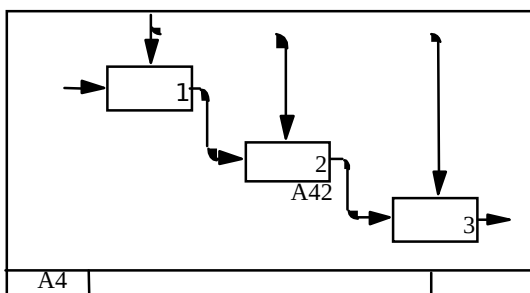


Более детальное

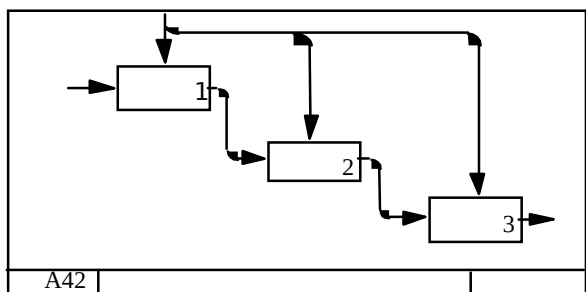
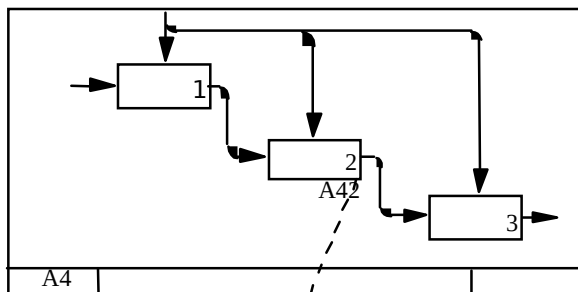


Более детальное

Этот блок родительский для этой диаграммы



Этот блок - родительский для этой диаграммы



ПРИМЕЧАНИЕ : Номер узла показывает что этот блок был декомпозирован С-номер или номер листа дочерней диаграммы может использоваться вместо узлового номера

### 5.8 Текст и глоссарий.

Диаграмме может быть поставлен в соответствие структурированный текст, представляющий собой краткий комментарий к содержанию диаграммы. Текст используется для объяснений и уточнений характеристик, потоков, внутриблочных соединений и т.д. Текст не должен использоваться для описания и без того понятных блоков и стрелок на диаграммах.

Глоссарий предназначен для определения аббревиатур (акронимов), ключевых слов и фраз, используемых в качестве имен и меток на диаграммах. Глоссарий определяет понятия и термины, которые должны быть одинаково понимаемы всеми участниками разработки и пользователями модели, чтобы правильно интерпретировать ее содержание.

5.9 Диаграммы - иллюстрации (FEO). Эти диаграммы используются в качестве дополнений, поясняющих специфику содержания основных диаграмм в тех случаях, когда это необходимо. Диаграмма FEO не должна подчиняться синтаксическим правилам IDEF0.

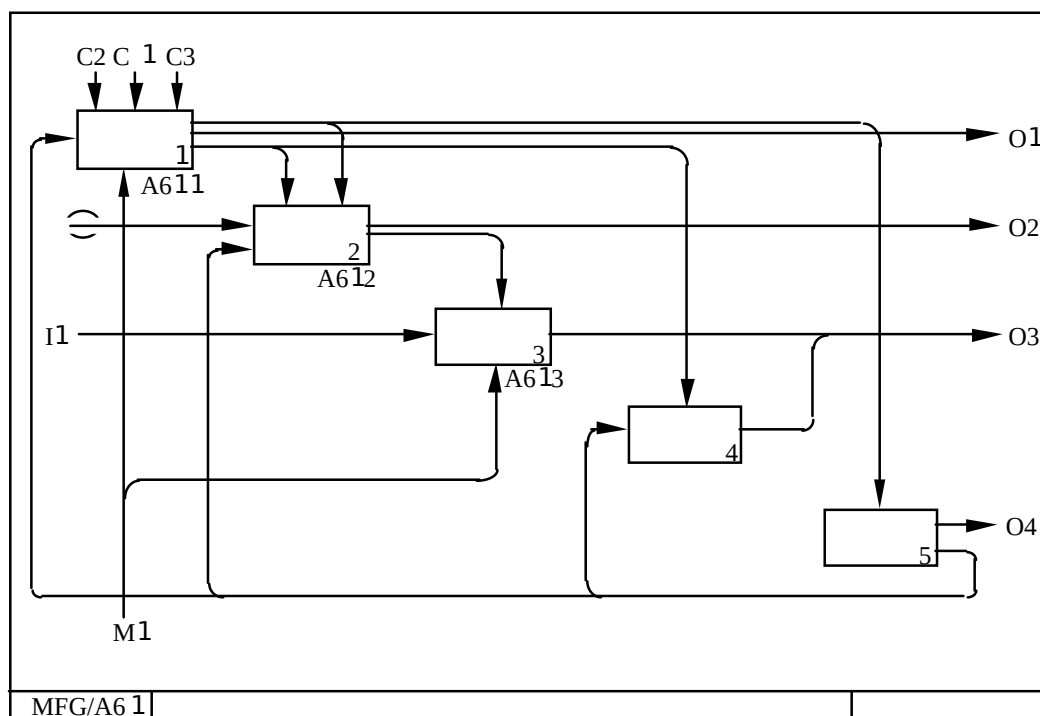
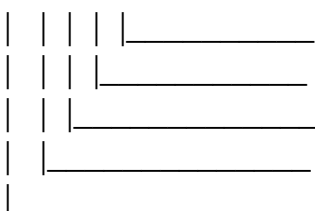


Рис 7. Пример дочернего блока.

A 6 1 \* \* \* \*



и т.д.

Номер блока на диаграмме A 6  
 Номер блока на диаграмме A 6  
 Номер блока на диаграмме A 0  
 Имя блока A 0

P

Составитель: А.Х. Александров  
 07.04.2020 г.