

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»

Институт прикладной математики компьютерных наук
Кафедра «Информационная безопасность»

Утверждено на заседании кафедры
«Информационная безопасность»
«29» января 2019 г., протокол № 6

Заведующий кафедрой

_____ А.А. Сычугов

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению контрольно-курсовой работы
по дисциплине
«Модели и методы анализа проектных решений»

основной профессиональной образовательной программы
высшего образования – программы бакалавриата

по направлениям подготовки
09.03.03 Прикладная информатика

с направленностью (профилем)
Прикладная информатика в промышленности

Формы обучения: заочная

Идентификационный номер образовательной программы: 090303-01-19

Тула 2019 год

Разработчик(и) методических указаний

Волошко А.Г., доцент, к.т.н.

(ФИО, должность, ученая степень, ученое звание)

(подпись)

Методология IDEF0. Разработка структурно-функциональной модели сложной системы

1. Цель и задачи работы

Цель работы – освоить навыки разработки структурно-функциональной модели для сложных систем.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- Освоить принципы построения модели IDEF0.
- Освоить систему VS Visio для построения модели IDEF0
- Разработать структурно-функциональную модель по индивидуальному заданию.

2. Общие положения (теоретические сведения)

Основные элементы и понятия IDEF0

Графический язык IDEF0 удивительно прост и гармоничен. В основе методологии лежат четыре основных понятия:

Первым из них является понятие **функционального блока (Activity Box)**. Функциональный блок графически изображается в виде прямоугольника (см. рис. 1) и олицетворяет собой некоторую конкретную функцию в рамках рассматриваемой системы. По требованиям стандарта 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 (роль), при этом:

- Верхняя сторона имеет значение “Управление” (Control);
- Левая сторона имеет значение “Вход” (Input);
- Правая сторона имеет значение “Выход” (Output);
- Нижняя сторона имеет значение “Механизм” (Mechanism).

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

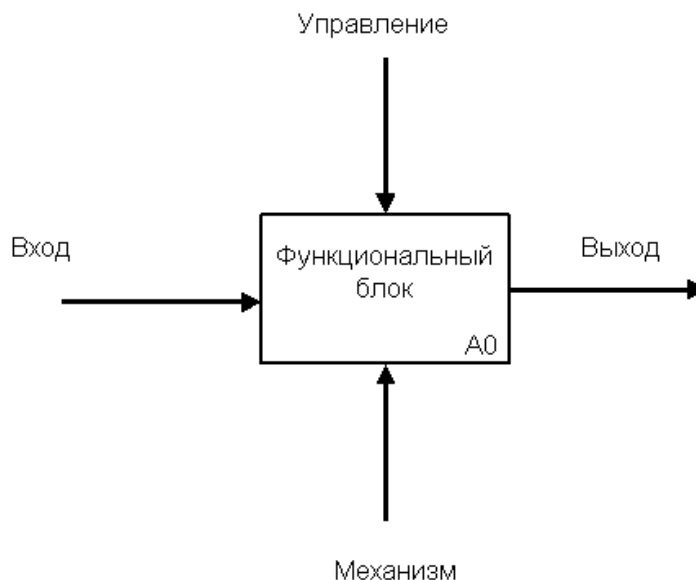


Рисунок 1 - Функциональный блок

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

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

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

управляющими только относящиеся к потокам документов и информации, а дугами-механизмами только ресурсы.

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

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

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

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

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

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

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

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 рисунке 4. 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.

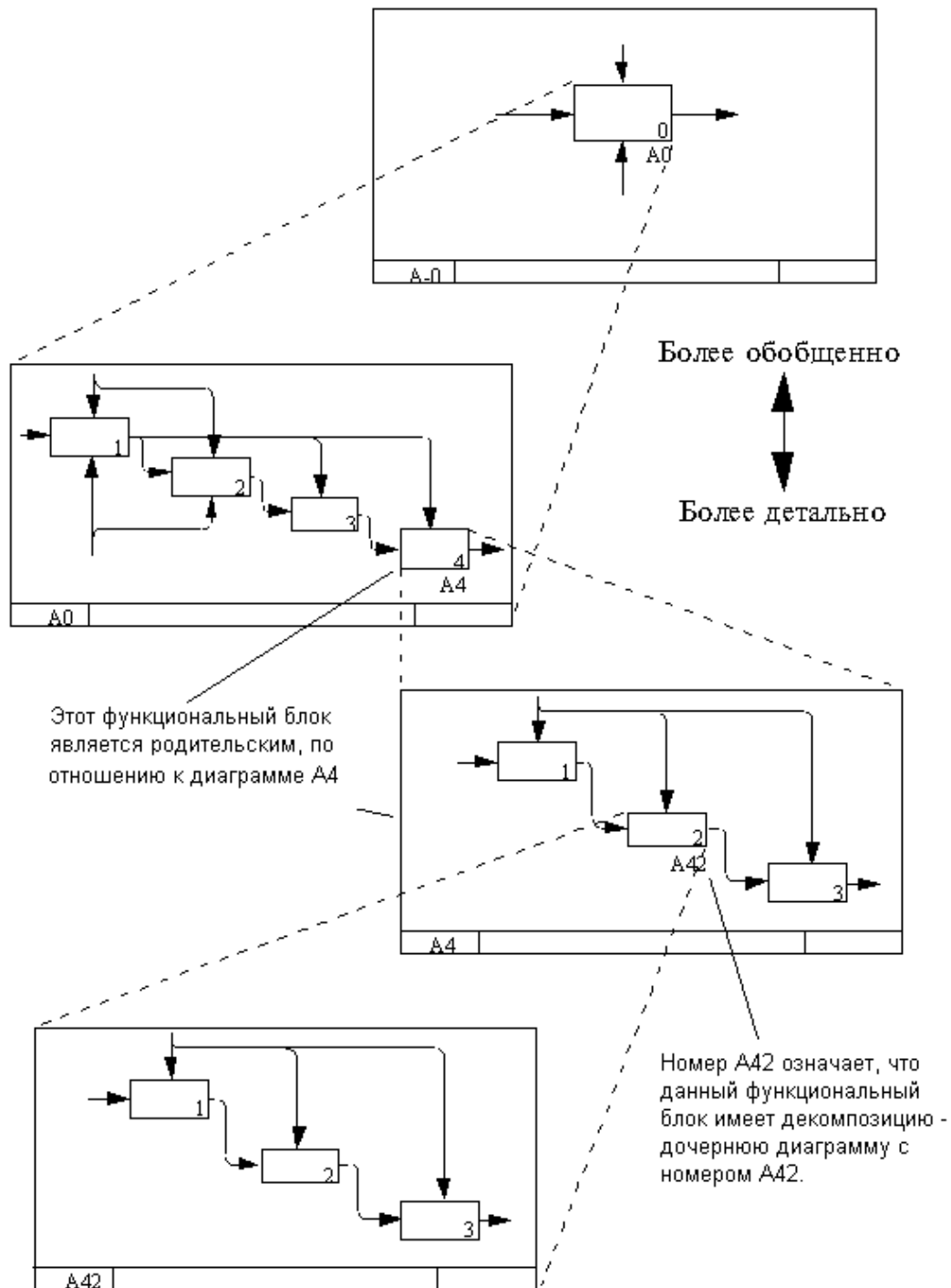


Рисунок 3 - Декомпозиция функциональных блоков

Часто бывают случаи, когда отдельные интерфейсные дуги не имеет смысла продолжать рассматривать в дочерних диаграммах ниже какого-то определенного уровня в иерархии, или наоборот - отдельные дуги не имеют практического смысла выше какого-то уровня. Например, интерфейсную дугу, изображающую “деталь” на входе в функциональный блок “Обработать на токарном станке” не имеет смысла отражать на диаграммах более высоких уровней – это будет только перегружать диаграммы и делать их сложными для восприятия. С другой стороны, случается необходимость избавиться от отдельных “концептуальных” интерфейсных дуг и не детализировать их глубже некоторого уровня. Для решения подобных задач в стандарте IDEF0 предусмотрено понятие

туннелирования. Обозначение “туннеля” (Arrow Tunnel) в виде двух круглых скобок вокруг начала интерфейсной дуги обозначает, что эта дуга не была унаследована от функционального родительского блока и появилась (из “туннеля”) только на этой диаграмме. В свою очередь, такое же обозначение вокруг конца (стрелки) интерфейсной дуги в непосредственной близости от блока – приёмника означает тот факт, что в дочерней по отношению к этому блоку диаграмме эта дуга отображаться и рассматриваться не будет. Чаще всего бывает, что отдельные объекты и соответствующие им интерфейсные дуги не рассматриваются на некоторых промежуточных уровнях иерархии – в таком случае, они сначала “погружаются в туннель”, а затем, при необходимости “возвращаются из туннеля”.

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

Принципы ограничения сложности IDEF0-диаграмм

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

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

Дисциплина групповой работы над разработкой IDEF0-модели

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

- Создание модели группой специалистов, относящихся к различным сферам деятельности предприятия. Эта группа в терминах IDEF0 называется авторами (Authors). Построение первоначальной модели является динамическим процессом, в течение которого авторы опрашивают компетентных лиц о структуре различных процессов. На основе имеющихся положений, документов и результатов опросов создается черновик (Model Draft) модели.
- Распространение черновика для рассмотрения, согласований и комментариев. На этой стадии происходит обсуждение черновика модели с широким спектром компетентных лиц (в терминах IDEF0- читателей) на предприятии. При этом каждая из диаграмм черновой модели письменно критикуется и комментируется, а затем передается автору. Автор, в свою очередь, также письменно соглашается с критикой или отвергает её с изложением логики принятия решения и вновь возвращает откорректированный черновик для дальнейшего рассмотрения. Этот цикл продолжается до тех пор, пока авторы и читатели не придут к единому мнению.
- Официальное утверждение модели. Утверждение согласованной модели происходит руководителем рабочей группы в том случае, если у авторов модели и читателей отсутствуют разногласия по поводу ее адекватности. Окончательная модель

представляет собой согласованное представление о предприятии (системе) с заданной точки зрения и для заданной цели.

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

Особенности национальной практики применения функционального моделирования средствами IDEF0

В последние годы интерес в России к методологиям семейства IDEF неуклонно растет. Это я постоянно наблюдаю, просматривая статистику обращений к своей персональной web-странице (<http://consulting.psi.ru>), на которой кратко описаны основные принципы этих стандартов. При этом интерес к таким стандартам, как IDEF3-5 я бы назвал теоретическим, а к IDEF0 вполне практически обоснованным. Собственно говоря, первые Case-средства, позволяющие строить DFD и IDEF0 диаграммы появились на российском рынке еще в 1996 году, одновременно с выходом популярной книги по принципам моделирования в стандартах SADT.

Тем не менее, большинство руководителей до сих пор расценивают практическое применение моделирования в стандартах IDEF скорее как дань моде, нежели чем эффективный путь оптимизации существующей системы управления бизнесом. Вероятнее всего это связано с ярко выраженным недостатком информации по практическому применению этих методологий и с непременным софтверным уклоном абсолютного большинства публикаций.

Не секрет, что практически все проекты обследования и анализа финансовой и хозяйственной деятельности предприятий сейчас в России так или иначе связаны с построением автоматизированных систем управления. Благодаря этому, стандарты IDEF в понимании большинства стали условно неотделимы от внедрения информационных технологий, хотя с их помощью порой можно эффективно решать даже небольшие локальные задачи, буквально при помощи карандаша и бумаги.

При проведении сложных проектов обследования предприятий, разработка моделей в стандарте IDEF0 позволяет наглядно и эффективно отобразить весь механизм деятельности предприятия в нужном разрезе. Однако самое главное – это возможность коллективной работы, которую предоставляет IDEF0. В моей практической деятельности было достаточно много случаев, когда построение модели осуществлялось с прямой помощью сотрудников различных подразделений. При этом, консультант за достаточно короткое время объяснял им основные принципы IDEF0 и обучал работе с соответствующим прикладным программным обеспечением. В результате, сотрудники различных отделов создавали IDEF-диаграммы деятельности своего функционального подразделения, которые должны были ответить на следующие вопросы:

- Что поступает в подразделение “на входе”?
- Какие функции, и в какой последовательности выполняются в рамках подразделения?
- Кто является ответственным за выполнение каждой из функций?
- Чем руководствуется исполнитель при выполнении каждой из функций?
- Что является результатом работы подразделения (на выходе)?

После согласования черновиков диаграмм внутри каждого конкретного подразделения, они собираются консультантом в черновую модель предприятия, в которой увязываются все входные и выходные элементы. На этом этапе фиксируются все неувязки отдельных диаграмм и их спорные места. Далее, эта модель вновь проходит через функциональные отделы для дальнейшего согласования и внесения необходимых корректив. В результате, за достаточно короткое время и при привлечении минимума человеческих ресурсов со стороны консультационной компании (а эти ресурсы, как

известно, весьма недешевы), получается IDEF0-модель предприятия по принципу “Как есть”, причем, что немаловажно, она представляет предприятие с позиции сотрудников, которые в нем работают и досконально знают все нюансы, в том числе неформальные. В дальнейшем, эта модель будет передана на анализ и обработку к бизнес-аналитикам, которые будут заниматься поиском “узких мест” в управлении компанией и оптимизацией основных процессов, трансформируя модель “Как есть” в соответствующее представление “Как должно быть”. На основании этих изменений и выносится итоговое заключение, которое содержит в себе рекомендации по реорганизации системы управления.

Разумеется, подобный подход требует ряда организационных мер, в первую очередь со стороны руководства обследуемого предприятия. Это обусловлено тем, что эта техника подразумевает возложение на некоторых сотрудников дополнительных обязанностей по освоению и практическому применению новых методологий. Однако в конечном итоге это оправдывает себя, так как дополнительные один-два часа работы отдельных сотрудников в течение нескольких дней позволяют существенно экономить средства на оплату консультационных услуг сторонней компании (которые в любом случае будут отрываться от работы тех же работников анкетами и вопросами). Что касается самих работников предприятия, так или иначе выраженного противодействия с их стороны я в своей практике не встречал.

Вывод из всего этого можно сделать следующий: совершенно не обязательно каждый раз самим придумывать решения для стандартных задач. Всегда, когда Вы сталкиваетесь с необходимостью анализа той или иной функциональной системы (от системы проектирования космического корабля, до процесса приготовления комплексного ужина) – используйте годами проверенные и обкатанные методы. Одним из таких методов и является IDEF0, позволяющий с помощью своего простого и понятного инструментария решать сложные жизненные задачи.

3. Объекты исследования, оборудование, материалы и наглядные пособия

Объектом исследования данной лабораторной работы является структурно-функциональная модель сложной системы.

Для выполнения работы необходимы ПК и соответствующее программное обеспечение:

- MS WINDOWS;
- MS Visio
- MS Office (для оформления отчета).

4. Задание на работу (рабочее задание)

Построить функциональную модель сложной системы с помощью диаграмм IDEF0.

Разработать структурно-функциональную модель следующих процессов:

1. Изготовление стула
2. Изготовление втулки
3. Написание реферата по курсу «Философия»
4. Выполнение контрольно-курсовой работы по курсу «Модели и методы анализа проектных решений»
5. Посадка картофеля
6. Уборка комнаты
7. ТО для автомобиля
8. Ремонт компьютера
9. Подключение Интернета
10. Подготовка и проведение зачетной сессии 1 семестра

5. Ход работы (порядок выполнения работы)

Для выполнения контрольно-курсовой работы необходимо:

- 1) Ознакомиться с теоретической справкой;
- 2) Разработать концепцию структурно-функциональной модели;
- 3) Выполнить декомпозицию модели (минимум 2 уровня);
- 4) Оформить графическое представление в среде Visio;
- 5) Оформить отчет по лабораторной работе;
- 6) Защитить работу преподавателю, ответив на вопросы по ее содержанию и выполнению.

6. Содержание отчета

В отчете должны присутствовать следующие пункты:

- 1) задание;
- 2) концепция структурно-функциональной модели;
- 3) декомпозиция модели (минимум 2 уровня)
- 4) Глоссарий к каждому уровню модели;

7. Список использованных источников

Ямникова О.А. Разработка САПР: учеб. пособие / О.А. Ямникова. – Тула: Издв-во ТулГУ, 2010 – 67 с.