

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ  
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЁВА  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

*М.Е. Кременецкая, А.В. Иващенко*

МОДЕЛИРОВАНИЕ ПРОЦЕССОВ ЖИЗНЕННОГО  
ЦИКЛА ИЗДЕЛИЙ АВИАЦИОННОЙ ТЕХНИКИ  
ПО МЕТОДОЛОГИИ ARIS

Утверждено Редакционно-издательским советом университета  
в качестве лабораторного практикума

САМАРА  
Издательство СГАУ  
2010

УДК СГАУ:629.7(075)

ББК 68.53

К 795

*Кременецкая М. Е.*

К 795 **МОДЕЛИРОВАНИЕ ПРОЦЕССОВ ЖИЗНЕННОГО ЦИКЛА ИЗДЕЛИЙ АВИАЦИОННОЙ ТЕХНИКИ ПО МЕТОДОЛОГИИ ARIS** : лабораторный практикум / *М.Е. Кременецкая, А.В. Иващенко*. - Самара: Изд-во СГАУ, 2010. - 88 с.

**ISBN 978-5-7883-0819-7**

Моделирование процессов жизненного цикла изделий является важной и актуальной задачей, которую приходится решать при проведении комплексной автоматизации производства изделий авиационной техники и проектировании единого информационного пространства предприятий аэрокосмической отрасли.

Поэтому специалистам в области авиастроения и CALS-технологий необходимо обладать знаниями и навыками по применению современных методологий моделирования бизнес-процессов.

В процессе освоения материала магистры получают универсальные знания и навыки, необходимые для реального использования методологии моделирования процессов ARIS, что обеспечивает непрерывность подготовки специалистов и облегчает их интеграцию в информационное сообщество на промышленном предприятии.

Лабораторный практикум содержит рекомендации по выполнению лабораторных работ, соответствующих курсу «Моделирование процессов жизненного цикла изделий авиационной техники». Приведены основные научно-теоретические сведения по методологии моделирования процессов ARIS и построению графических диаграмм согласно основным представлениям.

Предназначен для магистров направления 160100.68 «Авиационное строительство», обучающихся по магистерской программе «Проектирование, конструкция и CALS-технологии в авиационной технике».

Подготовлен на кафедре конструкции и проектирования летательных аппаратов.

УДК СГАУ: 629.7(075)

ББК 68.53

**ISBN 978-5-7883-0819-7**

© Самарский государственный  
аэрокосмический университет, 2010

## СОДЕРЖАНИЕ

Общие сведения.....	4
1 Краткое описание методологии ARIS.....	7
1.1 Концепция архитектуры ARIS.....	7
1.2 Сетевая многопользовательская работа.....	11
1.3 Модели ARIS.....	12
1.4 Методологические фильтры.....	14
1.5 Проводник — ARIS Explorer.....	14
1.5.1 Основные элементы управления.....	14
1.5.2 Работа с папками.....	16
1.5.3 Моделирование.....	16
1.5.4 Объекты моделей.....	18
1.5.5 Детализация (декомпозиция).....	19
1.5.6 Связи объектов.....	20
1.5.7 Атрибуты элементов ARIS.....	20
2 Лабораторные работы.....	23
Лабораторная работа № 1. Знакомство со средой моделирования ARIS. Построение организационной структуры предприятия и функциональной модели.....	23
Лабораторная работа № 2. Построение диаграммы добавленной стоимости VAD (Value Added Chain Diagrams).....	39
Лабораторная работа № 3. Построение цепочки процесса, управляемого событиями eEPC (extended Event Driven Process Chains).....	46
Лабораторная работа №4. Построение информационной модели eERM (Extended entity-relationship model).....	60
Библиографический список.....	88

## ОБЩИЕ СВЕДЕНИЯ

Фактически проблема комплексной автоматизации стала актуальной для каждого предприятия аэрокосмической отрасли. Уже не стоит вопрос «надо или не надо автоматизировать», предприятия столкнулись с проблемой: каким образом это осуществить. Подобная переориентация предприятий объясняется следующими основными причинами:

- повышением степени организационной и финансовой самостоятельности;
- необходимостью изготавливать конкурентоспособную продукцию;
- стремлением ряда западных компаний производить свои товары в России;
- завершением периода «островковой» автоматизации;
- возрастающей ориентацией предприятий на бизнес-процессы (см. [«Бизнес-процесс»](#)), т.е. деятельности, имеющие ценность для клиента;
- появлением на рынке как зарубежных, так и отечественных систем автоматизации, опыта их внедрения и использования и др.

Автоматизация деятельности большинства машиностроительных предприятий во многом сводится к внедрению и освоению CALS-технологий (см. [CALS-технологии](#)) [1] Как правило, для этого необходим предваряющий автоматизацию этап - наведение порядка в их деятельности, создание рациональных технологий и бизнес-процессов. Для этого проводят построение моделей (см. [«Модель»](#)) деятельности предприятия по результатам обследования и анализа. Разрабатывают модели следующих двух видов:

- модели «как есть», представляющие собой «снимок» положения дел на предприятии (оргштатная структура, взаимодействия подразделений, принятые технологии, автоматизированные и неавтоматизированные бизнес-процессы и т.д.) на момент обследования и позволяющие понять, что делает и как функционирует данное предприятие с позиций системного анализа, а также на основе автоматической верификации выявить ряд ошибок и узких мест и сформулировать ряд предложений по улучшению ситуации;
- модели «как должно быть», интегрирующие перспективные предложения руководства и сотрудников предприятия, экспертов и системных аналитиков и позволяющие сформировать видение новых рациональных технологий работы предприятия. [2]

Каждая из моделей включает в себя полную организационно-структурную, функциональную модель деятельности, информационную модель (как правило, с использованием нотации «сущность-связь»), а так же, в случае необходимости, событийную (описывающую поведение) модель.

Построенные модели являются не просто реализацией начальных этапов разработки системы и техническим заданием на последующие этапы. Они представляют собой самостоятельный отделяемый результат,

имеющий большое практическое значение. Например, модель “как есть” включает в себя существующие неавтоматизированные технологии, работающие на предприятии. Формальный анализ этой модели позволит выявить узкие места в технологиях и предложить рекомендации по ее улучшению (независимо от того, предполагается на данном этапе автоматизация предприятия или нет). Она позволяет осуществлять автоматизированное и быстрое обучение новых работников конкретному направлению деятельности предприятия (так как ее технология содержится в модели) с использованием диаграмм (известно, что «одна картинка стоит тысячи слов»). С ее помощью можно осуществлять предварительное моделирование нового направления деятельности с целью выявления новых потоков данных, взаимодействующих подсистем и бизнес-процессов. [2]

Для обеспечения согласованности модели и ее высокой степени формализации, которая требуется при описании технического решения, необходимо использовать современные технологии автоматизированного проектирования – CASE-технологии (Computer Aided Software/System Engineering).

CASE-технология представляет собой методологию проектирования, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область и производить ее анализ на всех этапах разработки.

Основные подходы к моделированию (структурный, объектно-ориентированный, процессный) изложены в наиболее распространенных методологиях, таких как [SADT](#) (Structured Analysis and Design Technique), [UML](#) (Unified Modeling Language), [ARIS](#) (Architecture of Integrated Information Systems).

Предложенный лабораторный практикум включает теоретические сведения и четыре лабораторных работы, направленные на освоение методологии ARIS и соответствующих CASE средств, поддерживающих указанную методологию.

При разработке этого практикума ставились две задачи: с одной стороны, познакомить магистров с CASE средствами, концепциями моделированию процессов, научить работать с интерфейсом пользователя, с другой – привить навыки системного анализа, моделирования данных и процессов. В связи с этим лабораторные работы содержат описание и задание на построение учебных моделей предметных областей, позволяющие осуществить полный цикл моделирования бизнес-процессов.

В первой лабораторной работе магистры знакомятся со средой моделирования ARIS Toolset, приобретают навыки построения организационной структуры предприятия и «дерева функций».

Вторая работа посвящена построению диаграммы добавленной стоимости VAD (Value Added Chain Diagrams), применяемой для описания процессов верхнего уровня.

В третьей лабораторной работе изучаются принципы построения

цепочки процесса, управляемого событиями eEPC (extended Event Driven Process Chains), который является дальнейшей детализацией процессов верхнего уровня. Это более подробный способ описания процесса, где отражается логика его выполнения, и указываются основные данные, в первую очередь ресурсное окружение выполняемых процессов и функций.

Четвертая лабораторная работа предполагает изучение принципов построения информационной модели eERM (Extended entity -relationship model), являющейся расширенной нотацией Чена. Данные диаграммы предназначены для разработки моделей данных и обеспечивают стандартный способ определения данных и отношений между ними.

Выполнение каждой лабораторной работы рассчитано на 4 академических учебных часа. На каждом занятии магистры получают свой вариант задания. Предполагается, что работу они осуществляют либо индивидуально, либо небольшими бригадами (не более 3 человек).

Лабораторный практикум может быть использован при проведении лабораторных работ, а также и при самостоятельном изучении дисциплины «Моделирование процессов жизненного цикла продукции изделий авиационной техники».

# 1 КРАТКОЕ ОПИСАНИЕ МЕТОДОЛОГИИ ARIS

## 1.1 Концепция архитектуры ARIS

Методология ARIS основана на разработанной профессором А.В. Шеером теории «Архитектура интегрированных информационных систем» (Architecture of Integrated Information System — ARIS). Она определяет принципы моделирования практически всех аспектов деятельности организаций, что является ее коренным отличием от других методологий. Согласно терминологии, принятой в области структурного анализа, термин «архитектура» описывает типы используемых методов, их функциональные свойства и взаимоотношения между составными частями моделируемой системы.

Методология ARIS предполагает определенный подход к формализации информации о деятельности организации и представление ее в виде графических моделей, удобном для понимания и анализа. Модели, создаваемые по методологии ARIS, отражают существующую ситуацию с той или иной степенью приближенности. Степень детализации описания зависит от целей проекта, в рамках которого проводится моделирование. Модели ARIS могут быть использованы для анализа и выработки различного рода решений по реорганизации деятельности предприятия, в том числе по внедрению информационной системы управления, разработке систем менеджмента качества.

Концепция ARIS основана на идее интеграции, которая является составной частью комплексного анализа бизнес-процессов. Первый шаг при создании архитектуры состоит в разработке модели бизнес-процесса, описывающей все его основные функции. Полученная таким образом чрезвычайно сложная модель разделится на подмодели, или типы моделей, в соответствии с типами представлений. Это позволяет существенно снизить степень ее сложности. Содержимое типов моделей может быть описано методами, предназначенными для конкретного типа представления.

Другим подходом, уменьшающим сложность модели бизнес-процесса, является анализ каждого типа моделей на различных уровнях. В соответствии с концепцией модели жизненного цикла (см. [«Жизненный цикл»](#)) различные методы описания информационных систем дифференцируются по степени их близости к информационным технологиям. Это гарантирует целостность описания на всех этапах, начиная от проблем управления бизнесом до технической реализации информационной системы.

Методология ARIS реализует принципы структурного анализа (см. [«Структурный анализ»](#)) и позволяет определить и отразить в моделях основные компоненты организации, протекающие процессы, производимую и потребляемую продукцию, используемую информацию, а так же выявить взаимосвязи между ними. [6]

Создаваемые модели представляют собой документированную совокупность знаний о системе управления, включая организационную структуру, протекающие процессы, взаимодействия между организацией и субъектами рынка, состав и структуру документов, последовательность шагов процессов, должностные инструкции отделов и их сотрудников. В отличие от других подходов, методология ARIS предполагает хранение всей информации в едином репозитории (см. «[Репозиторий](#)»), что обеспечивает целостность и непротиворечивость процесса моделирования и анализа, а также позволяет проводить верификацию моделей.

Преимущества методологии ARIS:

- возможность рассматривать объект с разных точек зрения; разные уровни описания, обеспечивающие поддержку концепции жизненного цикла систем; дифференцированный взгляд на анализируемый объект (организацию, систему управления и т.д.);
- богатство методов моделирования, отражающих различные аспекты исследуемой предметной области (см. «[Предметная область](#)»), позволяет моделировать широкий спектр систем (организационно-хозяйственных, технологических и прочих);
- единый репозиторий; все модели и объекты создаются и хранятся в единой базе проекта, что обеспечивает построение интегрированной и целостной модели предметной области;
- возможность многократного применения результатов моделирования; накопленное корпоративное знание обо всех аспектах деятельности организации может в дальнейшем служить основой при разработке различных проектов непосредственно в среде ARIS и с использованием интерфейсов и других средств.

В общем случае архитектура ARIS выделяет в организации такие подсистемы как:

- *Организационная*. Определяет структуру организации — иерархию подразделений, должностей и конкретных лиц, многообразие связей между ними, а также территориальную привязку структурных подразделений;
- *Функциональная*. Содержит описание функций бизнес-процесса, определяет функции, выполняемые в организации и их взаимосвязи между собой;
- *Подсистемы входов/выходов*. Определяют потоки используемых и производимых продуктов и услуг;
- *Информационная (подсистема данных)*. Описывает состояния информационных объектов (данных), получение, распространение и доступ к информации;
- *Подсистема процессов управления*. Определяет логическую последовательность выполнения функций посредством событий и сообщений. Можно сказать, что подсистема управления — это совокупность разнесенных во времени сообщений разного рода;
- *Подсистема целей организации*. Описывает иерархию целей, достигаемых в ходе выполнения того или иного процесса;



- *Подсистема средств производства.* Описывает жизненный цикл основных и вспомогательных средств производства;
- *Подсистема человеческих, ресурсов.* Описывает прием на работу, обучение и продвижение по службе персонала организации;
- *Подсистема расположения организационных структур.* Описывает территориальное расположение организационных единиц. [4]

Все эти подсистемы организации в реальности и в моделях должны быть связаны между собой. Методология ARIS дает возможность описывать достаточно разнородные подсистемы в виде взаимоувязанной и взаимосогласованной совокупности различных моделей, которые хранятся в едином хранилище. Именно взаимосвязанность и взаимосогласованность моделей являются отличительными особенностями методологии ARIS.

В соответствии с правилами структурного анализа каждая из этих подсистем разбивается на элементарные блоки (модули), совокупность которых и составляет нотацию структурной модели той или иной подсистемы организации.

Естественно, что эти подсистемы не являются обособленными. Они взаимно проникают друг в друга, и поэтому одни и те же элементарные модули могут использоваться для описания различных структурных моделей. Для устранения избыточности методология ARIS ограничивает число типов моделей.

В связи с этим в методологии ARIS выделено четыре основных вида моделей, отражающих основные аспекты организации — пять типов представлений:

- *организационные модели*, описывающие иерархическую структуру системы — иерархию организационных подразделений, должностей, полномочий конкретных лиц, многообразие связей между ними, а также территориальную привязку структурных подразделений;
- *функциональные модели*, описывающие функции (процессы, операции), выполняемые в организации;
- *информационные модели* (модели данных), отражающие структуру информации, необходимой для реализации всей совокупности функций системы;
- *модели процессов/управления*, представляющие комплексный взгляд на реализацию деловых процессов в рамках системы и объединяющие вместе другие модели;
- *модели входов/выходов*, описывающие потоки материальных и нематериальных входов и выходов, включая потоки денежных средств.

Остальные подсистемы могут моделироваться с использованием объектов, входящих в перечисленные выше типы представления. Графически такой подход представлен на рисунке 1 (см. [Рисунок 1](#)).

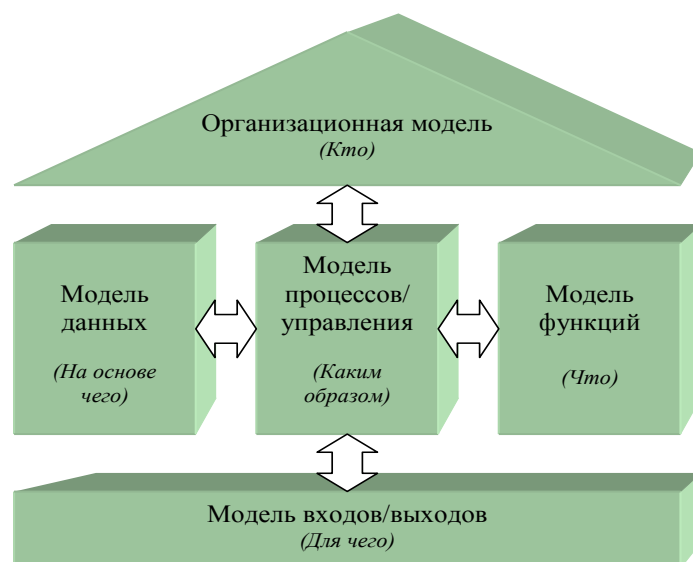


Рисунок 1 - Взаимосвязь видов моделей ARIS (здание ARIS)

Типы представления являются первой компонентой архитектуры. Они позволяют структурировать бизнес-процессы и выделять их составные части, что делает рассмотрение более простым. Применение этого принципа позволяет с различных точек зрения описывать содержание отдельных частей бизнес-процесса, используя специальные методы, наиболее полно соответствующие каждой точке зрения. Это избавляет пользователя от необходимости учитывать множество связей и соединений.

В рамках каждого типа представления создаются модели, отражающие ту или иную сторону исследуемой системы. Методология ARIS включает большое количество методов моделирования, в том числе известных как диаграммы Чена ERM, язык UML (Unified Modeling Language), методики OMT (Object Modeling Technique), BSC (Balanced Scorecard) и т.п. [6, 7]

Достоинство такого подхода заключается в том, что при анализе деятельности организации каждому аспекту можно уделять достаточное внимание, не отвлекаясь на его связь с другими аспектами. И только после детального изучения всех аспектов можно перейти к построению интегрированной модели, отражающей все существующие связи между подсистемами организации.

Организационные и функциональные модели, а также модели данных, входов/выходов и процессов/управления рассматриваются как поля в специальной базе данных, называемой репозиторием, Репозиторий является ядром информационной системы, реализующей методологию ARIS. Он оказывает решающее воздействие на эффективность применения моделей. Интеграция различных видов моделей становится возможной благодаря хранению их в едином репозитории.

Методология ARIS не накладывает ограничений на последовательность подготовки пяти типов представления. Процесс

анализа и проектирования можно начинать с любого из них, в зависимости от конкретных условий и целей, стоящих перед исполнителями.

Интегрированная инструментальная среда ARIS является сложной системой и состоит из комплекса взаимосвязанных и взаимодополняющих модулей, выполняющих различные функции.

Средства ARIS могут применяться как однопользовательская среда, а также поддерживать коллективные разработки в среде «клиент-сервер».

Модуль ARIS *Easy Design* представляет собой упрощенное средство моделирования и анализа, имеющее ограниченные функциональные возможности по сравнению с ARIS *Toolset*.

С точки зрения методологии, ARIS *Easy Design* поддерживает все методы моделирования, применяемые в ARIS *Toolset*. В этом отношении между ними существует полная совместимость.

## 1.2 Сетевая многопользовательская работа

Инструментальная система ARIS является многопользовательской и может работать в компьютерной сети, объединяющей множество отдельных рабочих мест.

Сеть ARIS является главной папкой (структурный элемент MS Windows) структуры, отображаемой в проводнике (*ARIS Explorer*). Сеть включает в себя все серверы баз ARIS, к которым имеет доступ клиент ARIS. Вместо имени *ARIS Network* главной папке можно дать любое другое имя. Через данный элемент устанавливается связь со всеми остальными серверами баз ARIS.

Основной структурной единицей сети ARIS является сервер. Локальным называется сервер, расположенный на конкретном рабочем месте. Он обладает всеми свойствами обычного сервера. ARIS позволяет иметь практически неограниченное число серверов. Любой из серверов содержит одну или несколько баз данных ARIS.

Сервер включает в себя:

- *метаданные ARIS*, состоящие из *метамodelей ARIS* — описаний всех моделей, доступных пользователю при моделировании, и конфигурационной базы, которая хранит информацию об используемых на данном сервере методологических фильтрах, шаблонах моделей, используемом формате шрифтов и описания диаграмм;
- *неограниченное число баз данных*, содержащих информацию о модели организации (или ее части).

Приведенные выше понятия отображаются на левой панели одного из основных модулей ARIS — его проводника.

С технической точки зрения, термин *возможности многопользовательской работы* подразумевает, что локальные компьютеры (клиенты), на которых установлена система ARIS (ПО клиента), соединены сетью с одним или несколькими компьютерами — серверами баз ARIS. Все данные пользователей обрабатываются системой,

установленной на этих серверах.

Каждый клиент ARIS имеет доступ к данным (т.е. моделям, именам моделей, именам папок, структуре папок, описаниям объектов, атрибутам и т.д.), которые хранятся на одном или нескольких серверах. Чтобы пользователь мог изменить или просмотреть эти данные, они должны быть считаны с сервера, переданы по сети, а затем обработаны и отображены ПО клиента. Точно так же измененные данные могут быть переданы от клиента к управляющей системе базы и сохранены в ней.

Многопользовательский режим работы означает, что несколько пользователей могут работать с базой одновременно. В большинстве случаев при многопользовательском режиме работы ни один пользователь не может эксклюзивно войти в базу, так как это закрывает доступ к ней другим пользователям. Временно закрытыми для записи могут оказаться лишь отдельные объекты базы — это происходит в тот момент, когда один из пользователей вносит или записывает в базу изменения, связанные с этими объектами. Данное решение значительно сокращает временные задержки, возникающие при конфликтах между системами различных пользователей.

Право вносить изменения в модель на уровне экземпляров дается только первому пользователю, открывшему эту модель. Так, второй пользователь, который открывает ту же модель, не может перемещать объекты внутри нее или вносить изменения свойства модели. Однако этот пользователь может переименовать описания объектов с помощью проводника и редактировать атрибуты соответствующих описаний объектов с помощью модуля *ARIS Attributes*.

Чтобы исключить параллельное редактирование, атрибут, изменяемый пользователем, становится защищенным от записи для других пользователей. Только первый пользователь, который открыл атрибут для редактирования, является единственным пользователем, которому дается право вносить изменения в этот атрибут. Редактируемые атрибуты остаются защищенными от записи до тех пор, пока внесенные данным пользователем изменения не будут сохранены. [5]

### 1.3 Модели ARIS

Процесс работы в архитектуре ARIS состоит в создании набора различных видов моделей и проработка каждой из них по уровням (детальности) описания в соответствии со сложностью сформулированной проблемы.

В архитектуре ARIS зафиксирован ряд видов моделей, каждая из которых «расписывается» по уровням детализации.

Критерии для выбора моделей следующие:

- простота и выразительность средств изображения;
- поддержка смыслового содержания;
- возможность отображения специфики предмета.

При построении моделей методология ARIS требует соблюдения определенных принципов. К ним относятся:

- корректность модели;
- релевантность (следует моделировать только те фрагменты реальной системы, которые соответствуют назначению системы, т.е. модель не должна содержать избыточной информации);
- соизмеримость затрат и выгод;
- прозрачность, т.е. понятность и удобство использования модели;
- сравнимость моделей;
- иерархичность;
- систематизация структуры, что предполагает в качестве обязательного условия возможность интеграции моделей различных типов.

Модель характеризуется типом, именем и свойствами. Тип модели определяет, что описывает данная модель — организационную структуру, функции, данные, процессы или выходы. Имя модели является частью ее атрибутов. Модель ARIS так же является частью более развернутой модели организации. В то же время модель может представлять отдельный объект, будучи его детальным описанием. [6]

Модели классифицируются с помощью методологических фильтров.

Каждая модель ARIS содержит:

- *объекты* — неделимые части модели, выделенные по какому-либо признаку, сформулированному в соответствии с методологией ARIS, и имеющие набор изменяемых характеристик — свойств, описывающих их поведение;
- *связи между объектами* — описанные взаимоотношения между объектами, имеющими свои свойства и характеристики. Так же как и объекты, связи характеризуются свойствами, выделены внешний вид и атрибуты.

Модель может включать:

- внешние встроенные объекты, например, рисунки, документы текстовых редакторов и т.п.;
- текст, размещенный в любом месте модели;
- геометрические фигуры.

Каждая модель имеет ряд свойств:

- атрибуты (Attributes),
- заполненные атрибуты (Maintained Attribute},
- внешний вид модели (Model Appearance);
- варианты (Variants) ;
- предложения по улучшению (Improvement proposals),
- управление изменениями (Change management).

Каждая модель может содержать один или несколько десятков объектов. Число связей, возможных в каждой модели, сильно варьируется. Модели классифицируются при помощи методологических фильтров и классов.

## 1.4 Методологические фильтры

Большинство из доступных видов моделей, включенных в ARIS, никогда не используется в рамках одного проекта.

Для того, чтобы облегчить работу, обеспечив выбор ограниченного числа необходимых моделей, в ARIS реализованы *методологические фильтры*, представляющие собой регулируемые, переключаемые наборы моделей. Пользователь может создавать собственные методологические фильтры.

Основные стандартные методологические фильтры ARIS приведены в таблице 1 (см. [Таблица 1](#)).

Таблица 1 - Основные стандартные методологические фильтры ARIS

Название фильтра	Сокращенное название	Описание
Entire Method (полный фильтр)	FULLMETHOD	Содержит все модели, объекты, отношения, символы и типы атрибутов, определенные в системе ARIS.
Easy Filter (простой фильтр)	EASYFILTER	Содержит простейшие методы моделирования деятельности.
Standard Filter (стандартный фильтр)	STANDARD	Содержит наиболее часто используемые методы моделирования.
Extended Standard Filter (расширенный стандартный фильтр)	EXTENDEDSTANDARD	В дополнение к стандартному фильтру содержит модели и типы объектов, которые предназначены для объектно-ориентированного моделирования.

В ARIS существует понятие стандартного методологического фильтра. *Стандартный методологический фильтр* — это фильтр, который присваивается базе. Он автоматически ставится в соответствие каждому новому пользователю и каждой новой их группе. По умолчанию, в качестве стандартного, каждой базе присваивается методологический фильтр *Easy Filter*.

## 1.5 Проводник — ARIS Explorer

### 1.5.1 Основные элементы управления

*ARIS Explorer* — это основное средство навигации в ARIS, предназначенное для управления базами данных, папками, моделями, объектами и другими элементами (см. [Рисунок 2](#)). Его функционирование схоже с Проводником системы Microsoft Windows.

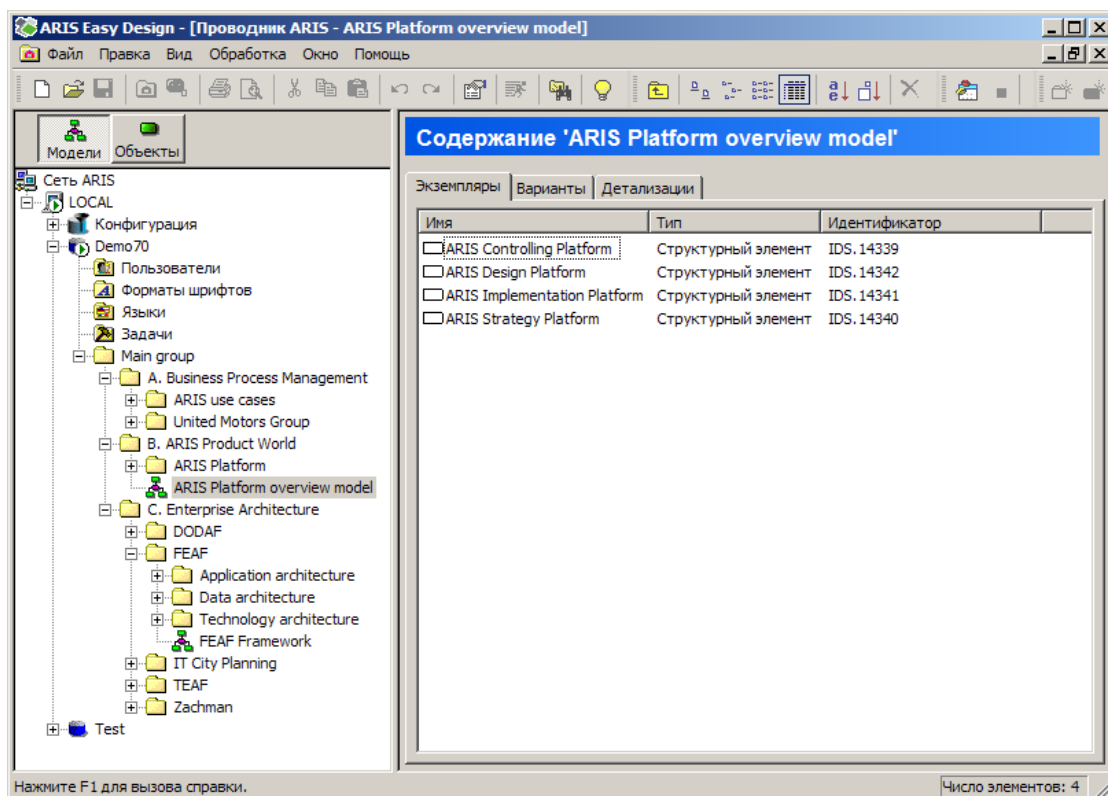


Рисунок 2 - ARIS Explorer

К основным элементам, которые управляются посредством *ARIS Explorer* (Проводника), относятся:

- серверы;
- базы данных;
- пользователи и группы пользователей;
- пароли, определяющие различные привилегии (уровни) доступа к моделям;
- форматы шрифтов;
- языки баз данных;
- методологические фильтры;
- модели и содержащие их папки;
- объекты;
- таблицы.

В окне *ARIS Explorer* (см. [Рисунок 2](#)) отображаются все серверы баз данных, к которым имеется доступ. Используя проводник, можно определить новые соединения или удалить уже существующие.

Окно *ARIS Explorer* разделено на две части. В левой части в виде иерархической структуры, т.е. в форме дерева, представлены элементы ARIS. Слева от каждого из них находится пиктограмма, отображающая его тип. В корне дерева размещается элемент *ARIS Network* (Сетевое окружение). Под ним располагаются серверы, папки и модели.

После того, как в левой части окна выбран элемент структуры, в правой части окна появится список вложенных в него компонентов.

## 1.5.2 Работа с папками

Папки используются в пользовательской базе ARIS для создания иерархической структуры для хранения моделей. Структура папок в базе ARIS сравнима со структурой папок файловой системы на дисках.

Папка верхнего уровня называется главной папкой. Она является корневой в структуре базы ARIS и не может быть удалена. Во всех других отношениях главная папка может быть отредактирована, как любая другая папка. Можно создать любое число папок, подчиненных главной папке. Они, в свою очередь, также могут иметь подчиненные папки.

Папки, как и другие элементы ARIS, обладают определенными свойствами, а именно привилегиями доступа (*Access Privileges*) и атрибутами.

В папках размещаются модели и объекты ARIS. Модели разделяются в соответствии с пятью типами представления и тремя уровнями описания. Это принято представлять в виде здания ARIS.

## 1.5.3 Моделирование

Процесс моделирования начинается с создания новой модели. При создании модели можно воспользоваться Помощником конструирования моделей (*Model Wizard*). [8]

Типы моделей делятся на группы в соответствии с типами представлений организации: *Organization* (Организационные модели), *Data* (Модели данных), *Processes* (Модели процессов), *Functions* (Функциональные модели).

После завершения диалога в окне *Model Wizard* на экране появится пустое окно ARIS Designer для новой модели, в котором можно создавать и изменять объекты и связи между ними, а также вводить атрибуты объектов, связей и самой новой модели.



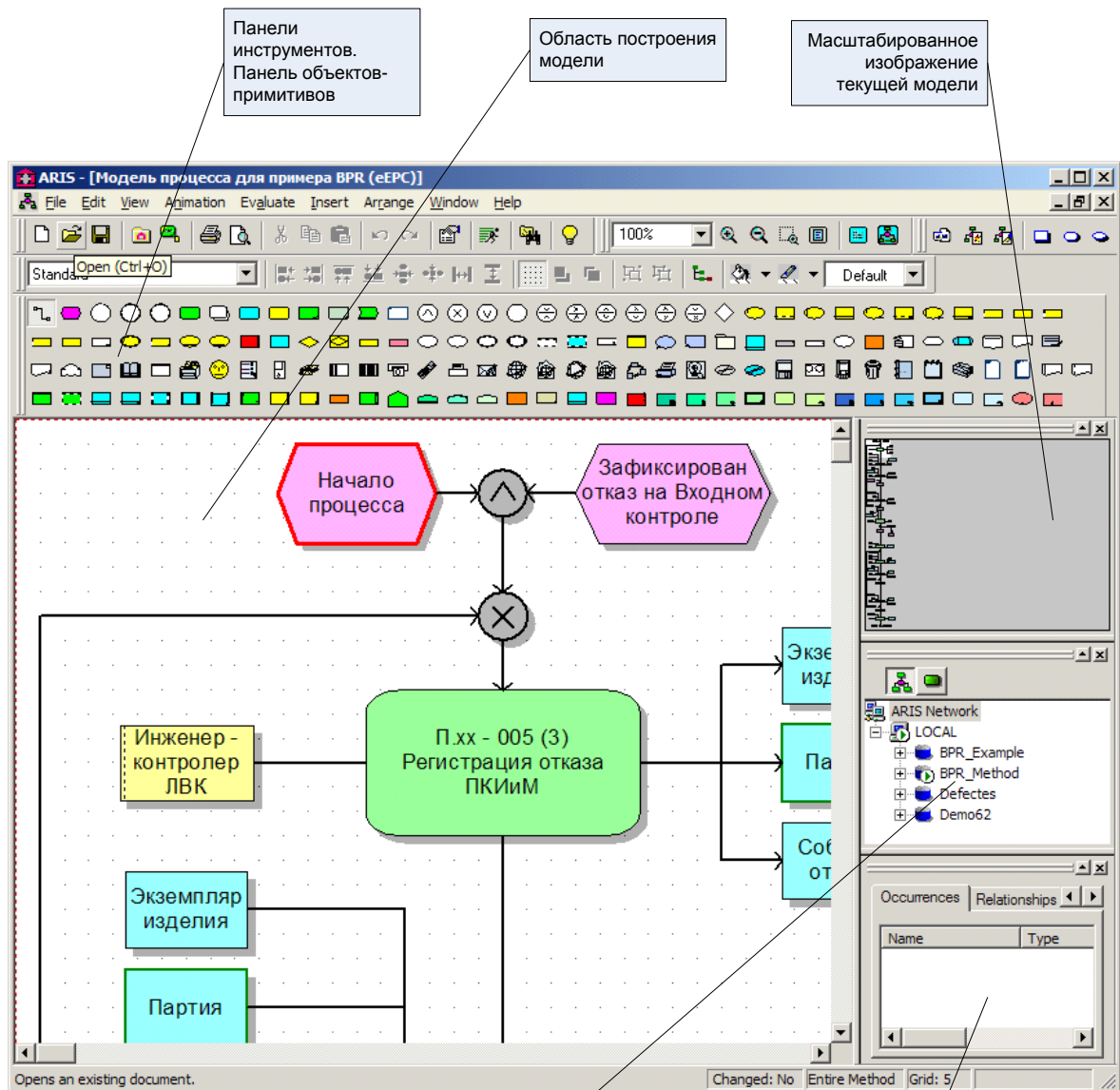


Рисунок 3 - Снимок рабочей среды ARIS

Окно *ARIS Designer* состоит из основного окна, где происходит построение моделей, и вспомогательных окон, которые открываются и закрываются командами меню. Используя команды пункта меню *View* или кнопки панели инструментов *Model View*, можно управлять форматом отображения информации в основном окне *ARIS Designer*

К основному окну моделирования может быть добавлено еще несколько ниже перечисленных вспомогательных окон. В скобках указаны команды меню, которые активизируют эти окна.

Окно *Просмотра модели* (команда меню *View/Model Overview*). В этом окне (на рисунке 3 справа в верхней части (см. [Рисунок 3](#))) представлена модель в уменьшенном масштабе. Белый прямоугольник — это часть модели, которая отражается в основном окне *ARIS Designer*.

Передвигая этот прямоугольник курсором мыши, можно изменять представление в основном окне.

*Окно просмотра объекта* (команда меню *View/Object Window*) — вспомогательное окно для работы с выбранным объектом модели (он окружен маленькими черными квадратами на рисунке. 3 (см. [Рисунок 3](#))). Оно используется для рассмотрения его в увеличенном масштабе, просмотра его связей и т.п. На рисунке 3 (см. [Рисунок 3](#)) это окно расположено справа в нижней части.

*Окно вывода информации* (команда меню *View/Output Window*) - вспомогательное окно для вывода системной информации для пользователя (внизу на рисунке 3 (см. [Рисунок 3](#))).

Окно проводника (команда меню *View/ Explorer Window*) — окно ARIS Explorer (правее центра на рисунке 3 (см. [Рисунок 3](#))).

Находясь в основном окне *ARIS Designer*, можно создавать объекты и связи между ними, а также редактировать их атрибуты. Объект может представлять собой носитель информации или же логический оператор. Все эти типы объектов доступны через панель инструментов *Modeling* (Моделирование). Для каждого типа объекта, в свою очередь, определены символы представления его в модели.

Декомпозиция объекта позволяет определять для объекта текущей модели ссылки на другие модели, детализирующие этот объект.

#### **1.5.4 Объекты моделей**

Каждый тип модели предполагает использование определенного набора объектов (см. «[Объект](#)»).

Каждый объект:

- имеет уникальное имя;
- принадлежит к определенному типу объектов;
- соединен одной или несколькими связями с другими объектами;
- имеет свое определенное значение в методологии и описывается свойствами, определяющими конкретный объект данного типа;
- может использоваться в одной или нескольких типах моделей;
- может создавать свои экземпляры, представленные одним или несколькими символами.

Внешний вид экземпляра объекта может быть изменен в определенных пределах (выбран из нескольких заданных вариантов).

Каждый объект моделей ARIS имеет следующие свойства:

- размещение атрибутов (*Attribute placements*),
- взаимосвязи (*Relationships*);
- назначение (*Assignments*);
- местонахождение (*Occurrences*);
- внешний вид (*Object Appearance*);
- варианты (*Variants*);
- управление изменениями (*Change Management*);

- предложения по улучшениям (*Improvement Proposals*);
- заполненные атрибуты (*Maintained Attributes*);
- атрибуты объекта (*Attributes*).

Объекты классифицируются на:

- *структурно-зависимые*. Эти типы объектов определяют общую структуру модели. Конкретный тип модели подразумевает наличие конкретных структурно-зависимых типов объектов;
- *структурно-независимые*.

При создании нового объекта в модели в базе данных ARIS появляется отдельная запись, задающая описание объекта. Типичные описания объекта включают атрибуты, которые характеризуют объект. Описания объектов должны быть уникальными по отношению к их именам. Таким образом, когда описания заданы, всегда производится проверка, не существует ли уже тот или иной объект в базе.

Экземпляр объекта всегда связан с отдельной моделью. Описание объекта всегда связано с отдельной группой. На основании существующего описания экземпляры данного объекта могут использоваться в других моделях. При изменении характеристик одного из экземпляров объекта меняются свойства всех остальных экземпляров и соответственно его описание в хранилище.

При удалении одного из экземпляров объекта все остальные остаются неизменными. В то же время при удалении описания объекта из хранилища уничтожаются все существующие его экземпляры.

В моделях ARIS используется большое число различных типов объектов, и, соответственно, самих объектов значительно больше, поскольку один тип содержит, как правило, несколько объектов.

### 1.5.5 Детализация (декомпозиция)

Методология ARIS предусматривает возможность описывать объекты более подробно с помощью детализирующих моделей. Такая связь между типом объекта (базовым объектом) и типом модели называется *типом детализации*. Отдельный экземпляр типа детализации, т.е. связь между определенным объектом и определенной моделью, называется *детализацией (декомпозицией)*. Типы детализации, разрешенные к использованию в конкретном случае, зависят от типа объекта, к которому принадлежит базовый объект.

При создании детализации пользователь может создавать связи между базовым объектом и одним или большим количеством объектов из детализированной модели. Такая связь между типами объектов, пересекающая границу между моделями, называется *типом детализирующей связи*. Отдельная связь между двумя объектами называется *детализирующей связью*.

Выбор типов связей и детализации позволяет управлять принципами и глубиной декомпозиции, а также «загруженностью» моделей.

### 1.5.6 Связи объектов

Важную роль при моделировании играет правильный выбор связей между объектами модели. *Связь* — описание взаимоотношений между двумя объектами, имеющая определенные свойства. Каждая связь:

- имеет уникальное имя;
- принадлежит к определенному типу;
- соединяет исходный и конечный объекты;
- характеризуется своими свойствами;
- может создавать свои экземпляры, внешний вид которых может быть изменен в определенных пределах.

*Тип связи* зависит как от типов соединяемых объектов, так и от направления соединения. Между некоторыми типами объектов связи не могут существовать.

Среди связей различают *связи соединения* и *детализирующие связи*.

*Связи соединения* — это такие связи, экземпляры которых представляют собой видимые соединения между двумя объектами внутри одной модели. Соединения классифицируются на структурно-зависимые и структурно-независимые. При детализации можно создавать связи между базовым объектом и одним или большим количеством объектов из детализирующей модели. Связь между типами объектов, пересекающая границу между моделями, называется *типом детализирующей связи*. Отдельная связь между двумя объектами называется *детализирующей связью*.

Каждая связь в рамках методологии ARIS имеет следующие свойства:

- размещение атрибутов (*Attribute placements*),
- внешний вид связи (*Connection Appearance*);
- местонахождение связи (*Connection Occurrences*);
- атрибуты связи (*Attributes*);
- заполненные атрибуты (*Maintained Attributes*).

Между двумя объектами может быть создана одна или несколько связей различных типов. В рамках одной модели число типов связей может меняться от единиц (например, три — в дереве функций) до нескольких десятков (например, 24 — в организационной схеме). Один и тот же тип связи может присутствовать в нескольких моделях.

### 1.5.7 Атрибуты элементов ARIS

Каждый элемент ARIS (кроме серверов) обладает собственным набором атрибутов (см. [«Атрибут»](#)).

Атрибуты организованы в иерархическую древовидную структуру папок, аналогичную структуре «Сеть ARIS». Родственные по смыслу типы атрибутов собраны в папки. Состав папок зависит от типа модели, объекта или связи.

Атрибуты имеют следующие элементы ARIS:

- пользователи (*User*);
- группы пользователей (*User Group*);
- папки (*Groups*);
- базы данных (*Databases*);
- модели (*Models*);
- объекты (*Objects*);
- связи (*Relationships*);
- языки (*Languages*);
- шрифты (*Font formats*);
- текст произвольной формы (*Free Form Text*).

Число атрибутов, особенно для объектов, весьма велико и может достигать нескольких десятков. Выбор типов атрибутов зависит от решаемых задач. Например, при оптимизации продолжительности выполнения процессов необходимо задавать значения времени для выполнения отдельных функций. В других проектах этот атрибут использовать не имеет смысла.

Для того чтобы при моделировании получить понятные и модифицируемые результаты, необходимо решить, какие типы атрибутов будут заполнены в моделях, объектах и связях. На выбор тех или иных типов атрибутов влияют цели моделирования.

В ARIS информация о модели содержится в атрибутах этой модели. В зависимости от типа модели, можно заполнять различные атрибуты. Помимо имени модели, которое должно присутствовать обязательно, рекомендуется также для соответствующей модели заполнять атрибут «описание» (*Description/Definition*). Кроме того, исходя из целей конкретного проекта, рекомендуется заполнять специальные атрибуты.

Особенности некоторых типов атрибутов, содержащихся в главной папке:

- *Name*. Имя рекомендуется указывать обязательно для всех элементов. Исключение составляют только «вспомогательные объекты», такие как *operators/rules* (операторы/правила), например, XOR, AND, и т.д., или *generalization/specialization* (обобщение/специализация) в моделях данных. Атрибут *Name* используется для выделения объекта (при создании копии, поиске, объединении объектов). Уникальность имени имеет очень важное значение. Имя и его уникальность очень важны при объединении баз данных. В этом случае имя может использоваться как идентификатор объекта, т.е. как некоторая альтернатива атрибута *Identifier* (идентификатор).

Тем не менее, система поддерживает возможность создания нескольких объектов одного типа с одинаковыми именами, что иногда является необходимым с точки зрения особенностей проводимого проекта.

- *Identifier*. Идентификатор используется в больших базах данных, чтобы избежать проблем с синонимами/омонимами в именах элементов. К

тому же можно создавать группы идентификаторов с уникальными префиксами для своих областей.

Идентификатор можно создать автоматически. Он состоит из префикса, который может быть легко определен, и числового индекса, задаваемого автоматически. В ARIS идентификатор может быть использован как альтернатива для имени объекта — во-первых, при создании копий объектов, не связанных между собой, и, во-вторых, при объединении объектов, когда в качестве критерия для отбора совпадающих объектов и моделей могут служить как «имя», так и «идентификатор». Идентификатор может быть использован для фильтрации объектов (поиска, вывода в отчет).

Следующие атрибуты заполняются автоматически:

- тип элемента (*Type*);
- время создания (*Time of generations*);
- автор (*Created by*) последнего изменения (*Last change*);
- последний пользователь (*Last user*).

Они выделены серым цветом и недоступны для изменения.

Моделирование процессов по методологии ARIS может содержать следующие этапы:

1. Определение цели моделирования и точки зрения на проект.
2. Построение организационной структуры предприятия и «дерева функций».
3. Построение диаграммы VAD и определение управляющей модели на высоком уровне.
4. Построение диаграмм eEPC/PCD, описывающих логику бизнес-процессов в виде взаимодействующих событий, функций, информационных объектов и организационных единиц.
5. Построение иерархий функций, данных и организационных единиц на уровне описания требований путем обобщения или уточнения абстракций, появившихся на диаграммах VAD и eEPC/PCD.
6. Построение информационной модели на диаграмме eERM, обеспечивающей стандартный способ определения данных и отношений между ними.
7. Проработка функционального, информационного и организационного представлений на уровнях спецификации и реализации.
8. Корректировка eEPC/PCD и VAD диаграмм по результатам проработки функционального, информационного и организационного представлений.

## 2 ЛАБОРАТОРНЫЕ РАБОТЫ

### **Лабораторная работа № 1. Знакомство со средой моделирования ARIS. Построение организационной структуры предприятия и функциональной модели.**

**Цель работы** – знакомство с пользовательским интерфейсом программного пакета ARIS Toolset, изучение принципов построения организационной модели предприятия и «дерева функций» отдельного подразделения.

#### **Краткие теоретические сведения.**

##### *Модель оргструктуры предприятия.*

Организационная структура наиболее легко поддается моделированию, поскольку она в большинстве случаев четко определена и понятна. Рассматривая ее, следует различать организацию структуры и организацию процедур компании. Организационная структура включает правила, позволяющие произвести статичное структурирование подразделений компании. Организация процедур основана на правилах, отвечающих задачам, которые должны выполняться. Структура, ориентированная на задачи (в смысле распределения функций между исполнителями), в архитектуре ARIS представляется с помощью моделей управления. Таким образом, организационная схема является компонентой, которая позволяет анализировать организационную структуру компании.

Анализ организационной структуры должен проводиться с целью выявления:

- обоснованного количества уровней иерархии;
- наличия более чем 5-6 подчиненных подразделений у одного руководителя;
- наличия малого количества подчиненных у одного руководителя;
- подчинения одних и тех же звеньев (сотрудников) различным руководителям.

Для анализа организационной структуры следует использовать схемы организационной структуры, представленные в произвольном формате, а также положения об отделениях, отделах, группах и т.д. На основе этой информации формируется организационная структура в виде организационной схемы ARIS.

Организационная схема описывает организационные единицы разного уровня и их взаимосвязи. Эта модель — одна из важнейших, так как она описывает субъекты, которые определяют выходы и входы потоков ресурсов предприятия.

В модели организационной структуры целесообразно показывать:

- подразделения предприятия;

- наименование должности и фамилии руководителей подразделений;
- физическое местоположение отделов на предприятии.

Моделирование организационной структуры — стартовая точка в создании топологии компьютерной прикладной системы, которая, как предполагается, будет поддерживать организационную структуру наиболее оптимальным образом. Соединения сети и сетевые узлы, расположенные в определенных местах компании, являются главными элементами топологии сети прикладной системы. [7, 8]

Организационная схема содержит 20 различных объектов, названия и графические изображения которых сведены в таблицу 2 (см. [Таблица 2](#)), а возможные связи между элементами — в таблице 3 (см. [Таблица 3](#)). На рисунке 4 (см. [Рисунок 4](#)) показана панель инструментов «Моделирование» модуля *ARIS Designer* для организационной схемы.



Рисунок 4 - Панель инструментов для моделирования организационных схем

Таблица 2 - Объекты организационной схемы

Английское название объекта	Русское название объекта
Connection	Связь
Organizational unit type	Тип организационной единицы
Organizational unit type	Тип организационной единицы
Organizational unit	Организационная единица
Organizational unit	Организационная единица
Cost center	Стоимостной фактор
Position type	Тип сотрудника
Position	Должность
System organization unit type	Тип системной организационной единицы
System organization unit type	Тип системной организационной единицы
System organization unit	Системная организационная единица
System organization unit	Системная организационная единица
Person type	Тип сотрудника
Internal person	Штатный сотрудник
External person	Внештатный сотрудник
Group	Группа
Location	Расположение
Organizational chat	Организационная схема
Workstation	Рабочая станция
Position description	Описание должности



Таблица 3 - Связи объектов организационной схемы

Английское название связи	Русское название связи
Belongs to	Принадлежит
Can be constituent	Может являться частью
Can be disciplinary superior	Может быть непосредственным руководителем
Can be technical superior	Может быть техническим руководителем
Can Belongs to	Может принадлежать
Cooperates with	Взаимодействует с
Depicts	Описывает
Is assigned to	Связан (а) с
Is assigned 1:1 (1:n, n:m)	Имеет отношение 1:1 (1:n, n:m)
Is composed of	Состоит из
Is disciplinary superior to	Является непосредственным руководителем
Is located at	Располагается
Is managed by	Находится под управлением
Is of type	Относится к типу
Is organization manager for	Является организационным управляющим
Is position of	Является должностью
Is responsible for	Отвечает за
Is superior	Имеет в подчинении
Is technical superior to	Является техническим руководителем
Has assigned	Связан (а)с
Has member	Имеет в своем составе
Occupies	Занимает
Performs	Формирует
Substitutes for	Замещает
Subsumes	Содержит

Модель строится иерархически, от верхнего уровня структуры к нижнему. В модель верхнего уровня включаются самостоятельные подразделения, входящие в структуру организации. Каждое из них детализируется на более низкие уровни — уровни структурных подразделений.

Каждое структурное подразделение, в свою очередь, детализируется на структурные подразделения в его составе, которые изображаются на одной диаграмме.

Низшим уровнем является описание подразделений на уровне должностей — штатных единиц, занимаемых конкретными сотрудниками. При детализации моделей подразделений до уровня сотрудников целесообразно полностью указывать должность в составе подразделения. В случае если в одном подразделении имеется несколько одинаковых должностей, то они нумеруются.

Пример организационной схемы приведен на рисунке 5 (см. [Рисунок 5](#)).

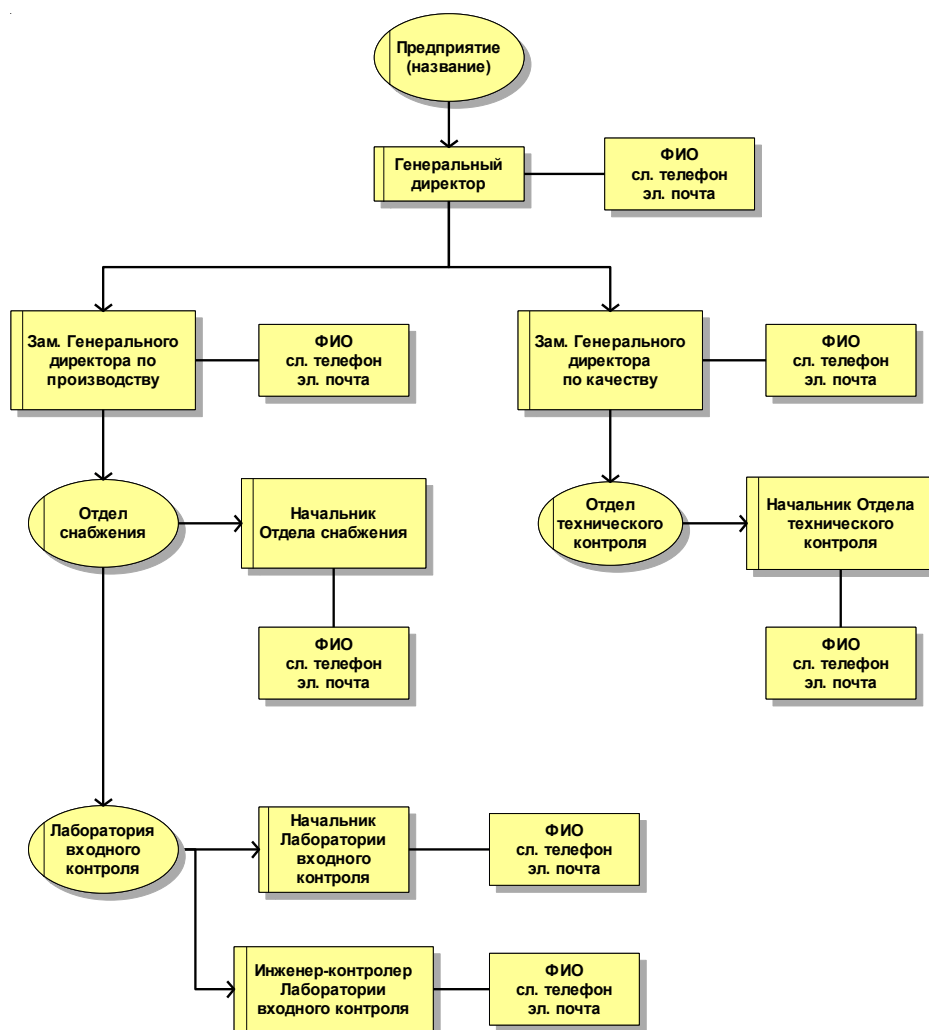


Рисунок 5 - Пример организационной схемы

*Дерево функций — Function Tree.*

Функция — описание элемента работы, образующего один логический этап в рамках процесса. В ARIS используется диаграмма «Дерево функций», посредством которой функции могут быть описаны с различными уровнями детализации. При этом функции представляются не обязательно в хронологическом порядке. [8]

На диаграммах функция обозначается прямоугольником зеленого цвета со скругленными краями (см. [Рисунок 6](#)). Возможные виды связей между функциями в модели «Дерево функций» приведены в таблице 4 (см. [Таблица 4](#))

Таблица 4 - Связи объектов диаграммы «Дерево функций»

Английское название связи	Русское название связи
Is execution-oriented superior	Подчиняется по способу выполнения
Is object-oriented superior	Подчиняется по объекту
Is process-oriented superior	Подчиняется по процессу

На самом верхнем уровне описываются наиболее сложные функции, представляющие собой отдельный бизнес-процесс или процедуру. Детализация функций образует иерархическую структуру их описаний. Пример дерева функций приведен на рисунке 6 (см. [Рисунок 6](#)).

Для более содержательного позиционирования отдельного уровня иерархии в общей структуре функций наряду с понятием «функция» могут быть использованы также другие термины: транзакция, процесс, подфункция, базовая функция (операция).

Разделение функций на элементы может происходить на нескольких иерархических уровнях. Базовые функции (см. [«Базовая функция»](#)) представляют самый нижний уровень в семантическом дереве функций.

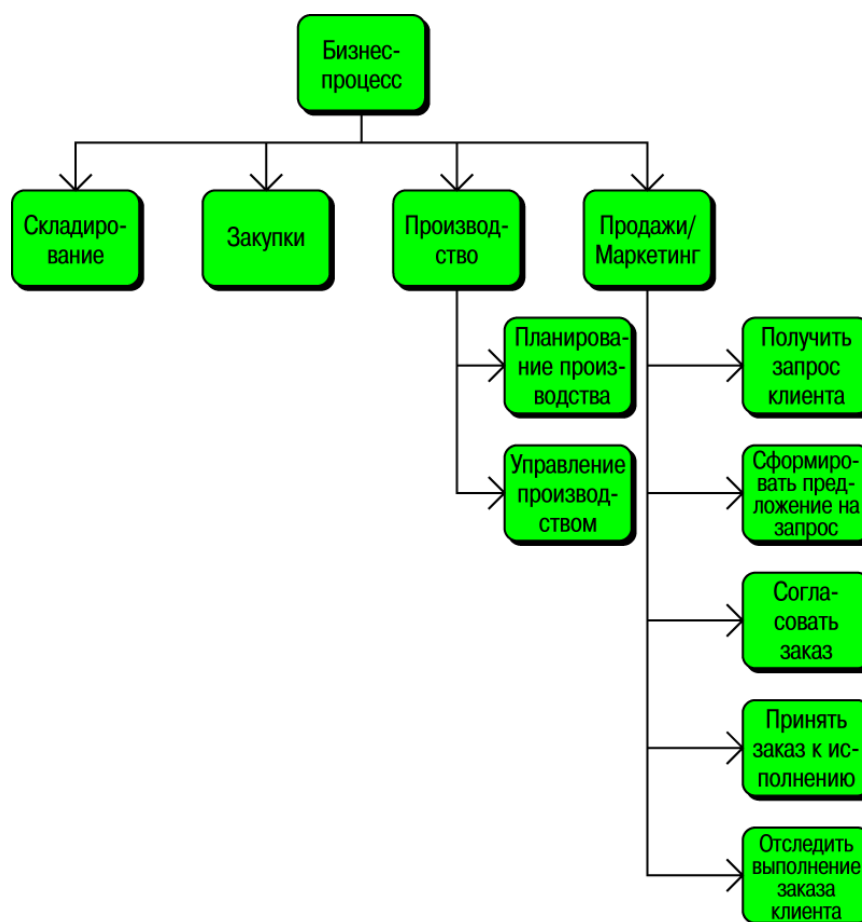


Рисунок 6 - Дерево функций

Функции объединяются в функциональное дерево в соответствии с различными критериями. Наиболее часто для этих целей используются такие критерии (см. [Рисунок 7](#)):

- объектно-ориентированный — обработка одного и того же объекта;
- процессно-ориентированный — принадлежность одному и тому же процессу;
- операционно-ориентированный — выполнение одинаковых операций.

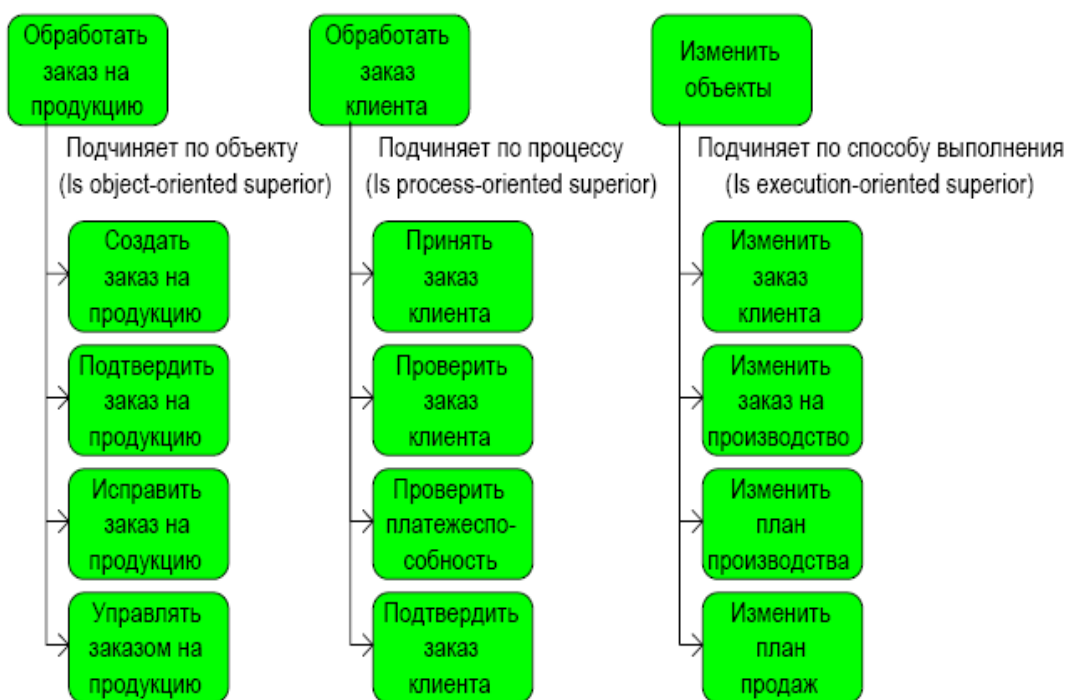


Рисунок 7 - Дерево функций: объектно-ориентированное (а); процессно-ориентированное (б); операционно-ориентированное (в)

Функция «Обработать заказ на продукцию» (см. [Рисунок 7](#)) описывает различные операции (создать, подтвердить, исправить и т.д.), которые выполняются над одним и тем же объектом, в данном случае — над объектом «Заказ на продукцию».

Если дерево функций используется в рамках моделирования бизнес-процесса, предпочтительнее применять метод, позволяющий построить процессно-ориентированное дерево. На рисунке 7 (б) (см. [Рисунок 7](#)) представлена процессно-ориентированная детализация функции (последовательность функций, составляющих процесс). В отличие от объектно-ориентированного разбиения при процессно-ориентированной детализации критерием служат операции, которые выполняются над различными объектами (заказ клиента, платежеспособность) в рамках одного бизнес-процесса.

При операционно-ориентированном подходе функция верхнего уровня декомпозируется на подфункции, каждая из которых выполняет ту же операцию, но с различными объектами. На рисунке 7(в) (см. [Рисунок 7](#)) приведена функция, выполняющая операцию «Изменить объекты». Функции могут принадлежать различным процессам и привлекаться к обработке различных объектов. Однако выполняемый ими тип операции над отдельными объектами всегда один и тот же.

Способ представления функций в виде дерева позволяет уменьшить степень сложности и является статичным описанием функции.

### Состав и особенности функциональной модели.

В данном разделе мы рассмотрим построение функциональной модели системы. Поскольку функциональная модель системы по итогам реинжиниринга (см. [«Реинжиниринг»](#)) является более полной, чем модель по итогам обследования, будем говорить о ней.

В общем виде функциональная модель (дерево функций) выглядит следующим образом (см. [Рисунок 8](#)):

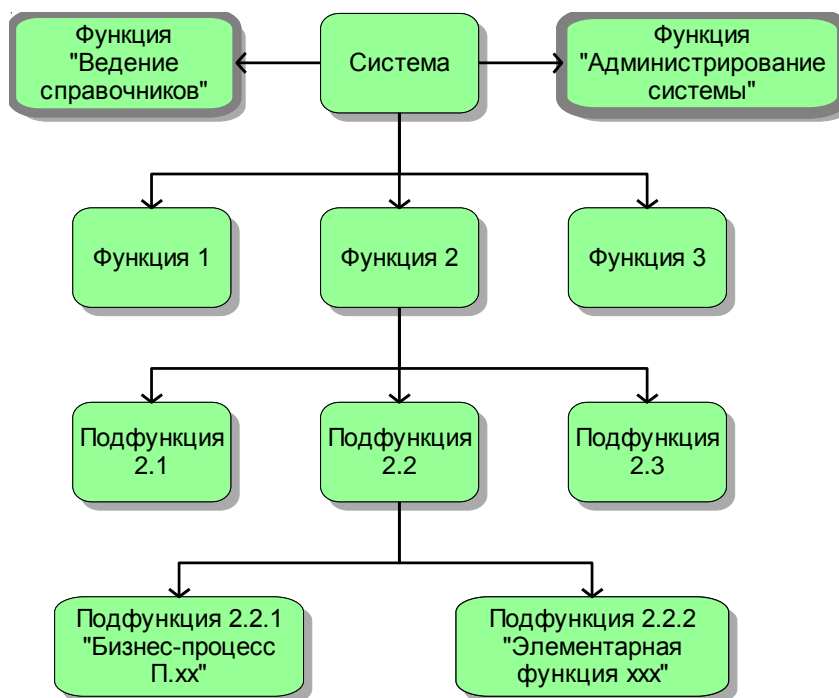


Рисунок 8 - Функциональная модель системы по результатам реинжиниринга

Элементы, обведенные жирной каймой, появляются на диаграмме в процессе реинжиниринга, как обязательные функции автоматизированной системы. Элементы двух первых уровней диаграммы должны быть окончательно определены еще на стадии обследования: на стадии реинжиниринга они могут только уточняться новыми элементами более низких уровней.

Самый нижний уровень диаграммы составляют элементарные функции, выполняемые одним человеком, или бизнес-процессы. Для каждого бизнес-процесса из функциональной модели должна быть впоследствии построена своя модель бизнес-процесса.

Необходимо особо отметить то обстоятельство, что при наличии на предприятии других автоматизированных решений, часто возникает задача если не полной их интеграции, то, по крайней мере, обмена данными между ними. Нам часто приходится сталкиваться с необходимостью организации обмена данными, и даже синхронизации баз данных внедряемого решения и АСУП предприятия. Функция организации обмена данными предлагаемого решения с другими автоматизированными

системами (при наличии в том необходимости) должна предусматриваться в функциональной модели, построенной по результатам реинжиниринга, отдельной функцией или несколькими отдельными подфункциями разных функций верхнего уровня. Практика показывает, что в случае внедрения PDM-системы, решение этой задачи требует специальной проработки [3].

### **Порядок выполнения работы.**

1. Ознакомьтесь с теоретической частью данного лабораторного практикума и заданием.
2. Проанализируйте исходные данные.
3. Постройте модели оргструктуры и дерева функций.
4. Продемонстрируйте созданную модель преподавателю
5. Оформите отчет.
6. Подготовьте ответы на контрольные вопросы.

### **Исходные данные задания.**

#### **Вариант №1.**

*Фрагмент оргструктуры.*

- ОАО «Трубострой» включает отдел главного конструктора и службу технической документации.
- Отдел главного конструктора подчиняется главному конструктору Байкову Б.Б.
- Служба технической документации подчиняется начальнику СТД Алехину А.А.
- Начальник СТД подчиняется главному инженеру Селезневу С.С.
- В отделе главного конструктора числятся заместитель главного конструктора Ковалев К.К. и конструкторы Самойлов С.С. и Володин В.В.
- В СТД числится оператор Соколов С.С.

*Фрагмент дерева функций.*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций:
  - Проектирование изделия:
    - Конструирование изделия;
    - Проектирование узла;
    - Детализовка узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства:
    - Разработка ТП;
    - Проектирование оснастки.
    - Изменение ТП;

- Аннулирование ТП;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Финансовая группа функций
- Группа функций по управлению персоналом
- Группа вспомогательных функций

## **Вариант №2.**

### *Фрагмент оргструктуры.*

- ОАО «Пневмотех» включает отдел главного конструктора и службу технической документации.
- Отдел главного конструктора подчиняется главному конструктору Малахову М.М.
- Служба технической документации подчиняется начальнику СТД Травину Т.Т.
- Начальник СТД подчиняется главному инженеру Конину К.К.
- В отделе главного конструктора числятся заместитель главного конструктора Астахов А.А. и конструктор Смуров С.С.
- В СТД числятся операторы Зорин З.З. и Волков В.В.

### *Фрагмент дерева функций.*

- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Группа вспомогательных функций

## **Вариант №3.**

### *Фрагмент оргструктуры*

- ОАО «Пневоавтомат» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия входит представитель заказчика (внештатный сотрудник) Воеводин В.В.
- Отдел главного конструктора подчиняется главному конструктору Лапину Л.Л.
- Служба технической документации подчиняется начальнику СТД Кострову К.К.
- Начальник СТД подчиняется главному инженеру Панину П.П.

- В отделе главного конструктора числятся заместитель главного конструктора Кузьмин К.К. и конструкторы Капитонов К.К. и Теркин Т.Т.
- В СТД числятся операторы Прохоров П.П. и Солдатов С.С.

*Фрагмент дерева функций*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства;
    - Разработка ТП;
    - Проектирование оснастки.
    - Изменение ТП;
    - Аннулирование ТП;

**Вариант №4.**

*Фрагмент оргструктуры*

• ОАО «Трубкомплект» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия входит представитель заказчика (внештатный сотрудник) Копытин К.К.

• Отдел главного конструктора подчиняется главному конструктору Молокову М.М.

• Служба технической документации подчиняется начальнику СТД Славину С.С.

• Начальник СТД подчиняется главному инженеру Панарину П.П.

• В отделе главного конструктора числятся заместитель главного конструктора Востряков В.В. и конструкторы Ломаков Л.Л. и Паршин П.П.

• В СТД числятся операторы Говоров Г.Г. и Стропилов С.С.

*Фрагмент дерева функций*

- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства;
    - Разработка ТП;



- Проектирование оснастки.
- Изменение ТП;
- Аннулирование ТП;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.

### **Вариант №5.**

#### *Фрагмент оргструктуры*

- ОАО «Автоматика» включает отдел главного конструктора и службу технической документации.
- Отдел главного конструктора подчиняется главному конструктору Котову К.К.
- Служба технической документации подчиняется начальнику СТД Мазину М.М.
- Начальник СТД подчиняется главному инженеру Багрову Б.Б.
- В отделе главного конструктора числятся заместители главного конструктора Юхов Ю.Ю., Тарасов Т.Т. и конструктор Аверьянов А.А.
- В СТД числится оператор Колосова А.А.

#### *Фрагмент дерева функций*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Финансовая группа функций
- Группа функций по управлению персоналом
- Группа вспомогательных функций

### **Вариант №6.**

#### *Фрагмент оргструктуры*

- ОАО «Пневматика» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия входит представитель заказчика (внештатный сотрудник) Калинин К.К.
- Отдел главного конструктора подчиняется главному конструктору Жарову В.В.
- Служба технической документации подчиняется начальнику СТД

Страхову С.С.

- Начальник СТД подчиняется главному инженеру Власову В.В.
- В отделе главного конструктора числятся заместители главного конструктора Пылин П.П., Ступин С.С. и конструктор Морозов М.М.

- В СТД числится оператор Соколов С.С.

*Фрагмент дерева функций*

- Маркетинговая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства;
    - Разработка ТП;
    - Проектирование оснастки.
    - Изменение ТП;
    - Аннулирование ТП;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Группа функций по управлению персоналом

### **Вариант №7.**

*Фрагмент оргструктуры*

- ОАО «Гидроавтомат» включает отдел главного конструктора и службу технической документации.

- Отдел главного конструктора подчиняется главному конструктору Марфину М.М.

- Служба технической документации подчиняется начальнику СТД Горлову Г.Г.

- Начальник СТД подчиняется главному инженеру Вдовину В.В.

- В отделе главного конструктора числятся заместитель главного конструктора Ковалев К.К. и конструкторы Астахов А.А. и Баев Б.б.

- В СТД числится оператор Птицына П.П.

*Фрагмент дерева функций*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;

- ✓ Изменение чертежа;
  - Изменение узла;
- Финансовая группа функций
- Группа функций по управлению персоналом
- Группа вспомогательных функций

### **Вариант №8.**

#### *Фрагмент оргструктуры*

- ОАО «Пневмогидроагрегат» включает отдел главного конструктора и службу технической документации.
- Отдел главного конструктора подчиняется главному конструктору Манину М.М.
- Служба технической документации подчиняется начальнику СТД Акимову А.А.
- Начальник СТД подчиняется главному инженеру Конину К.К.
- В отделе главного конструктора числятся заместитель главного конструктора Назаркин Н.Н. и конструктор Смурова С.С.
- В СТД числятся операторы Никитин Н.Н. и Волков В.В.

#### *Фрагмент дерева функций*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства;
    - Разработка ТП;
    - Проектирование оснастки.
    - Изменение ТП;
    - Аннулирование ТП;
- Финансовая группа функций
- Группа вспомогательных функций

### **Вариант №9.**

#### *Фрагмент оргструктуры*

- ОАО «Пневмосистемы» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия входит представитель заказчика (внештатный сотрудник) Обрезкин О.О.
- Отдел главного конструктора подчиняется главному конструктору Липину Л.Л.
- Служба технической документации подчиняется начальнику СТД

Кострову К.К.

- Начальник СТД подчиняется главному инженеру Орлову О.О.
- В отделе главного конструктора числятся заместитель главного конструктора Седов С.С. и конструкторы Капитонов К.К. и Павлов П.П.
- В СТД числятся операторы Прошкин П.П. и Солдатова С.С.

*Фрагмент дерева функций*

- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
  - Технологическая подготовка производства;
    - Разработка ТП;
    - Проектирование оснастки.
    - Изменение ТП;
    - Аннулирование ТП;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Финансовая группа функций
- Группа вспомогательных функций

### **Вариант №10.**

*Фрагмент оргструктуры*

• ОАО «Мехкомплект» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия входит представитель заказчика (внештатный сотрудник) Тюрин Т.Т.

• Отдел главного конструктора подчиняется главному конструктору Сидорову С.С.

• Служба технической документации подчиняется начальнику СТД Смирнову С.С.

• Начальник СТД подчиняется главному инженеру Панарину П.П.

• В отделе главного конструктора числятся заместитель главного конструктора Востряков В.В. и конструкторы Ломакова Л.Л. и Паршин П.П.

• В СТД числятся операторы Говорова Г.Г. и Стропилов С.С.

*Фрагмент дерева функций*

- Маркетинговая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Проектирование узла;
    - Детализация узла;

- ✓ Разработка чертежа;
- ✓ Изменение чертежа;
- Технологическая подготовка производства;
  - Разработка ТП;
  - Изменение ТП;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Группа функций по управлению персоналом
- Группа вспомогательных функций

### **Вариант №11.**

#### *Фрагмент оргструктуры*

- ОАО «Гидросистемы» включает отдел главного конструктора и службу технической документации.
- Отдел главного конструктора подчиняется главному конструктору Уткину М.П.
- Служба технической документации подчиняется начальнику СТД Мазину М.М.
- Начальник СТД подчиняется главному инженеру Федотову Б.Б.
- В отделе главного конструктора числятся заместители главного конструктора Юхов Ю.Ю., Тарасова Т.Т. и конструктор Аверьянов А.А.
- В СТД числится оператор Колосов А.А.

#### *Фрагмент дерева функций*

- Маркетинговая группа функций
- Логистическая группа функций
- Проектная группа функций
  - Проектирование изделия;
    - Конструирование изделия;
    - Проектирование узла;
    - Детализация узла:
      - ✓ Разработка чертежа;
      - ✓ Изменение чертежа;
    - Изменение узла;
- Производственная группа функций
  - Изготовление изделия;
  - Обеспечение качества.
- Финансовая группа функций
- Группа функций по управлению персоналом
- Группа вспомогательных функций

### **Вариант №12.**

#### *Фрагмент оргструктуры*

- ОАО «Пневмоавтомат» включает отдел главного конструктора и службу технической документации. Также в структуру предприятия

входит представитель заказчика (внештатный сотрудник) Филатов Ф.Ф.

- Отдел главного конструктора подчиняется главному конструктору Белову В.В.

- Служба технической документации подчиняется начальнику СТД Блохину Б.Б.

- Начальник СТД подчиняется главному инженеру Власову В.В.

- В отделе главного конструктора числятся заместители главного конструктора Волков В.В., Ступин С.С. и конструктор Морозова М.М.

- В СТД числится оператор Соколов С.С.

*Фрагмент дерева функций*

- Маркетинговая группа функций

- Логистическая группа функций

- Проектная группа функций

- Проектирование изделия;

- Конструирование изделия;

- ✓ Разработка чертежа;

- ✓ Изменение чертежа;

- Изменение узла;

- Технологическая подготовка производства;

- Разработка ТП;

- Проектирование оснастки.

- Изменение ТП;

- Аннулирование ТП;

- Группа функций по управлению персоналом

- Группа вспомогательных функций

### **Требования к отчёту по работе.**

Отчёт о работе должен содержать следующее:

1. Название работы;

2. Вариант;

3. Текст задания;

4. Распечатки разработанных моделей;

5. Выводы по проделанной работе.

### **Контрольные вопросы.**

1. Что понимается под детализацией (декомпозицией)?

2. Какую информацию позволяет описать модель оргструктуры?

3. Какую информацию позволяет описать модель «дерево функций»?

4. С какой целью проводится анализ организационной структуры предприятия?

5. В соответствии с различными критериями функции объединяются в функциональное дерево?

Опрос осуществляется в устной форме.

## **Лабораторная работа № 2. Построение диаграммы добавленной стоимости VAD (Value Added Chain Diagrams).**

**Цель работы** – изучение принципов построения моделей процессов верхнего уровня и их декомпозиции посредством диаграммы VAD (Value Added Chain Diagrams).

### **Краткие теоретические сведения.**

Диаграмма VAD (Value Added Chain Diagrams) описывает цепочку процессов, добавляющих стоимость, и применяется для описания процессов верхнего уровня.

Нотация VAD позволяет отобразить логическую связь между приоритетными процессами организации. Логическая связь отображается в виде двух отношений предыдущий-следующий или целое-часть. Точка зрения, с которой пишутся диаграммы – чаще всего, точка зрения руководства.

Например, модель VAD уровня взаимодействия «дирекция — филиал» может показывать, построение бизнес-процессов при распределении функций между исполнительной дирекцией холдинга и филиалами компании. На ней целесообразно отразить ответственные подразделения, уточнить результаты, создаваемые по ходу процесса, определить контрольные показатели. [8]

При построении этой диаграммы, как правило, выделяют 6-8 основных бизнес-процессов верхнего уровня. Затем описывают их в нотации VAD. Следующим этапом проводится декомпозиция полученных процессов верхнего уровня, при этом используется либо та же нотация VAD, либо eEPC (см. [Лабораторная работа №3](#)).




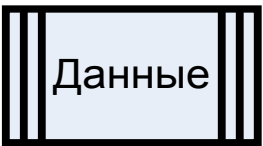
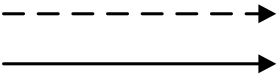
Основным объектом нотации VAD является объект «Добавочная ценность». Фактически это процесс или некоторая группа функций организации, которая служит для получения добавленной стоимости. Объекты соединяются между собой пунктирной стрелкой, которая имеет тип «предшественник». Этот тип связи показывает, что один процесс является предшественником другого. Для отображения обратных связей, или многоитерационных процессов применяются различные другие типы связей.

Между основными процессами могут быть отображены потоки материальных ресурсов и информации. Для описания этих потоков можно воспользоваться объектами типа «Данные» - для описания информации, и «Technical term» - для описания материальных потоков. Для описания инфраструктуры, необходимой для выполнения процесса могут быть выбраны типы объектов «Product/Service» и «Information service». Объекты «Орг. Единица», отображают организационные подразделения, выполняющие соответствующие процессы. В таблице 5 (см. [Таблица 5](#)) представлены основные графические элементы, используемые для

построения диаграммы VAD.

В связи с ограниченными возможностями программного пакета ARIS Toolset 7.0 по построению диаграммы VAD возможно использование для этой цели информационной среды Microsoft Office Visio. Следует отметить, что программный пакет Microsoft Office Visio позволяет также строить и другие виды диаграмм, например eEPC.

Таблица 5– Графические элементы для описания процесса с использованием нотации VAD

Обозначение	Наименование	Назначение
	Процесс	Процесс или группа функций организации для получения добавленной ценности
	Ресурс	Служит для отображения потоков материальных ресурсов
	Организационная единица	Организация или подразделение выполняющая процесс
	Ресурсная инфраструктура	Необходимая ресурсная инфраструктура для выполнения процесса
	Информационная инфраструктура	Необходимая информационная инфраструктура для выполнения процесса
	Данные, информация	Служит для отображения потоков информации необходимой для выполнения процесса
	Стрелки связи	Служат для соединения объектов диаграммы, могут иметь имя (тип)
используется предшественник исполняет Явл. входом Явл. выходом	Типы связей	Характерные типы связей

На рисунке 9 (см. [Рисунок 9](#)) изображена основная панель моделирования процессов, добавляющих стоимость, в программном пакете Microsoft Visio 2007.





Рисунок 9 – Панель моделирования диаграммы VAD

Объекты связываются между собой при помощи связей определенного типа. Например, информационный поток, отображаемый объектом «Данные», и являющийся входящим для процесса связан с ним при помощи стрелки типа «Явл. входом». Другим примером может служить тип связи «исполняет» между объектами «Добавочная ценность» и «Орг. единица». Тип связи «используется» показывает, что «Product/Service» используется процессом и т.д. Таким образом, в методологии важнейшим требованием является корректный выбор объектов и применение связей определенного типа.

Стрелки могут входить в любую сторону объекта «Добавочная ценность». Но важным является то, что в нотации VAD все обратные связи должны быть охарактеризованы или иметь ссылку.

В качестве примера рассмотрим этап проектирования самолета, выполненный с использованием графической нотации VAD (см. [Рисунок 10](#)).

На диаграмме представлено 5 бизнес-процессов, являющихся основными при проектировании самолета, связи между ними, входящие и выходящие ресурсы (документ, чертеж, 3D модель, оборудование, комплектующие и другое). Каждый из бизнес-процессов может быть подвергнут декомпозиции.

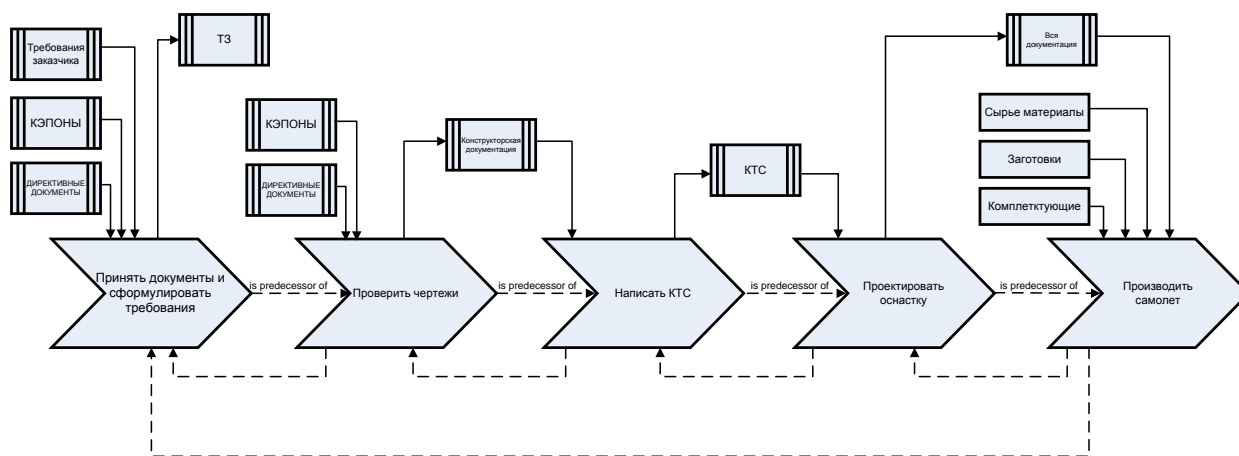


Рисунок 10 – Процесс проектирования самолета, выполненный в нотации VAD

### Порядок выполнения работы.

1. Ознакомьтесь с теоретической частью данного лабораторного практикума и заданием.

2. Проведите анализ предложенного бизнес-процесса, выявите его составляющие, разделите на 5-6 этапов (подпроцессов). Анализ проводится на основе имеющихся теоретических знаний, практического опыта и предложенных преподавателем источников.

3. Для каждого подпроцесса определите: входы и выходы, взаимосвязь между ними, используемые ресурсы, инфраструктуры (информационная и ресурсная), организационные единицы, данные.

4. Постройте модель процесса добавленной стоимости, используя диаграмму VAD.

5. Проведите декомпозицию одного из подпроцессов на 4-5 составляющих. Постройте модель декомпозиции, используя диаграмму VAD.

6. Продемонстрируйте созданные модели преподавателю.

7. Оформите отчет.

8. Подготовьте ответы на контрольные вопросы.

### Исходные данные задания.

Лабораторная работа предполагает проведение исследований предметной области и анализа предложенного (по вариантам) бизнес-процесса. Предметная область (см. [«Предметная область»](#)) – машиностроительное предприятие, выпускающее изделия авиационной техники.

С функциональной точки зрения модель машиностроительного предприятия можно представить в виде описания основных процессов деятельности: маркетинг, проектирование, производство, планирование, снабжение, формирование бюджетов, подготовка производства, контроль

качества, реализация, транспортировка, управление кадрами.

### **Вариант № 1.**

Бизнес-процесс – *маркетинг* (анализ рынка во всем его многообразии (покупатели и потребители, конкуренты, окружающая среда) и оценка результатов предпринятых действий; выбор целевых рынков, позиционирование и определение стратегии торговой марки, определение концепции продукта, ценообразование, выбор каналов сбыта, разработка стратегии коммуникации и продвижения товаров и выбор стратегии персонализации).

### **Вариант № 2.**

Бизнес-процесс – *конструкторское проектирование* (разработка технического задания (ТЗ), технического проекта, рабочих чертежей опытных образцов, разработка рабочих чертежей для производства).

### **Вариант № 3.**

Бизнес-процесс – *технологическая подготовка* (отработка изделия на технологичность; разработка технологических маршрутов и процессов; разработка специальной технологической оснастки; технологическое оснащение производства; техническое сопровождение изготовления опытного образца).

### **Вариант № 4.**

Бизнес-процесс – *подготовка производства* (изготовление специальной технологической и контрольной оснастки; расчеты количества и номенклатуры дополнительного оборудования, составление заявок и размещение заказов на оборудование; планирование работы вспомогательных цехов и служб, а также обслуживающих подразделений; расчет загрузки рабочих мест; расчеты оперативно-плановых нормативов, циклов, величин партий, заделов).

### **Вариант № 5.**

Бизнес-процесс – *производство* (формирование календарных планов выпуска продукции; движение товарно-материальных ценностей внутри производственного цикла; технологический процесс; изготовление изделия.)

### **Вариант № 6.**

Бизнес-процесс – *планирование* (формирование планов деятельности предприятия: формирование долгосрочных, краткосрочных и календарных планов продаж, выпуска продукции, закупок; планирование работы вспомогательных цехов и служб, а также обслуживающих подразделений; определение себестоимости и цены изделий.)

### **Вариант № 7.**

Бизнес-процесс – *снабжение* (обеспечение материалами, заготовками, деталями и узлами, получаемыми по кооперации, поиск поставщиков, заключение договоров).

### **Вариант № 8.**

Бизнес-процесс – *формирование бюджетов* (формирование целевых установок на заданный период планирования; разработка бюджета продаж; формирование плана производства в натуральных показателях, а также бюджетов производственных запасов и прямых затрат на материалы (сырье); расчет бюджета управленческих расходов, формирование бюджета по прибылям и убыткам, бюджета движения денежных средств и планового агрегированного баланса)

### **Вариант № 9.**

Бизнес-процесс – *контроль качества* (проведение измерений, экспертизы, испытаний или оценки параметров объекта и сравнение полученных величин с установленными требованиями к этим параметрам (показателями качества)).

### **Вариант № 10.**

Бизнес-процесс – *реализация* (поступление изготовленной продукции в народно-хозяйственный оборот с оплатой её по существующим ценам, выполнение договорных обязательств по поставкам продукции по ассортименту, качеству и срокам поставок; отслеживание изменений в структуре поставок продукции на экспорт; отслеживание и устранение причин отклонений в исполнении договорных обязательств).

### **Вариант № 11.**

Бизнес-процесс – *транспортировка* (комплектация партии сборных грузов на складе; дополнительная упаковка ряда грузов и создание оптимальных условий для перевозки с учетом специфики каждого груза; подготовка необходимой документации для транспортировки; выбор транспортного средства; доставка груза по заявленному адресу).

### **Вариант № 12.**

Бизнес-процесс - *управление кадрами* (подбор персонала, оценка персонала, расстановка кадров, обучение персонала).

### **Требования к отчёту по работе.**

Отчёт о работе должен содержать следующее:

1. Название работы;
2. Вариант;
3. Текст задания;

4. Распечатки разработанных моделей;
5. Выводы по проделанной работе.

### **Контрольные вопросы.**

1. Что понимается под процессами, добавляющими стоимость?
  2. Какую информацию позволяет описать диаграмма VAD?
  3. Какие графические примитивы используются при построении диаграммы VAD?
  4. С какой целью проводится анализ предметной области и бизнес-процессов?
  5. Что на диаграммах VAD обозначает обратная связь?
- Опрос осуществляется в устной форме.

### Лабораторная работа № 3. Построение цепочки процесса, управляемого событиями eEPC (extended Event Driven Process Chains).

**Цель работы** – изучение особенностей разработки моделей цепочки бизнес-процессов, управляемых событиями, посредством диаграммы eEPC (extended Event Driven Process Chains)

#### **Краткие теоретические сведения.**

*Особенности и правила построения модели бизнес-процесса.*

Модель бизнес процесса является самой важной и самой сложной с точки зрения построения. Бизнес-процессы необходимо проектировать в процессе реинжиниринга таким образом, чтобы они обеспечивали достижение поставленной цели при минимальной внутренней сложности.

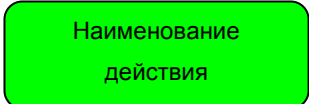
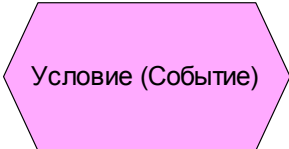
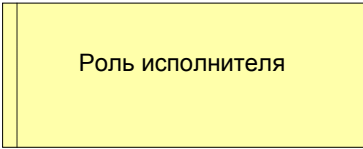
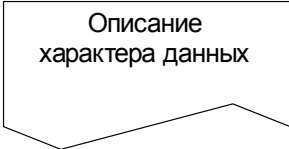



Событийная цепочка процесса (модель eEPC) предназначена для детального описания процессов, выполняемых в рамках одного подразделения, несколькими подразделениями или конкретными сотрудниками. Она позволяет выявлять взаимосвязи между организационной и функциональной моделями. Модель eEPC отражает последовательность функциональных шагов (действий) в рамках одного бизнес-процесса, которые выполняются организационными единицами, а также ограничения по времени, налагаемые на отдельные функции. [7]

Для каждой функции могут быть определены начальное и конечное события, ответственные исполнители, материальные и документарные потоки, сопровождающие модель, а также проведена декомпозиция на более низкие уровни (подфункции и т.д.). Модель eEPC является наиболее информативной и удобной при описании деятельности подразделений организации. Панель инструментов для создания модели eEPC показана на рисунке 11 (см. [Рисунок 11](#)). Объекты eEPC и их названия приведены в таблице 6 (см. [Таблица 6](#)).



Рисунок 11 - Панель инструментов для создания моделей eEPC — событийных цепочек процессов

Таблица 6-Объекты eEPC

Объект-примитив	Назначение в модели
	<p>Наименование действия – краткое описание характера действия и его целей. Например: «Чтение сообщения». Используется элемент типа «Function».</p>
	<p>Условие – описание условия (условий) выполнения следующего действия (процесса). Событие – описание некоторого события, совершение которого является условием выполнения следующего действия (процесса). Используется элемент типа «Event».</p>
	<p>Наименование роли – наименование роли, которую исполнитель данного действия играет в данном потоке работ. В общем случае роль человека в потоке работ зависит от его должности, но необязательно с ней совпадает. Используется элемент типа «Person type».</p>
	<p>Внешние исходные данные для выполнения определенных действий (в т.ч. – бумажный документ). Описание характера данных – указание на назначение рассматриваемого блока данных («Данные о ...» или наименование документа). Используется элемент типа «Information carrier - document».</p>
<p>Логическая связка «И»</p> 	<p>Связывает между собой на диаграмме действия (процессы), все из которых при данных начальных условиях должны начинаться, или заканчиваться для достижения данных результирующих условий. Также связывает одновременно выполняющиеся условия.</p>
<p>Логическая связка «ИЛИ»</p> 	<p>Связывает между собой на диаграмме действия (процессы), все или некоторые из которых при данных начальных условиях могут начинаться, или заканчиваться для достижения данных результирующих условий. Также связывает условия, из которых могут выполняться одно или несколько.</p>
<p>Логическая связка «Исключающее «ИЛИ»</p> 	<p>Связывает между собой на диаграмме действия (процессы), только одно из которых при данных начальных условиях может начинаться, или заканчиваться для достижения данных результирующих условий. Также связывает условия, из которых выполняется только одно.</p>

*Правила построения структур.*

Процедурная последовательность функций в рамках бизнес-процессов отображается в виде цепочки процесса, где для каждой функции могут быть определены начальное и конечное события (см. [«Событие»](#)). События активизируют функции, т.е. передают управление от одной функции к другой. Они могут быть также результатом выполнения функций. В отличие от функций, которые имеют некоторую продолжительность, события происходят мгновенно.

Описание события должно содержать не только информационный объект («заказ»), но и описание изменения состояния («получен»). События переключают функции и могут быть результатом выполнения функции. Упорядочивание комбинации событий и функций в последовательность позволяет создать событийные цепочки процессов. С помощью этих диаграмм процедуры бизнес-процесса представляются как логические последовательности событий/функций.

Рекомендованы следующие правила расположения графических элементов на диаграмме eEPC:

- графические элементы процесса (последовательность событий и функций) располагаются сверху вниз;
- графические элементы, обозначающие исполнителей функций (сотрудников или подразделения) располагаются справа от функций;
- документы, используемые при выполнении функций, а также формируемые в результате выполнения функций, располагаются слева от функций.

#### *Этап потока работ.*

Поток работ бизнес-процесса делится на этапы. *Этап потока работ* – неразрывная последовательность действий, выполняемая одним Исполнителем. Этапы должны быть типизированы в зависимости от предметной области и отражены в соответствующих типовых моделях этапов. Начинать построение модели этапа в ARIS следует со стандартного условия «Начало этапа», с которым с помощью связи «И» могут быть связаны специальные условия начала этапа (см. [Рисунок 12](#)).

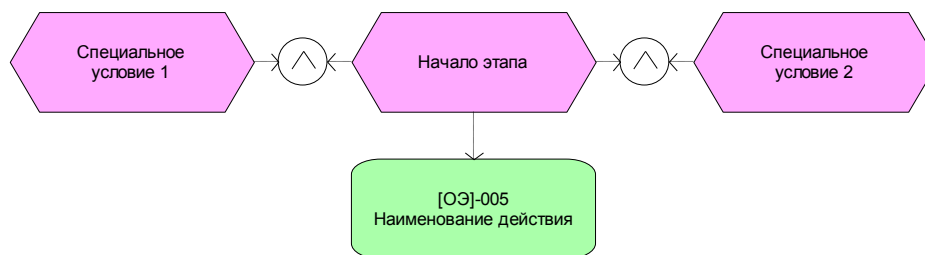


Рисунок 12-Начало модели этапа

Каждому действию следует сопоставить роль Исполнителя данного действия, исходные данные и результаты. В тех случаях, когда результаты предыдущего действия являются исходными данными для последующего действия, это должно быть явно отражено на диаграмме с помощью отношения «Является» (см. [Рисунок 13](#)).

Начальные условия и результаты каждого действия могут быть обязательными (имеющими место во всех случаях) и необязательными (имеющими место не во всех случаях) (см. [Рисунок 14](#)).





Рисунок 13- Последовательность действий

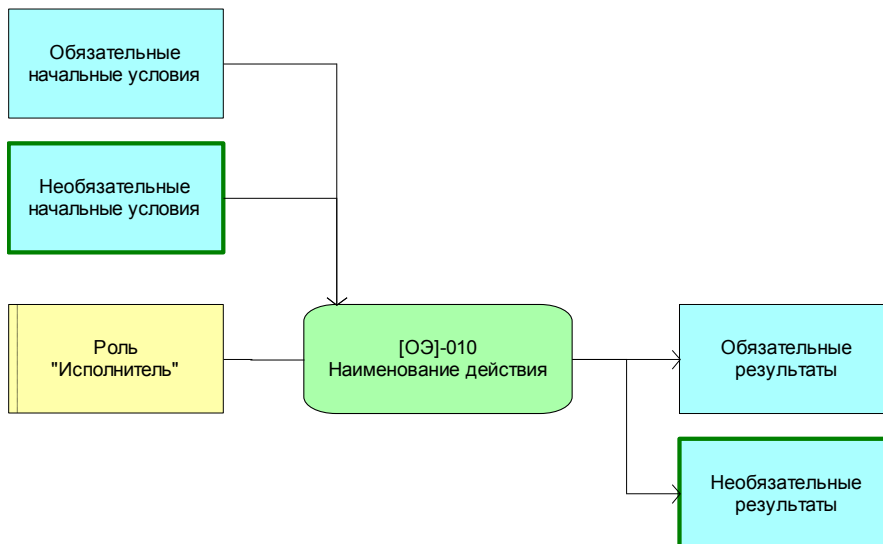


Рисунок 14 - Обязательные и необязательные условия и результаты

Заканчивать модель этапа следует стандартным условием «Завершение этапа», с которым с помощью связки «И» могут быть связаны специальные условия - результаты этапа (см. [Рисунок 15](#)).

Принимая во внимание некоторые особенности отображения на бумаге диаграмм, построенных в ARIS, следует всем элементам диаграмм назначать яркость цвета равно 200, что обеспечит удовлетворительное отображение диаграмм, на экране и на бумаге при черно-белой и цветной печати.

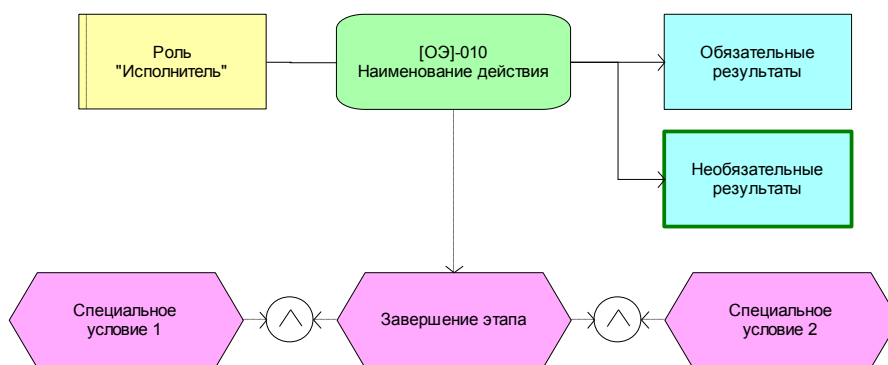


Рисунок 15- Завершение этапа

*Использование логических связей между действиями и условиями.*

В том случае, когда необходимо одновременно и параллельно выполнить несколько действий, или показать одновременное выполнение нескольких условий используется логическая связка «И» (см. [Рисунок 16](#)). Также связка «И» используется в тех случаях, когда несколько параллельно выполняемых действий должны закончиться одновременно для перехода к одному общему последующему действию (см. [Рисунок 17](#)).

Логическая связка «ИЛИ» должна использоваться в тех случаях, когда необходимо выбрать один или несколько вариантов действий или наборов условий из множества возможных (см. [Рисунок 18](#)).

Логическая связка «Исключающее ИЛИ» должна использоваться в тех случаях, когда надо выбрать только один вариант действий или одно условие из нескольких возможных (см. [Рисунок 19](#)).

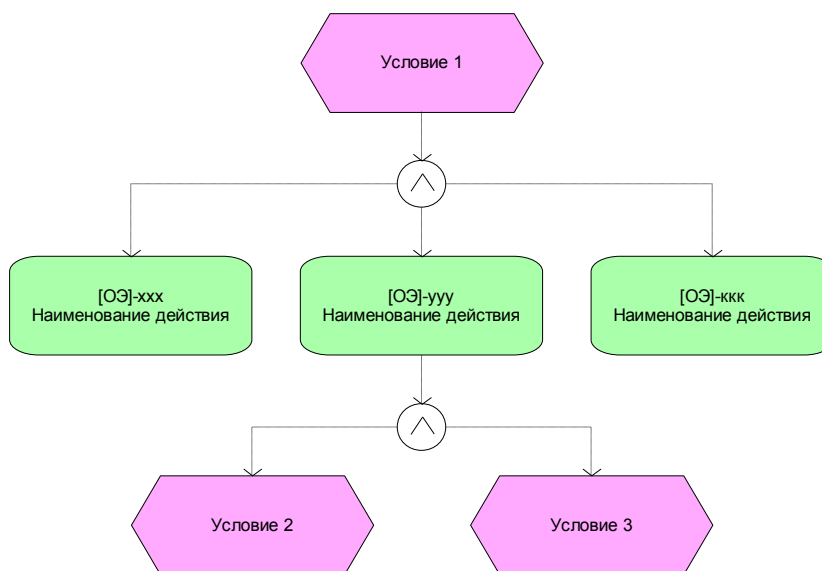


Рисунок 16 - Пример использования связки «И»

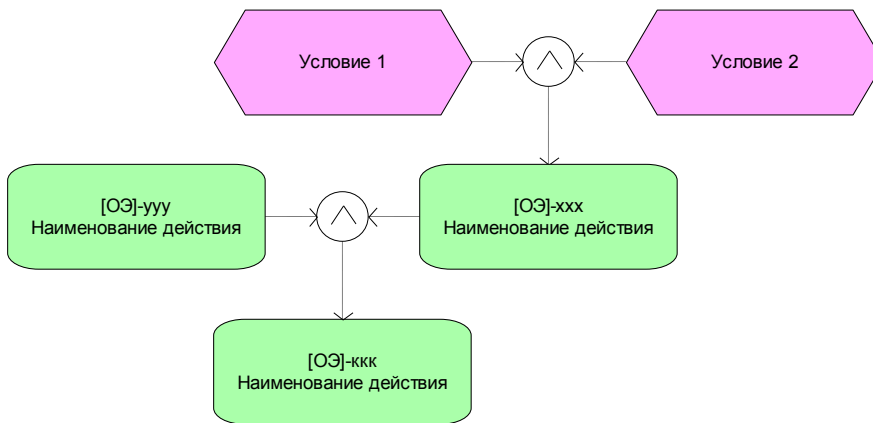


Рисунок 17- Пример использования связки "И"

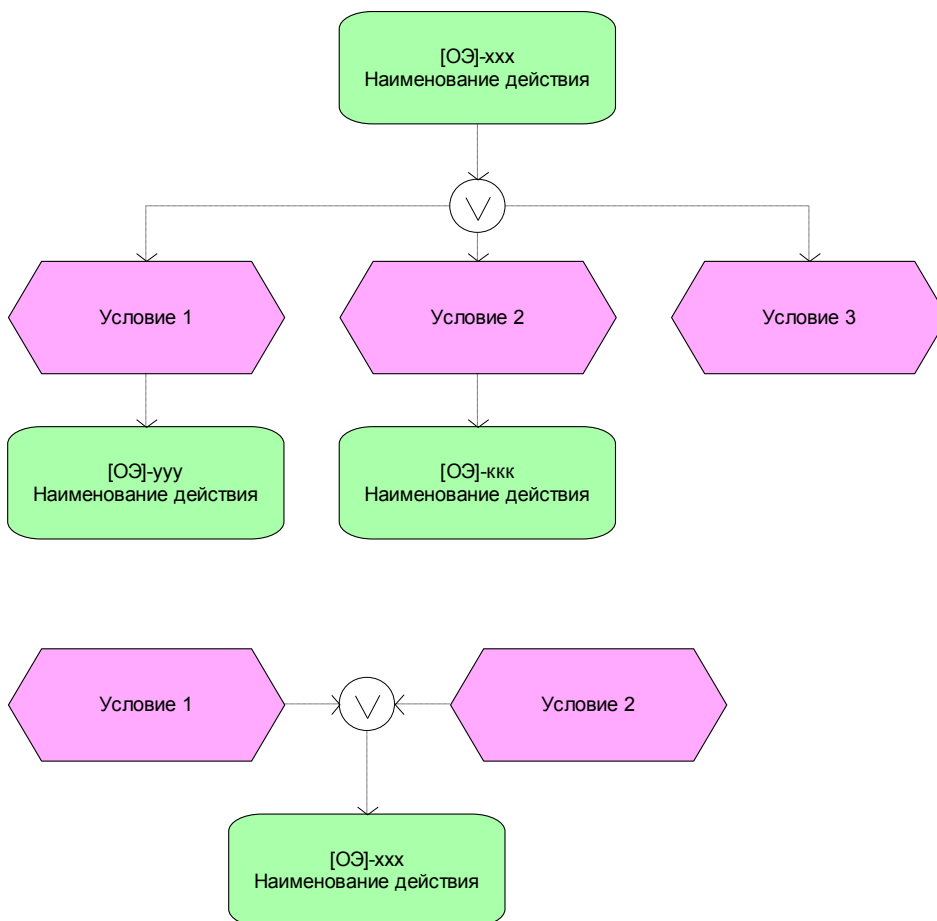


Рисунок 18-Примеры использования связки "ИЛИ"

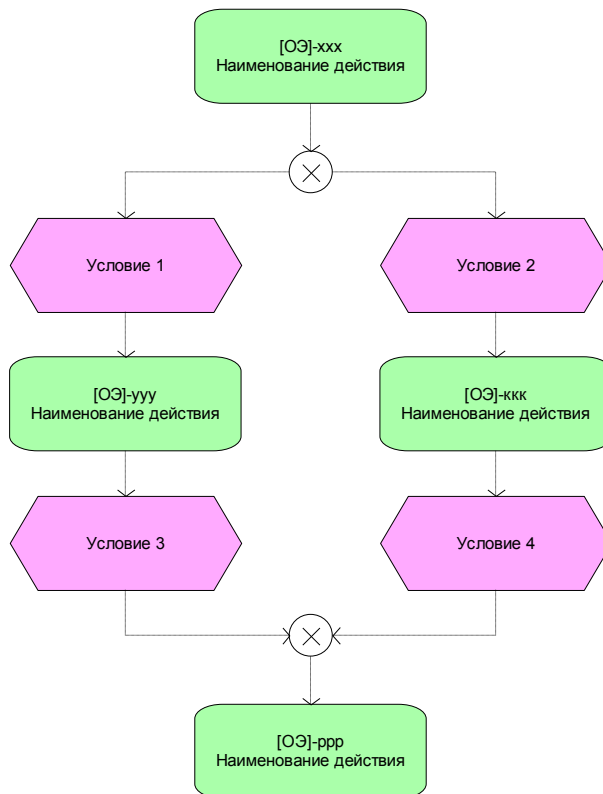


Рисунок 19- Пример использования связки "Исключающее ИЛИ"

*Создание сложных логических связей между действиями (условиями).*

С помощью логических связок при построении диаграмм можно создавать сложные логические связи между действиями или условиями, с помощью которых можно точнее описывать реальные бизнес-процессы с их особенностями.

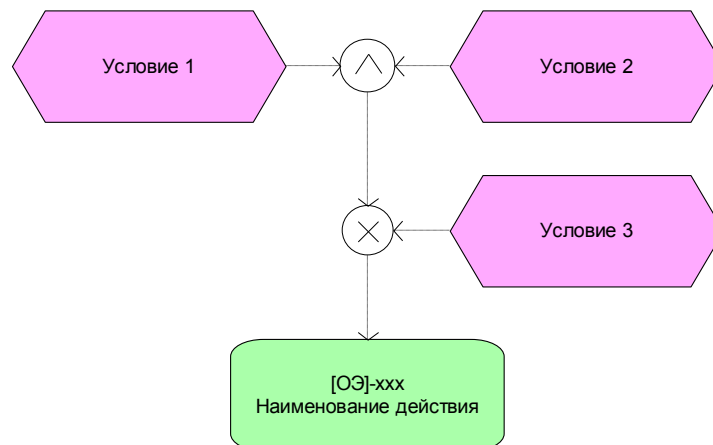


Рисунок 20- Пример сложной логической связи условий выполнения действия

На рисунке 20 (см. [Рисунок 20](#)) представлена ситуация, когда для выполнения действия необходимо либо одновременное выполнение Условий 1 и 2, либо выполнение Условия 3.

### *Разбиение диаграмм на фрагменты.*

Большие диаграммы, описывающие сложные потоки работ, целесообразно разбивать на отдельные взаимосвязанные фрагменты, с размерами, соответствующими возможностям текущего принтера.

Для этого используются метки перехода (см. [Рисунок 21](#)), расставляемые в момент ссылки на очередной фрагмент и в начале этого фрагмента. Возможно использование в диаграмме нескольких ссылок на один и тот же фрагмент.

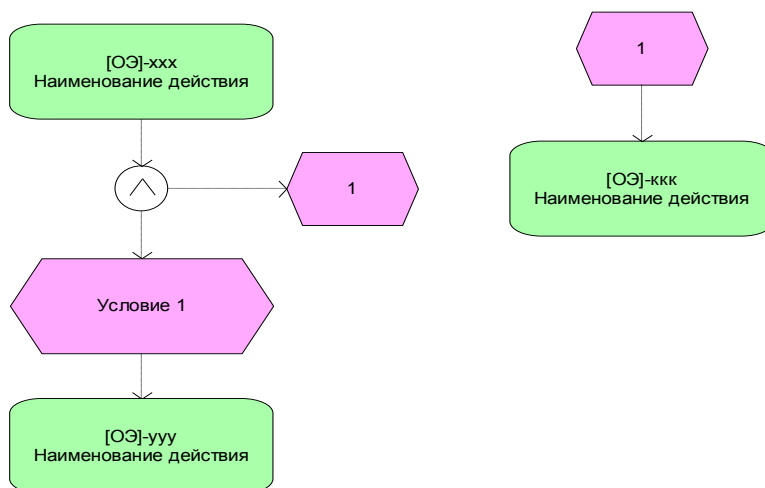


Рисунок 21 - Пример фрагментирования диаграммы

### *Детализация модели бизнес-процесса моделями этапов в ARIS.*

Все рекомендации, высказанные ранее по поводу моделирования этапа бизнес-процесса, справедливы и для бизнес-процесса в целом. Но так же, как модели назначения атрибутов расширяли информационную модель задачи, модели, описывающие поток работ в рамках одного этапа, будут расширять модель бизнес-процесса в целом. Каждому объекту, обозначающему этап в модели бизнес-процесса, с помощью функции *Assignments* должна быть поставлена в соответствие модель потока работ для этого конкретного этапа.

Таким образом, модель бизнес-процесса неизбежно приобретает двухуровневый иерархический характер. Как нетрудно заметить, модель бизнес-процесса строится на основе данных других моделей.

На рисунке 22 (см. [Рисунок 22](#)) представлен пример фрагмента модели бизнес-процесса, один из этапов которого детализируется своей собственной моделью.

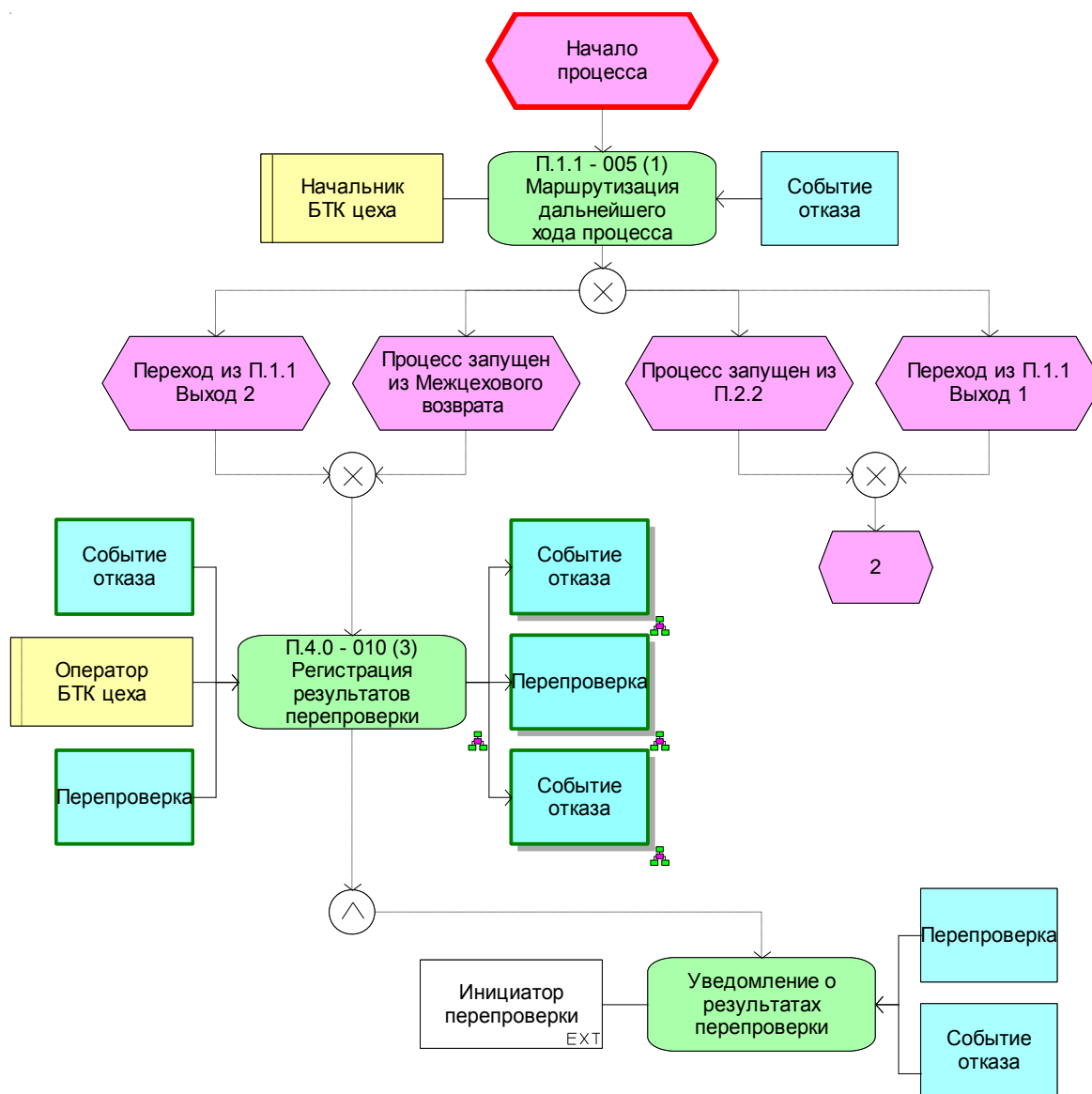


Рисунок 22-Пример фрагмента модели бизнес-процесса

Вид окна построения модели бизнес-процесса был приведен на рисунке 3 (см. [Рисунок 3](#)). Обратите внимание на богатство набора примитивов для построения этого типа диаграмм в ARIS (см. [Рисунок 11](#)).

### Порядок выполнения работы.

1. Ознакомиться с заданием и теоретической частью данных методических указаний.
2. Проанализировать исходные данные.
3. Построить проект бизнес-процесса в нотации eEPC. Каждый этап процесса должен выполняться носителем одной роли в Системе.
4. Продемонстрировать созданную модель преподавателю.
5. Оформить отчет.

## Исходные данные задания

### **Вариант №1.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. главного конструктора выдает задание конструктору на проведение изменений чертежа;
2. Выбранный конструктор производит изменения чертежа изделия;
3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Вариант №2.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;
2. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
3. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Начальник службы технической документации выдает задание оператору службы технической документации на проведение утвержденных изменений;
5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Вариант №3.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. главного конструктора выдает задание конструктору на проведение изменений чертежа;
2. Конструктор производит изменения чертежа изделия;
3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Произведенные изменения чертежа утверждаются главным

инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

5. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;

6. Начальник службы технической документации выдает задание оператору службы технической документации на проведение утвержденных изменений;

7. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.

8. Процесс завершен, утвержденные изменения вступают в силу.

#### **Вариант №4.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. Главного конструктора выдает задание конструктору на проведение изменений чертежа;

2. Конструктор производит изменения чертежа изделия;

3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

4. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;

5. Начальник службы технической документации производит процедуру проведения утвержденных изменений чертежа.

6. Процесс завершен, утвержденные изменения вступают в силу.

#### **Вариант №5.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;

2. Главный конструктор выдает задание на согласование произведенных изменений одному из своих заместителей;

3. Произведенные изменения чертежа согласовываются с заместителем главного конструктора, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

4. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

5. Начальник службы технической документации производит процедуру проведения утвержденных изменений чертежа.

6. Процесс завершен, утвержденные изменения вступают в силу.



### **Вариант №6.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;
2. Главный конструктор выдает задание на согласование произведенных изменений одному из своих заместителей;
3. Произведенные изменения чертежа согласовываются с заместителем главного конструктора, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;
5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Вариант №7.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. главного конструктора выдает задание конструктору на проведение изменений чертежа;
2. Выбранный конструктор производит изменения чертежа изделия;
3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Вариант №8.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;
2. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
3. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Начальник службы технической документации выдает задание оператору службы технической документации на проведение утвержденных изменений;

5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.

6. Процесс завершен, утвержденные изменения вступают в силу.

#### **Вариант №9.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. главного конструктора выдает задание конструктору на проведение изменений чертежа;

2. Конструктор производит изменения чертежа изделия;

3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

4. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

5. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;

6. Начальник службы технической документации выдает задание оператору службы технической документации на проведение утвержденных изменений;

7. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.

8. Процесс завершен, утвержденные изменения вступают в силу.

#### **Вариант №10.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Зам. Главного конструктора выдает задание конструктору на проведение изменений чертежа;

2. Конструктор производит изменения чертежа изделия;

3. Произведенные изменения чертежа согласовываются с главным конструктором, в случае наличия замечаний чертеж возвращается на доработку к конструктору;

4. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;

5. Начальник службы технической документации производит процедуру проведения утвержденных изменений чертежа.

6. Процесс завершен, утвержденные изменения вступают в силу.

#### **Вариант №11.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;
2. Главный конструктор выдает задание на согласование произведенных изменений одному из своих заместителей;
3. Произведенные изменения чертежа согласовываются с заместителем главного конструктора, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Произведенные изменения чертежа утверждаются главным инженером, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
5. Начальник службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Вариант №12.**

Процесс «Изменение чертежа изделия» состоит из следующей последовательности действий:

1. Конструктор производит изменения чертежа изделия;
2. Главный конструктор выдает задание на согласование произведенных изменений одному из своих заместителей;
3. Произведенные изменения чертежа согласовываются с заместителем главного конструктора, в случае наличия замечаний чертеж возвращается на доработку к конструктору;
4. Представитель заказчика должен одобрить изменения, в случае его несогласия с проведенными изменениями чертеж возвращается на доработку к конструктору;
5. Оператор службы технической документации производит процедуру проведения утвержденных изменений чертежа.
6. Процесс завершен, утвержденные изменения вступают в силу.

### **Требования к отчёту по работе.**

Отчёт о работе должен содержать следующее:

1. Название работы;
2. Вариант;
3. Текст задания;
4. Распечатки разработанных моделей;
5. Выводы по проделанной работе.

### **Контрольные вопросы.**

1. Каким образом происходит связывание отдельных диаграмм в единую модель?
2. Какую информацию позволяет описать модель процесса eEPC?
3. Каким образом задается логика выполнения процесса?
4. Как в модели процесса используется элемент «событие»?
5. Что понимается под ролью исполнителя?

## Лабораторная работа №4. Построение информационной модели eERM (Extended entity -relationship model)

**Цель работы** – изучение основ построения информационной модели посредством диаграммы eERM (Extended entity -relationship model)

### Краткие теоретические сведения.

*Расширенная модель «сущность - отношение» — Extended entity - relationship model (eERM).*

В течение последних лет оригинальная модель Чена (см. [«Диаграмма Чена»](#)) была значительно расширена. Модель eERM — расширенная модель «сущность - отношение» как раз и представляет собой расширение классической модели Чена.

Данная модель играет существенную роль при описании данных в архитектуре ARIS. Модель данных eERM используется для создания информационных моделей, отражающих структуру информации, которая обрабатывается в бизнес-процессах организации. [7, 8]

Объекты модели eERM приведены в таблице 7 (см. [Таблица 7](#)), а возможные связи — в таблице 8 (см. [Таблица 8](#)). Используемая при конструировании панель инструментов показана на рисунке 23 (см. [Рисунок 23](#)).

*Тип сущности* используется для представления материальных и нематериальных типов объектов.

*Тип отношений* необходим для представления устойчивых типов отношений между типами объектов при моделировании.

*Кластер* является совокупностью некоторого количества связанных типов сущностей и предназначен для представления сложных объектов, а также для указания структуры информации, изменение которой фиксируется в событии.

Объект *обобщение* предназначен для указания операции *обобщения/специализации* некоторых типов объектов. Обобщаемый тип сущности присоединяется к вершине треугольника, обобщающие/специализирующие типы сущностей присоединяются к основанию треугольника.



Рисунок 23 - Панель инструментов для создания моделей eERM

Таблица 7 - Объекты eERM-модели

Английское название связи	Русское название связи
Connection	Связь
Entity type	Тип сущности
Relationship type	Тип отношений
Reinterpretation relationship type	Переопределенный тип отношений
Cluster	Кластер
Generalization	Обобщение
D attribute (ERM)	Описательный атрибут ERM
K attribute (ERM)	Ключевой атрибут ERM
EK attribute (ERM)	Ключевой наследуемый атрибут ERM
ERM domain	ERM-домен
Attribute type group	Группа типов атрибутов
Enumeration	Перечисление
Unit	Элемент
Unit number	Номер элемента
Enumeration list occurrence	Экземпляр списка

Таблица 8 - Связи объектов в модели eERM

Английское название связи	Русское название связи
Belongs to	Принадлежит
Defines	Определяет
Differentiates according to value of	Дифференцируется в зависимости от
Has	Имеет
Has link to	Имеет связь с
Is attribute type group	Является группой типов атрибутов для
Is describing for	Описывает
Is primary key for	Является первичным ключом для
Is subtype of	Является подтипом
Is supertype of	Является супертипом

*Описательный атрибут* предназначен для описания свойств моделируемых объектов.

*Ключевой атрибут* служит для указания свойства объекта, значение которого однозначно определяет экземпляр типа объекта реального мира. Ключевые атрибуты могут быть только у сущностей. Связи не могут иметь ключевых атрибутов. Ключевой атрибут может быть простым или составным. Простой атрибут состоит из одного атрибута. Составной состоит из нескольких атрибутов, т.е. только совокупность значений этих атрибутов однозначно определяет экземпляр типа объекта реального мира. ARIS не имеет средств для поддержки составного идентификатора.

*ERM-домен* используется для указания множества значений атрибута, т.е. его области значений. С помощью одного домена могут быть определены несколько атрибутов.

*Группа типов атрибутов* предназначена для группировки атрибутов по некоторому признаку. Она может служить для указания составного идентификатора.

Объект *перечисление* применяется для указания значения атрибута, когда область его значений не соответствует наименованию атрибута, например, для уточнения смысла таких атрибутов, как *характер, тип производства, тип руководства*.

Опираясь на ряд различных подходов к расширению модели «сущность-отношение», можно выделить четыре основных оператора проектирования: *классификацию, обобщение, агрегацию и группировку*.

Операторы *проектирования* обеспечивают формальную поддержку процесса создания модели данных. Их применение гарантирует систематический подход и дает возможность тому, кто знает существующую структуру данных, понять суть процесса проектирования. Анализ условий выполнения бизнес-процессов с точки зрения их структур данных помогает разработчикам структурировать известные условия, базируясь на новом представлении, а также создавать новые отношения, не рассматривавшиеся до сих пор.

*Классификация.* При помощи классификации объекты (сущности) одного и того же типа идентифицируются и ассоциируются в соответствии с некоторым признаком (типом сущностей). Один объект идентичен другому, если он описан теми же свойствами (атрибутами).

*Обобщение/Специализация.* При обобщении аналогичные типы объектов группируются под одним, более старшим типом объекта. Как показано на рисунке 24(а) (см. [Рисунок 24](#)), тип сущности *клиент* и тип сущности *поставщик* объединены под общей концепцией «Участник процесса». Свойства, описанные атрибутами и являющиеся общими для обоих исходных объектов, присваиваются обобщенному объекту. Таким образом, остается описать только такие атрибуты, которые отличны от атрибутов исходных типов объектов. Образование нового типа сущности *участник бизнес-процесса* представляется графически в виде треугольника, который также является отношением и означает «есть».

Под *специализацией* понимается разделение некоторого общего множества (например, объектов) на подмножества. Оператор специализации является инверсным по отношению к оператору обобщения (пример: сущность *участник процесса* разделяется на сущности *клиент* и *поставщик*). Специализированные объекты наследуют свойства обобщенных объектов. Кроме наследования, специализированные типы объектов могут обладать также собственными атрибутами. Графически специализация и обобщение представляются одинаково. По этой причине соединения на рисунке не отображаются стрелками, указывающими направление.

Специализация в первую очередь поддерживает подход к структуре данных «сверху вниз», обобщение же используется при подходе «снизу вверх».

В рамках специализации полнота и степень разделяемости на поднаборы могут определяться в процессе их создания.

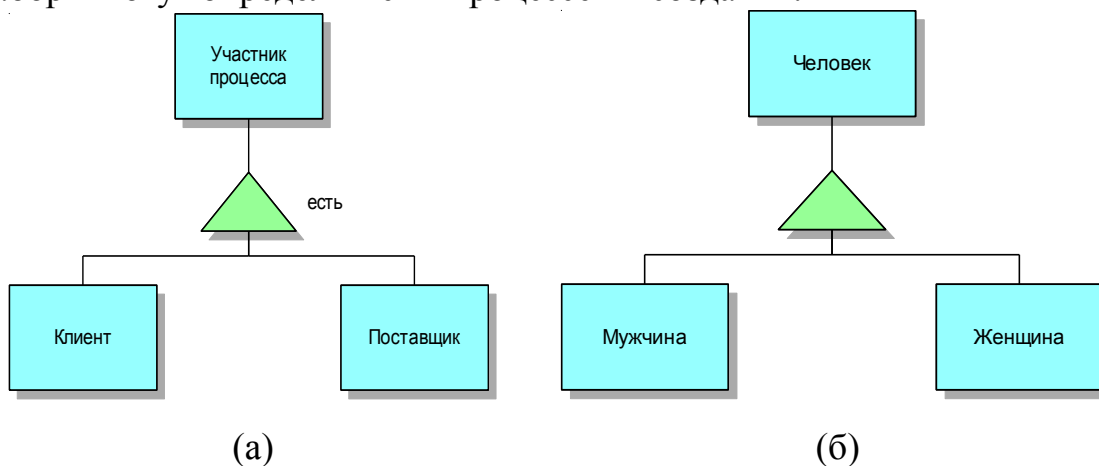


Рисунок 24- Обобщение/Специализация (а) и полная специализация (б)

О неразделяемых подмножествах будем говорить в том случае, когда экземпляр одного объекта может быть частью обоих подмножеств.

Для приведенного выше примера это означает, что некоторый клиент в то же самое время может быть и поставщиком. Если экземпляр ассоциируется только с одним подмножеством, то множества являются разделяемыми.

Когда все специализированные типы объектов, возможные для одного критерия специализации, входят, в состав одного обобщенного типа объекта, говорят о полной специализации. Например, тип сущности *человек* может быть разделен на типы сущностей *мужчина* и *женщина* (см. [Рисунок 24](#) (б)). Специализация полностью определена, если в качестве критерия специализации выбран «пол».

Комбинация этих критериев приводит в результате к следующим четырем случаям, которые выделяются, чтобы более точно определить оператор *обобщение/специализация*:

- раздельная/полная;
- раздельная/неполная;
- нераздельная/полная;
- нераздельная/неполная.

*Агрегация.* Агрегация описывает формирование нового типа объекта с помощью комбинации существующих типов объектов. В этом контексте новый тип объекта может нести новые свойства.

В модели eERM агрегация представляется как формирование типов отношений (см. [Рисунок 25](#)). Агрегация типов сущностей *заказ на производство* и *маршрутизация* создает новый объект *маршрутизация заказа*.

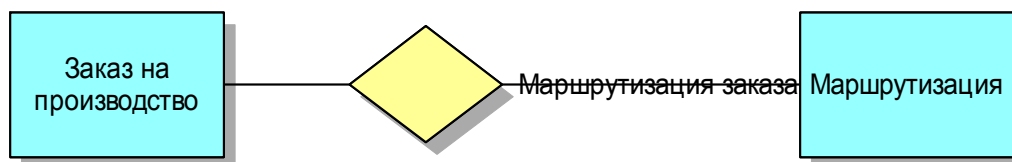


Рисунок 25-Пример агрегации

Оператор агрегации также применяется к отношениям. В этом случае существующий тип отношения, называемый переопределенным, трактуется как тип сущности и, таким образом, может стать отправной точкой для создания другого, нового отношения (см. [Рисунок 26](#)).

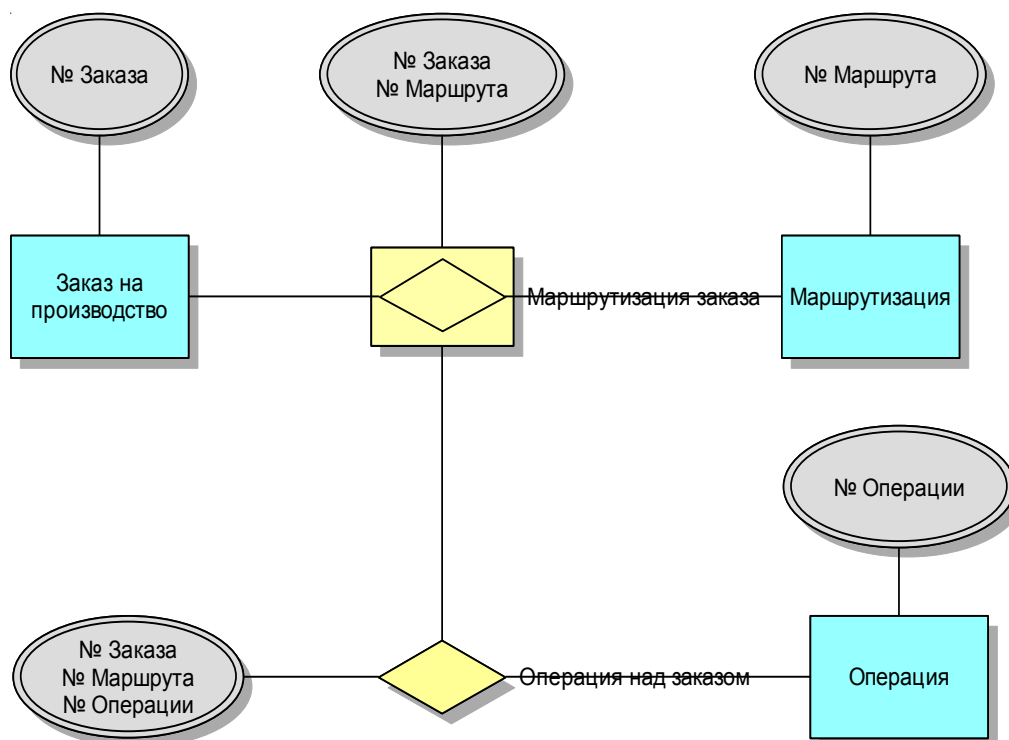


Рисунок 26-Агрегация с переопределением типа отношений

Первая агрегация используется для формирования типа отношения *маршрутизация заказа* из типов сущностей *заказ на производство* и *маршрутизация*. Ключевые атрибуты *№ заказа* и *№ маршрута* образуют сложный ключевой атрибут для отношения к переопределенному типу отношений *маршрутизация заказа*. Теперь многочисленные операции могут быть связаны непосредственно с ним. Следовательно, отношение *операция над заказом* формируется между типами отношений *маршрутизация заказа* и типом сущности *операция*. Поскольку отношения могут создаваться только между типами сущностей, необходимо, чтобы первоначальный тип отношения *маршрутизация заказа* был интерпретирован как тип сущности. На рисунке 26 (см. [Рисунок 26](#)) это иллюстрируется с помощью ромба, заключенного в прямоугольник.

Этот переопределенный тип отношения трактуется как нормальный тип сущности. Процедура создания типа отношения из нескольких



существующих типов сущностей отображается с помощью линий, направленных к вершинам ромба. Однако линии, означающие новые отношения по переопределенным типам отношений, подходят к краям прямоугольника, окружающим ромб, а не к самому ромбу (см. [Рисунок 26](#)).

Несмотря на то, что сложный ключ можно заменить более простым, целесообразнее использовать сложные ключи, поскольку они позволяют лучше отслеживать процесс создания модели данных.

В модели eERM сложный структурный элемент разделяется таким образом, чтобы образовать прозрачную структуру. Для лучшего понимания и для того, чтобы не нарушить стройность концепции, вводят сложные объекты в виде кластеров данных.

Кластер данных описывает логическую модель некоторых типов сущностей и отношений в модели данных. Это необходимо для описания сложных объектов. Кластеры данных содержат не только типы сущностей и отношений, но и другие кластеры данных. В отличие от типов сущностей и отношений кластеры данных могут быть легко встроены в иерархию, что позволяет в процессе создания модели данных поддерживать подход «сверху вниз». Использование кластеров данных может быть полезно и при объединении моделей в ходе проектирования «снизу вверх».

*Группировка.* В процессе группировки формируются группы, составленные из элементов некоторого множества сущностей.

В примере на рисунке 27 (см. [Рисунок 27](#)) все составные части сущности *оборудование* объединяются в *группу оборудования*. Группа оборудования — это независимый объект, который может быть описан более точно с помощью дополнительных атрибутов (наименование группы оборудования, число деталей оборудования), которые не характеризуют отдельные части оборудования. Другим примером может служить группировка рабочих мест в отделы или объединение элементов, связанных с обработкой заказа, в группу *заказ*.

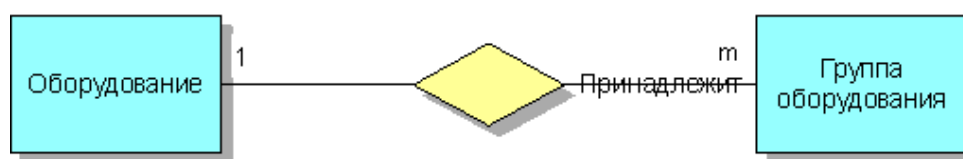


Рисунок 27-Группировка

В модели eERM используется расширенное понятие мощности отношений. При ссылке на число, характеризующее мощность отношений, ранее упоминались только верхние пределы допустимого числа экземпляров отношений.

Мощности на рисунке 28 (а) (см. [Рисунок 28](#)) указывают, что проект может быть связан с максимальным числом ( $w$ ) служащих и один служащий может участвовать в максимальном числе ( $n$ ) проектов.

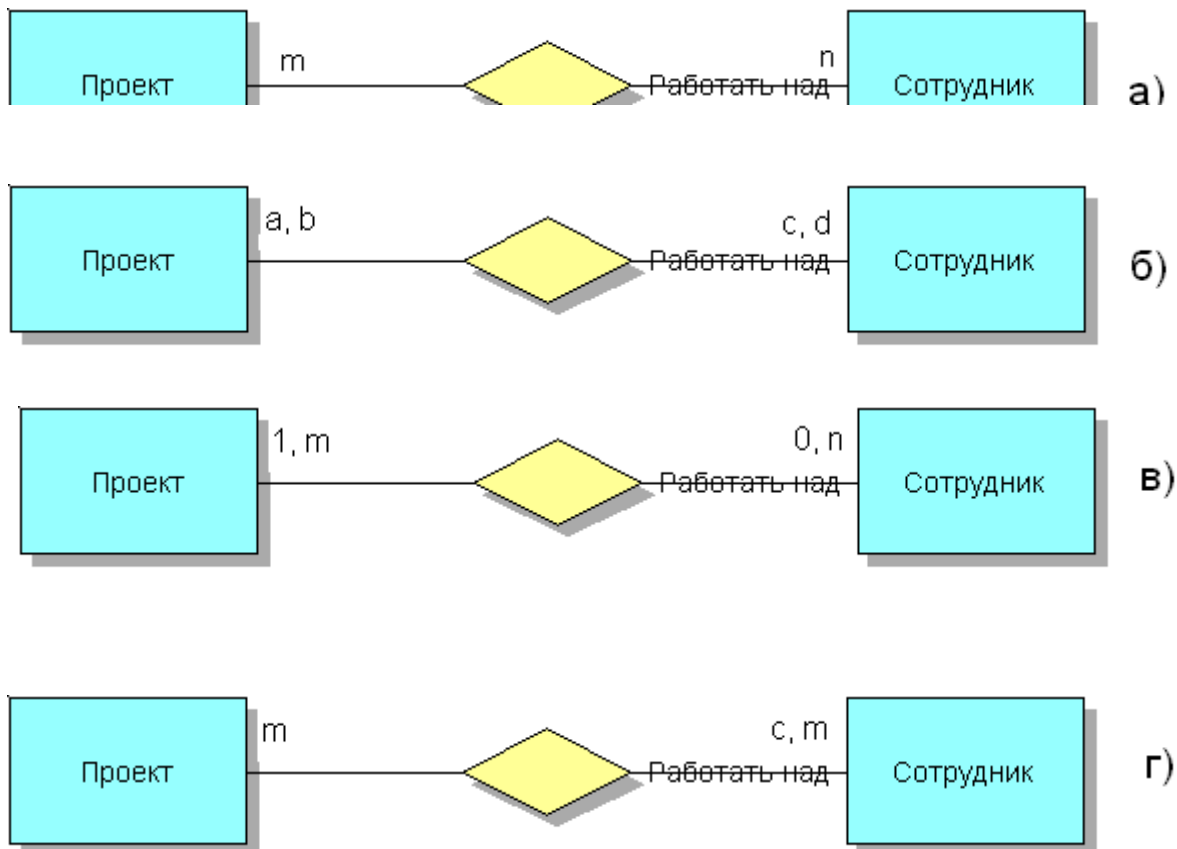


Рисунок 28- Верхнее/нижнее ограничение мощности отношений

Кроме верхнего ограничения, может также представлять интерес нижнее ограничение. Это имеет место в том случае, когда нужно специфицировать минимальное число экземпляров отношений. Для этих целей мощность отношений может быть выражена двумя буквами. Буквы ( $a, b$ ) на рисунке 28 (б) (см. [Рисунок 28](#)) указывают, что для каждого проекта можно определить, по крайней мере,  $a$  и не более чем  $b$  экземпляров отношений типа *работать над*. Это означает, что каждый проект может выполняться, по крайней мере,  $a$  и не более чем  $b$  сотрудниками. Буквы ( $c, d$ ) указывают, что один сотрудник может участвовать, по крайней мере, в  $c$  и не более чем в  $d$  проектах.

Таким образом, каждое отношение выражается двумя степенями сложности ( $min$  — минимальной и  $max$  — максимальной). Нижним границам часто присваиваются значения 0 или 1. Диапазон значений верхней границы определяется как  $1 < max < *$  (где  $*$  — универсальный знак).

Нижняя граница  $min = 0$  указывает, что сущность может участвовать в одном отношении, но это необязательно. Нижняя граница  $min = 1$  указывает, что сущность должна участвовать, по крайней мере, в одном отношении.

Нижние границы на рисунке 28(в) (см. [Рисунок 28](#)) указывают, что сотрудник может участвовать в проекте, но это необязательно ( $min = 0$ ).

Этим здесь подчеркивается, что могут быть сотрудники, которые не участвуют ни в одном проекте, в то время как проект должен выполняться, по крайней мере, одним сотрудником ( $\min = 1$ ). И, наоборот, по крайней мере, по одному сотруднику должно быть прикреплено к каждому проекту.

Если разрешены минимальное значение 0 или 1 и максимальные значения 1 или \*, то для пары  $(\min, \max)$  возможны следующие четыре сочетания:  $(1,1)$ ,  $(1, m)$ ,  $(0,1)$  и  $(0, m)$ . Приведенные ниже сокращения являются обычными для указанных сочетаний:

- 1 (соответствует  $(1,1)$ ),
- $c$  (соответствует  $(0,1)$ ),
- $m$  (соответствует  $(1, w)$ ),
- $cm$  (соответствует  $(0,m)$ ), где  $c = \text{choice}$  (выбор) и  $m == \text{multiple}$  (многократный).

Благодаря расширению понятия мощности посредством определения нижней и верхней границ может быть рассмотрена зависимость между объектами данных. По определению, типы отношения и типы переопределенных отношений существуют в силу существования типов сущностей, связанных с ними. Следовательно, они не существуют в изоляции. Это означает, что типы отношений зависят от других типов сущностей как с точки зрения существования, так и в плане идентификации.

Кроме того, имеют место типы сущностей, которые в действительности обладают единственным ключевым атрибутом, но при этом зависят от наличия других сущностей. Эти типы зависимостей могут появляться, например, при групповых операциях. Как показано на рисунке 30 (см. [Рисунок 29](#)), сущность *отдел* имеет смысл только в том случае, если он содержит, по крайней мере, одну сущность *рабочее место*. В свою очередь *рабочее место* только тогда имеет смысл, когда оно входит в *отдел*. Эти зависимости существования выражаются степенью сложности, или мощностью. В нотации  $(\min, \max)$  они определяются как  $(1,1)$  и  $(1,*)$ .

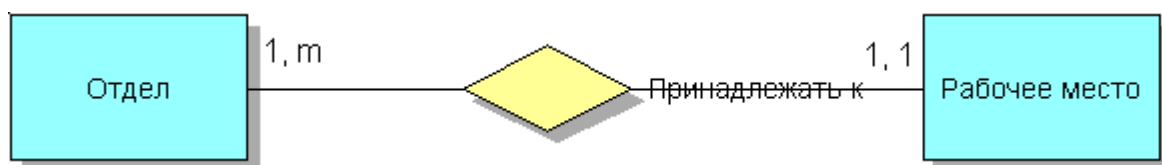


Рисунок 29- Зависимость существования

Определение зависимости существования в модели данных обеспечивает целостность основных данных, что является важным условием при реализации. Его выполнение гарантирует, что целостность содержимого базы данных обеспечивается даже после выполнения некоторых транзакций. В приведенном выше примере (см. [Рисунок 29](#)) удалить отдел можно только в том случае, если будут удалены все рабочие места, входящие в данный отдел.

С помощью диаграммы атрибутов eERM-модели можно описать атрибуты для каждого типа сущности и отношения на отдельной диаграмме. В нее можно включить тип объекта из диаграммы eERM (тип сущности или тип отношения) в виде копии экземпляра. Таким образом, может быть смоделировано распределение атрибутов по объектам eERM. В этом контексте можно различать, является ли атрибут, связанный с объектом eERM, ключевым атрибутом, внешним ключом или описательным атрибутом. На рисунке 30 (см. [Рисунок 30](#)) приведен соответствующий пример.

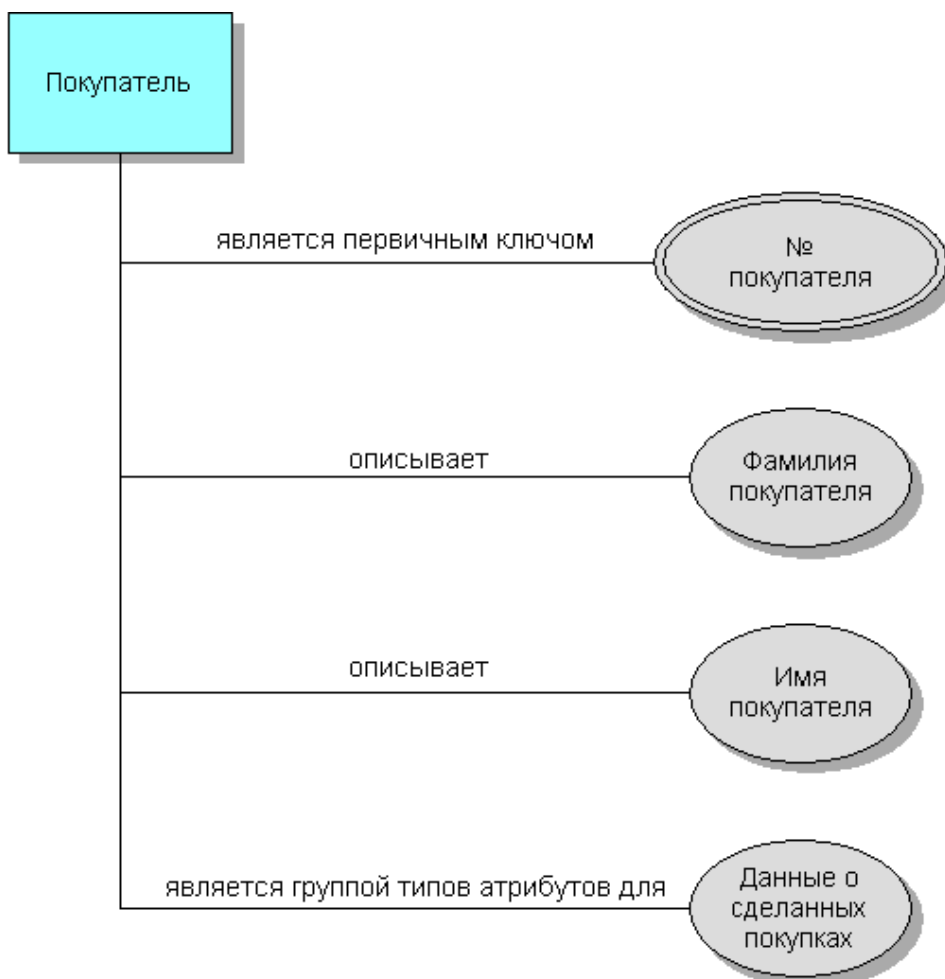


Рисунок 30-Диаграмма атрибутов eERM-модели для сущности «Покупатель»

Кроме представления и распределения отдельных атрибутов для объектов eERM-модели, в этом типе диаграммы можно также отобразить группу типов атрибутов (например, группа «Данные о сделанных покупках» на рисунке 30) и их распределение. Группа типов атрибутов представляет группу атрибутов одного типа сущности из диаграммы eERM, которые семантически близко связаны. Это позволяет создать группу атрибутов, содержащую все атрибуты объектов eERM-модели, которые вместе образуют, например, вторичный ключ.

### *Состав и особенности информационной модели задачи.*

Информационная модель задачи является чрезвычайно важной, поскольку, как уже было сказано, ошибки, допущенные в определении объемов информации, необходимой для решения задачи, и ее структуризации, очень дорого обходятся при дальнейшем выполнении проекта.

От правильного определения объема и состава необходимой информации зависит, например, количество и состав подразделений и информационных ресурсов предприятия, привлекаемых к использованию предлагаемого решения или обмену данными с ним.

Информационная модель задачи должна носить объектно-ориентированный характер. Обычно такая модель строится дважды: по результатам обследования предприятия и по результатам реинжиниринга.

В первом случае модель строится на основании анализа структур существующих баз данных, структур документов, используемых при существующем порядке решения задачи, и бесед с компетентными специалистами, знающими все формальные и неформальные (чаще всего это особенно важно!) аспекты решения задачи в данных подразделениях данного предприятия. Следует отметить, что многие особенности организации создания и хранения данных, а также обмена данными на предприятии связаны с особенностями системы подчинения тех или иных подразделений. В этом отношении при построении информационной модели следует иметь в виду уже построенную модель оргструктуры.

Во втором случае ранее построенная модель может подвергнуться изменению в части пересмотра состава множества объектов и их внутренней структуры, а также типов отношений между объектами в результате работы по достижению необходимой степени объектной «чистоты» модели. Совершенствование модели рекомендуется осуществлять с учетом облегчения в дальнейшем переноса полученной структуры данных в базу данных конкретного программного продукта (в нашем случае – PDM-системы).

Пример объектной информационной модели задачи приведен на рисунке 31 (см. [Рисунок 31](#)). Как видно из рисунка, данная модель очень напоминает классическую ER-диаграмму: на ней представлены сущности (классы) и типы отношений между ними. Также на диаграмме указываются мощности отношений между сущностями: «1 к n» и т.д. Под отношением понимается логическая связь между сущностями в контексте задачи.

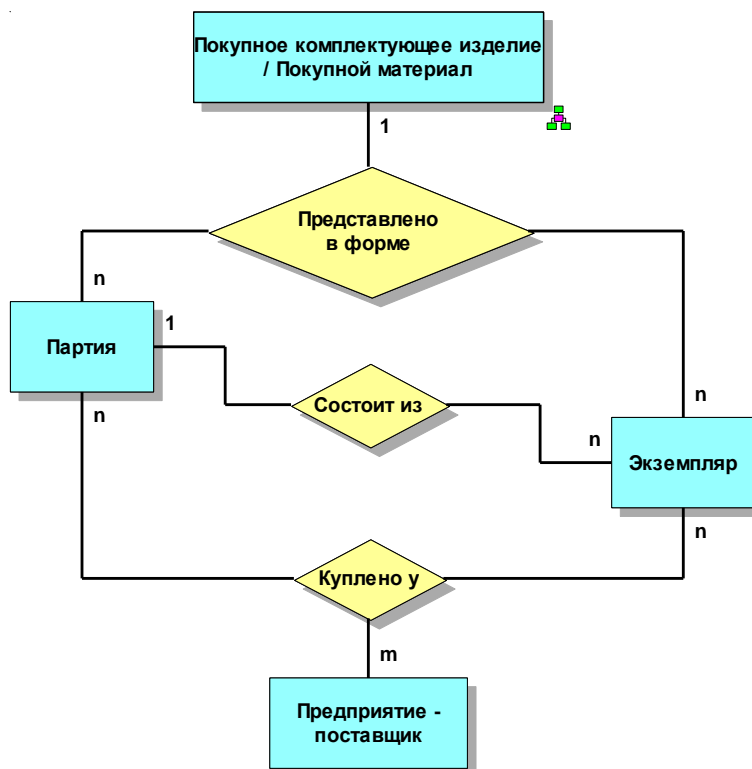


Рисунок 31-Пример объектной информационной модели

*Дополнение информационной модели моделями назначения атрибутов.*

В большинстве случаев сложно показать на одной модели одновременно и состав множества сущностей и их структуры (модель становится слишком громоздкой и сложной для восприятия). Поэтому информационную модель приходится дополнять *моделями назначения атрибутов* для каждой сущности. Для сущности «Покупное комплектующее изделие» (см. [Рисунок 31](#)) такая дополнительная модель может выглядеть, например, так, как показано на рисунке 32 (см. [Рисунок 32](#)).

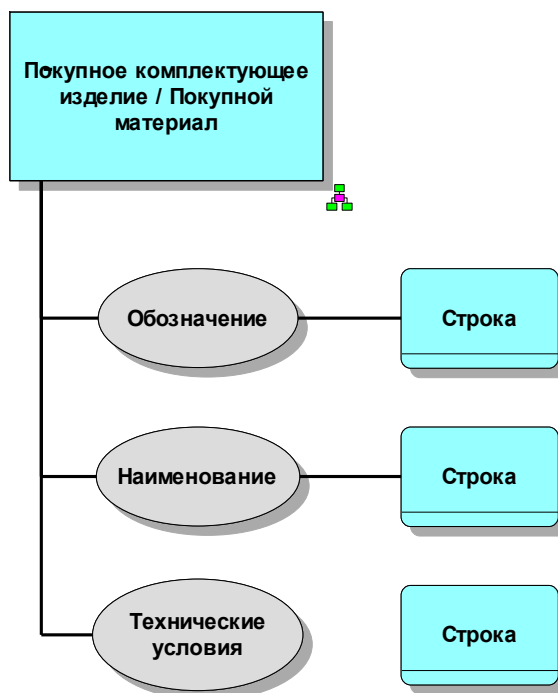


Рисунок 32 - Пример модели назначения атрибутов

Как видно из рисунка, в данной модели отражается состав атрибутов сущности и типы значений атрибутов.

В результате построения для каждой сущности модели назначения атрибутов информационная модель задачи неизбежно приобретает двухуровневую иерархическую структуру.

#### *Построение информационной модели задачи в ARIS.*

Для построения информационной модели обычно используются объекты-примитивы, представленные в таблице 7 (см. [Таблица 7](#)) и связи из таблицы 8 (см. [Таблица 8](#)).

На рисунке 33 (см. [Рисунок 33](#)) представлены варианты использования указанных примитивов при построении информационной модели. В зависимости от расположения отношения между сущностями на диаграмме, его следует читать либо слева – направо, либо - сверху – вниз.

Следует внимательно относиться к отнесению того или иного элемента данных к категории сущностей или категории атрибутов. Бывает так, что это не имеет значения с точки зрения описания структуры данных, но может создать потом проблемы при создании программного продукта.

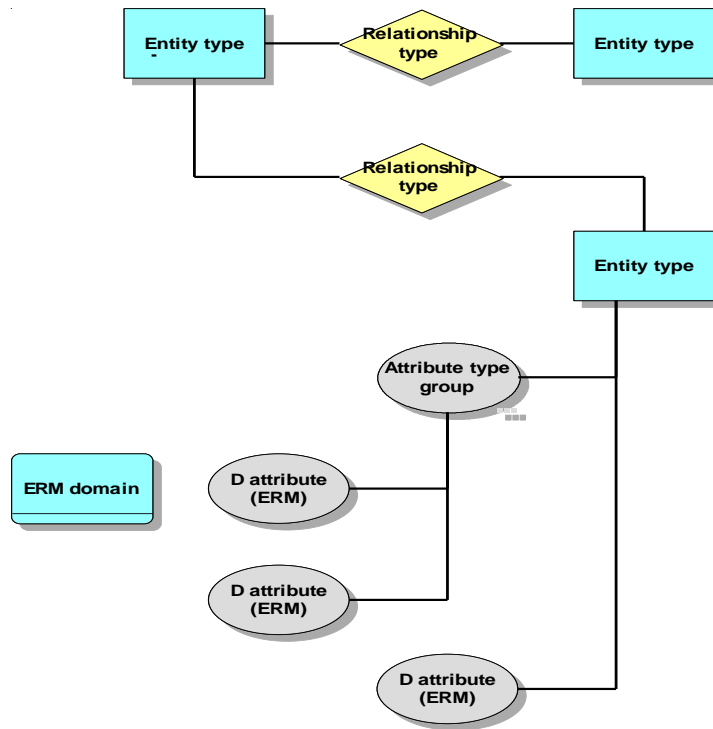


Рисунок 33 - Примеры комбинаций элементов информационной модели

На рисунке 34 (см. [Рисунок 34](#)) показано окно построения информационной модели, а на рисунке 35 (см. [Рисунок 35](#)) – окно построения модели назначения атрибутов.

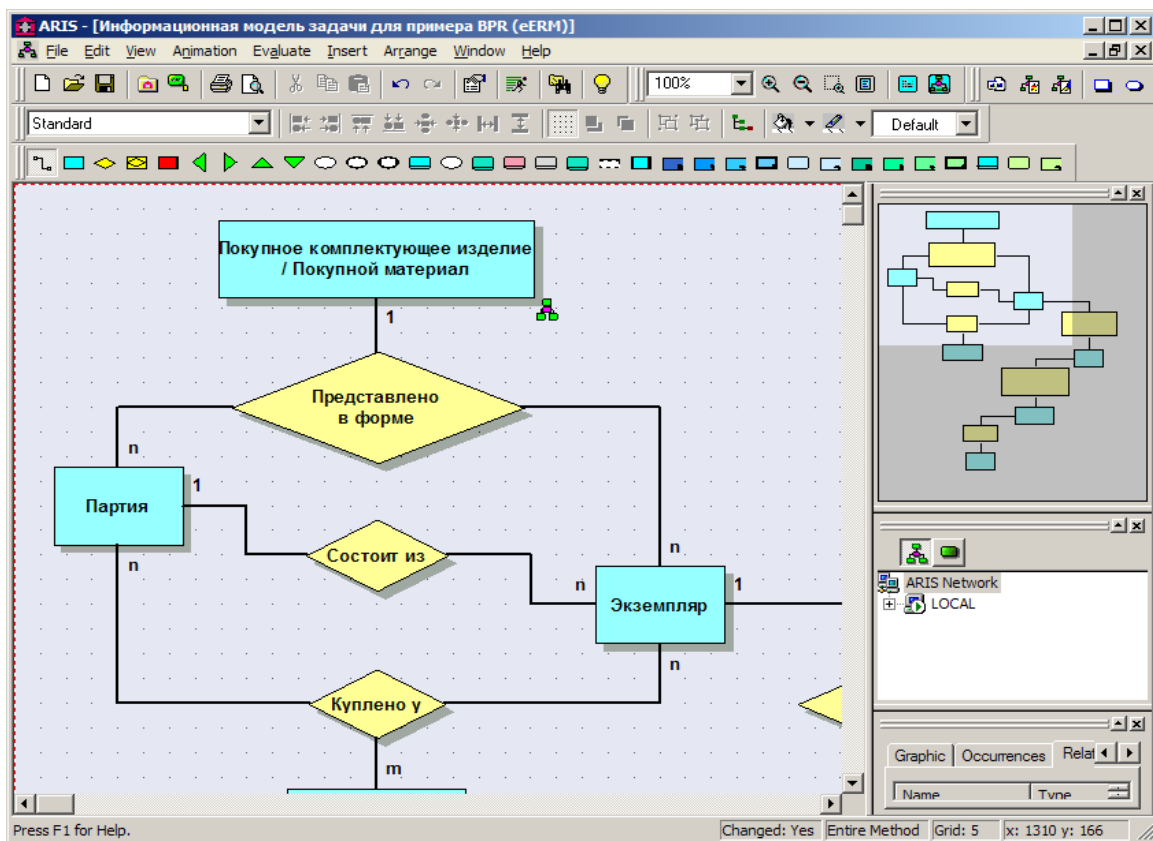


Рисунок 34 - Вид окна построения информационной модели



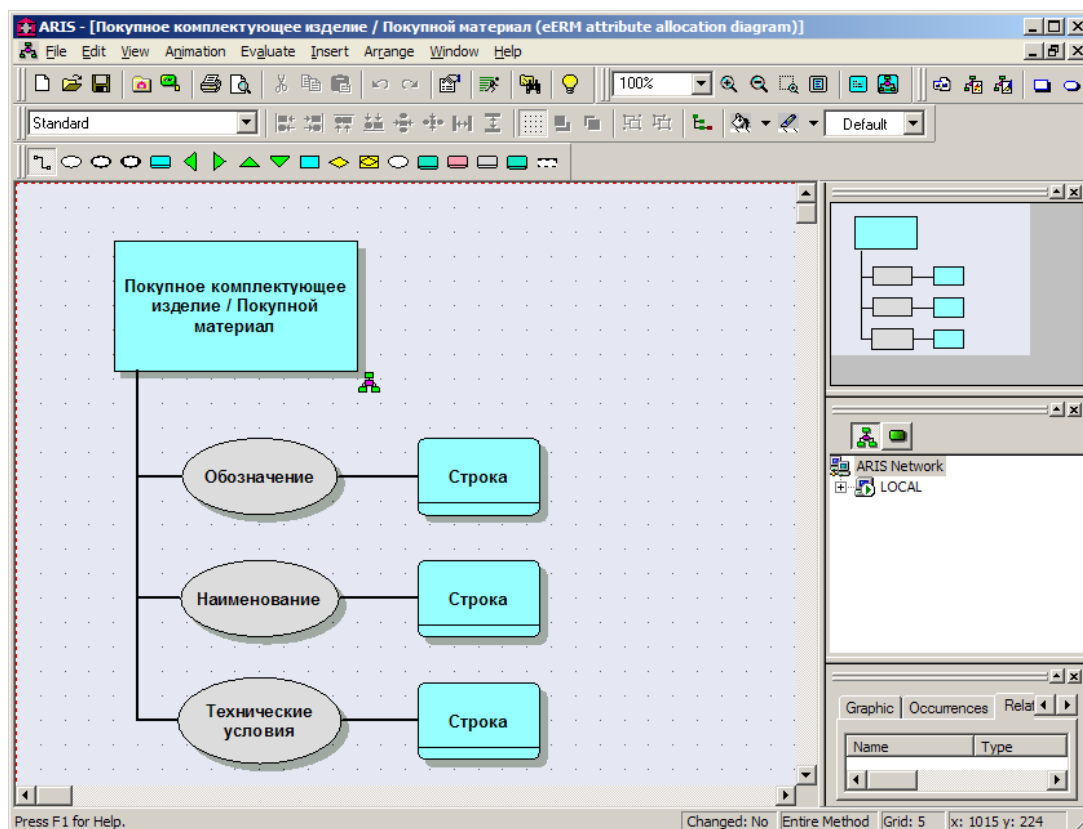


Рисунок 35 - Вид окна построения модели назначения атрибутов

### Порядок выполнения работы

1. Ознакомиться с заданием и теоретической частью данных методических указаний.
2. Проанализировать исходные данные;
3. Построить информационную модель, описывающую часть предметной области машиностроительного предприятия. Для этого создать:
  - а. одну диаграмму «сущность - отношение» (eERM);
  - б. набор диаграмм атрибутов, по одной для каждой сущности (декомпозированные из основной диаграммы «сущность - отношение» (eERM)).
4. Продемонстрировать созданную модель преподавателю;
5. Оформить отчет.

### Исходные данные задания

Часто информационное моделирование подразумевает создание набора моделей, описывающих информацию необходимую для решения конкретной задачи.

В этот набор, как правило, входит основная модель (в нотации ERM) и какие-либо дополнительные модели. В данной лабораторной работе предлагается построить информационную модель в нотации ERM и дополняющие ее модели атрибутов.

### **Вариант №1.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Чертеж детали
- Сборочный чертеж
- Спецификация
- Техпроцесс

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Позиция

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Чертеж детали» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера

С понятием «Сборочный чертеж» связаны свойства:

- Обозначение
- Наименование
- Литера

С понятием «Спецификация» связаны свойства:

- Обозначение

С понятием «Техпроцесс» связаны свойства:

- Обозначение
- Наименование
- Статус техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

### **Вариант №2.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица

- Чертеж детали
- Сборочный чертеж
- Спецификация
- Техпроцесс
- Маршрутная карта

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Чертеж детали» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Количество листов
- Формат

С понятием «Сборочный чертеж» связаны свойства:

- Обозначение
- Наименование
- Литера
- Количество листов

С понятием «Спецификация» связаны свойства:

- Обозначение

С понятием «Маршрутная карта» связаны свойства:

- Обозначение
- Наименование

С понятием «Техпроцесс» связаны свойства:

- Обозначение
- Наименование

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

Техпроцесс имеет маршрутную карту.

### **Вариант №3.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Чертеж детали

- Техпроцесс
- Технологическое разрешение
- Маршрутная карта
- Операционная карта

С понятием «*Деталь*» связаны свойства:

- Обозначение
- Наименование
- Зона
- Позиция
- Количество

С понятием «*Чертеж детали*» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера
- Количество листов
- Формат

С понятием «*Технологическое разрешение*» связаны свойства:

- Обозначение
- Наименование
- Расцеховка
- Подразделение
- Подпись технолога цеха
- Дата подписи технологом цеха

С понятием «*Маршрутная карта*» связаны свойства:

- Обозначение
- Наименование

С понятием «*Операционная карта*» связаны свойства:

- Обозначение
- Номер операции

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование
- Стадия техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Деталь имеет чертеж.

Техпроцесс имеет технологическое разрешение, маршрутную карту и операционную карту.

#### **Вариант №4.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица

- Чертеж детали
- Сборочный чертеж
- Спецификация

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Чертеж детали» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера
- Формат

С понятием «Сборочный чертеж» связаны свойства:

- Обозначение
- Наименование
- Литера
- Формат

С понятием «Спецификация» связаны свойства:

- Обозначение

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

### **Вариант №5.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Спецификация
- Техпроцесс
- Технологическое разрешение
- Маршрутная карта
- Операционная карта

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Зона
- Позиция

- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «*Спецификация*» связаны свойства:

- Обозначение

С понятием «*Технологическое разрешение*» связаны свойства:

- Обозначение
- Наименование
- Расцеховка
- Номер участка
- Подразделение
- Подпись технолога цеха
- Дата подписи технологом цеха
- Подпись главного технолога
- Дата подписи главным технологом

С понятием «*Маршрутная карта*» связаны свойства:

- Обозначение
- Наименование

С понятием «*Операционная карта*» связаны свойства:

- Обозначение
- Номер операции

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование
- Статус техпроцесса
- Стадия техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Сборочная единица имеет спецификацию.

Деталь и сборочная единица указываются в спецификации.

Техпроцесс имеет технологическое разрешение, маршрутную карту и операционную карту.

### **Вариант №6.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Техпроцесс
- Маршрутная карта
- Операционная карта

С понятием «*Деталь*» связаны свойства:

- Обозначение

- Наименование
- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Маршрутная карта» связаны свойства:

- Обозначение
- Наименование

С понятием «Операционная карта» связаны свойства:

- Обозначение
- Номер операции

С понятием «Техпроцесс» связаны свойства:

- Обозначение
- Наименование
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Техпроцесс имеет маршрутную карту и операционную карту.

### **Вариант №7.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Чертеж детали
- Сборочный чертеж
- Спецификация
- Техпроцесс

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Зона
- Позиция
- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Чертеж детали» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера
- Количество листов
- Формат

С понятием «Сборочный чертеж» связаны свойства:

- Обозначение
- Наименование
- Литера
- Количество листов
- Формат

С понятием «*Спецификация*» связаны свойства:

- Обозначение

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование
- Статус техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

### **Вариант №8.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Чертеж детали
- Сборочный чертеж
- Спецификация
- Техпроцесс
- Маршрутная карта

С понятием «*Деталь*» связаны свойства:

- Обозначение
- Наименование
- Количество

С понятием «*Сборочная единица*» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «*Чертеж детали*» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Формат

С понятием «*Сборочный чертеж*» связаны свойства:

- Обозначение
- Наименование



С понятием «*Спецификация*» связаны свойства:

- Обозначение

С понятием «*Маршрутная карта*» связаны свойства:

- Обозначение
- Наименование

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

Техпроцесс имеет маршрутную карту.

### **Вариант №9.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Чертеж детали
- Техпроцесс
- Технологическое разрешение
- Маршрутная карта
- Операционная карта

С понятием «*Деталь*» связаны свойства:

- Обозначение
- Наименование
- Зона
- Позиция
- Количество

С понятием «*Чертеж детали*» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера
- Количество листов
- Формат

С понятием «*Технологическое разрешение*» связаны свойства:

- Обозначение
- Наименование
- Подразделение
- Подпись главного технолога
- Дата подписи главным технологом

С понятием «*Маршрутная карта*» связаны свойства:

- Обозначение
- Наименование

С понятием «*Операционная карта*» связаны свойства:

- Обозначение
- Номер операции

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование
- Статус техпроцесса
- Стадия техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Деталь имеет чертеж.

Техпроцесс имеет технологическое разрешение, маршрутную карту и операционную карту.

### **Вариант №10.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Чертеж детали
- Сборочный чертеж
- Спецификация

С понятием «*Деталь*» связаны свойства:

- Обозначение
- Наименование
- Позиция
- Количество

С понятием «*Сборочная единица*» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «*Чертеж детали*» связаны свойства:

- Обозначение
- Наименование
- Масса детали
- Литера
- Количество листов

С понятием «*Сборочный чертеж*» связаны свойства:

- Обозначение
- Наименование
- Литера
- Формат

С понятием «*Спецификация*» связаны свойства:

- Обозначение

Сборочная единица может содержать детали и сборочные единицы.

Деталь имеет чертеж.

Сборочная единица имеет сборочный чертеж и спецификацию.

Деталь, сборочная единица и сборочный чертеж указываются в спецификации.

На сборочном чертеже изображаются деталь и сборочная единица.

### **Вариант №11.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Спецификация
- Техпроцесс
- Технологическое разрешение
- Маршрутная карта
- Операционная карта

С понятием «Деталь» связаны свойства:

- Обозначение
- Наименование
- Зона
- Позиция
- Количество

С понятием «Сборочная единица» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «Спецификация» связаны свойства:

- Обозначение

С понятием «Технологическое разрешение» связаны свойства:

- Обозначение
- Наименование
- Подпись технолога цеха
- Дата подписи технологом цеха
- Подпись главного технолога
- Дата подписи главным технологом

С понятием «Маршрутная карта» связаны свойства:

- Обозначение
- Наименование

С понятием «Операционная карта» связаны свойства:

- Обозначение
- Номер операции

С понятием «Техпроцесс» связаны свойства:

- Обозначение
- Наименование

- Статус техпроцесса
- Стадия техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Сборочная единица имеет спецификацию.

Деталь и сборочная единица указываются в спецификации.

Техпроцесс имеет технологическое разрешение, маршрутную карту и операционную карту.

### **Вариант №12.**

В процессе обследования были выделены следующие понятия предметной области:

- Деталь
- Сборочная единица
- Техпроцесс
- Маршрутная карта
- Операционная карта

С понятием «*Деталь*» связаны свойства:

- Обозначение
- Наименование
- Позиция
- Количество

С понятием «*Сборочная единица*» связаны следующие свойства:

- Обозначение сборочной единицы
- Наименование

С понятием «*Маршрутная карта*» связаны свойства:

- Обозначение
- Наименование

С понятием «*Операционная карта*» связаны свойства:

- Обозначение
- Номер операции

С понятием «*Техпроцесс*» связаны свойства:

- Обозначение
- Наименование
- Статус техпроцесса
- Литера

Деталь имеет техпроцесс изготовления.

Сборочная единица может содержать детали и сборочные единицы.

Техпроцесс имеет маршрутную карту и операционную карту.

### **Требования к отчёту по работе.**

Отчёт о работе должен содержать следующее:

1. Название работы;

2. Вариант;
3. Текст задания;
4. Распечатки разработанных моделей;
5. Выводы по проделанной работе.

### **Контрольные вопросы.**

1. Что описывает информационная модель?
  2. Какую информацию содержит диаграмма «сущность - отношение»?
  3. Какую информацию позволяет описать диаграмма атрибутов?
  4. Что понимается под понятием «сущность»?
  5. Что понимается под понятием «отношение»?
- Что такое мощность отношения?

## ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

1. **ARIS** (от англ. Architecture of Integrated Information Systems) — методология и программный продукт компании IDS Scheer для моделирования бизнес-процессов компании.
2. **CALS технология** – непрерывная информационная поддержка поставок и жизненного цикла — современный подход к проектированию и производству высокотехнологичной и наукоёмкой продукции, заключающийся в использовании компьютерной техники и современных информационных технологий на всех стадиях жизненного цикла изделия, обеспечивающая единообразные способы управления процессами и взаимодействия всех участников этого цикла: заказчиков продукции, поставщиков/производителей продукции, эксплуатационного и ремонтного персонала, реализованная в соответствии с требованиями системы международных стандартов, регламентирующих правила указанного взаимодействия преимущественно посредством электронного обмена данными.
3. **SADT** (от англ. Structured Analysis and Design Technique) — методология структурного анализа и проектирования, интегрирующая процесс моделирования, управление конфигурацией проекта, использование дополнительных языковых средств и руководство проектом со своим графическим языком.
4. **UML** (от англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования в основном программных систем. UML не является языком программирования, но в средствах выполнения UML-моделей как интерпретируемого кода возможна кодогенерация.
5. **Атрибут** - необходимое, существенное, неотъемлемое свойство объекта.
6. **Базовая функция** — это функция, которая уже не может быть разделена на составные элементы с целью анализа бизнес-процесса.
7. **Бизнес-процесс** — это последовательность взаимосвязанных мероприятий или задач, направленных на создание определенного продукта или услуги для потребителей.
8. **Диаграмма Чена** – по сути, диаграмма “сущность-связь” предназначены для разработки моделей данных и обеспечивают стандартный способ определения данных и отношений между ними. Фактически с помощью ERM осуществляется детализация хранилищ данных проектируемой системы, а также документируются сущности

системы и способы их взаимодействия, включая идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их отношений с другими объектами (связей).

9. **Жизненный цикл изделия** – совокупность процессов, выполняемых от момента выявления потребностей общества в определенной продукции до момента удовлетворения этих потребностей и утилизации продукта

10. **Модель** - совокупность объектов, объединенных друг с другом различными связями, и ряда вспомогательных элементов.

11. **Объект** — самостоятельная часть методологии, отражающая элемент описываемой предметной области.

12. **Предметная область** - часть реального мира, подлежащая изучению с целью организации управления и, в конечном счете, автоматизации. Предметная область представляется множеством фрагментов, например, предприятие - цехами, дирекцией, бухгалтерией и т.д. Каждый фрагмент предметной области характеризуется множеством объектов и процессов, использующих объекты, а также множеством пользователей, характеризуемых различными взглядами на предметную область.

13. **Реинжиниринг** - фундаментальное переосмысление и радикальное перепроектирование бизнес-процессов компаний для достижения коренных улучшений в наиболее важных показателях их деятельности – стоимость, качество и темпы.

14. **Репозиторий** - место, где хранятся и поддерживаются какие-либо данные (поля в специальной базе данных). Чаще всего данные в репозитории хранятся в виде файлов, доступных для дальнейшего распространения по сети.

15. **Событие** — это состояние, которое является существенным для целей управления бизнесом и которое оказывает влияние или контролирует дальнейшее развитие одного или более бизнес-процессов.

16. **Структурный анализ** - исследование статических характеристик системы путем выделения в ней подсистем и элементов различного уровня и определения отношений и связей между ними.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Норенков И.П., Кузьмик П.К. Информационная поддержка наукоемких изделий (CALS-технологии). – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002.
2. Консалтинг при автоматизации предприятий: подходы, методы, средства <http://vernikov.ru/component/k2/item/339.html#20-2>
3. Колчин А.Ф., Овсянников М.В., Стрекалов А.Ф., Сумароков С.В. Управление жизненным циклом продукции. – М.: Анахарсис, 2002.
4. Репин В.В., Елиферов В.Г. Процессный подход к управлению. Моделирование бизнес-процессов. – М.: РИА «Стандарты и качество», 2006. – 408 с.
5. Руководство пользователя ARIS
6. Шеер А.-В. Бизнес-процессы. Основные понятия. Теория. Методы. – М.: Весть-МетаТехнология, 1999. – 152 с.
7. Шеер А.-В. Моделирование бизнес-процессов. – М.: Весть-МетаТехнология, 2000. – 205 с.
8. Шматалюк А. и др. Моделирование бизнеса. Методология ARIS: Практическое руководство. – М.: Серебряные нити, 2001.

Учебное издание

*Кременецкая Марина Евгеньевна,  
Иващенко Антон Владимирович*

### МОДЕЛИРОВАНИЕ ПРОЦЕССОВ ЖИЗНЕННОГО ЦИКЛА ИЗДЕЛИЙ АВИАЦИОННОЙ ТЕХНИКИ ПО МЕТОДОЛОГИИ ARIS

*Лабораторный практикум*

*В авторской редакции*

Подписано в печать 16.11.2010. Формат 60x84 1/16.

Бумага офсетная. Печать офсетная. Печ. л. 5,5.

Тираж 50 экз. Заказ .

Самарский государственный аэрокосмический университет.  
443086, Самара, Московское шоссе, 34.

Изд-во Самарского государственного  
аэрокосмического университета.  
443086, Самара, Московское шоссе, 34.