

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный лесотехнический университет
имени Г.Ф. Морозова»

С.А. Евдокимова

CASE-ТЕХНОЛОГИИ

Практикум

ВОРОНЕЖ 2016

УДК 004

Печатается по решению учебно-методического совета
ФГБОУ ВО «ВГЛУ им. Г.Ф. Морозова» (протокол № от 2016 г.)

Рецензенты: кафедра электротехники и автоматики ФГБОУ ВПО «Воронежский ГАУ»
начальник лаборатории ФГУП «НИИЭТ», канд. техн. наук
А.И.Яньков

Евдокимова, С. А.

CASE-технологии [Текст] : практикум / С. А. Евдокимова ; М-во образования и науки РФ, ФГБОУ ВО «ВГЛУ». – Воронеж, 2016. – 128 с.

ISBN

Лабораторный практикум содержит тексты десяти лабораторных работ, рассматривающих создание диаграмм предметной области экономической организации по функциональной методологии IDEF0 в среде CA Erwin Process Modeler, по методологии ARIS в График-студии Лайт, имитационное моделирование экономических процессов.

Лабораторный практикум предназначен для студентов по направлению подготовки 38.03.01 – Экономика.

УДК 004

© Евдокимова С.А., 2016

ISBN © ФГБОУ ВО «Воронежский государственный
лесотехнический университет имени Г.Ф. Морозова», 2016

ОГЛАВЛЕНИЕ

Введение.....	4
Практическая работа № 1.	
Функциональная методология <i>IDEF0</i> . Создание контекстной диаграммы в среде <i>CA Erwin Process Modeler</i>	5
Практическая работа № 2.	
Создание диаграмм декомпозиции в среде <i>CA Erwin Process Modeler</i>	21
Практическая работа № 3.	
Создание диаграммы дерева узлов и <i>FEO</i> -диаграммы	36
Практическая работа № 4.	
Создание моделей в стандарте <i>DFD</i> (диаграммы потоков данных)	45
Практическая работа № 5.	
Создание моделей в стандарте <i>IDEF3</i> (метод описания процессов – <i>workflow diagramming</i>)	54
Практическая работа № 6.	
Методология <i>ARIS</i> . Создание организационной и функциональной моделей компании в <i>График-студии Лайт</i>	64
Практическая работа № 7.	
Создание функциональных диаграмм описания бизнес-процесса « <i>Издание журнала</i> » в <i>График-студии Лайт</i>	87
Практическая работа № 8.	
Моделирование бизнес-процессов в нотации <i>BPMN</i>	99
Практическая работа № 9.	
Система имитационного моделирования <i>Arena</i> . Построение модели процесса обработки документов	107
Практическая работа № 10.	
Моделирование ветвлений в системе <i>Arena</i> . Построение модели производственного цеха.....	122
Библиографический список.....	127

ВВЕДЕНИЕ

Практикум рассматривает CASE-средства построения функциональных и организационных диаграмм и предназначен для студентов по направлениям 38.03.01 – Экономика, 38.03.02 - Менеджмент, изучающих дисциплину «CASE-технологии».

Практикум содержит тексты десяти лабораторных работ. Практические работы № 1-3 рассматривают функциональную методологию IDEF0 и принципы создания контекстных диаграмм, диаграмм декомпозиции, дерева узлов в среде CA Erwin Process Modeler.

Практическая работа № 4 посвящена вопросам построения диаграмм потоков данных, работа № 5 – моделей в стандарте IDEF3.

Практические работы № 6-8 знакомят с методологией проектирования интегрированных информационных систем ARIS на примере таких диаграмм, как VAD, eEPC, FAD и других.

Практические работы № 9 и 10 рассматривают имитационное моделирование экономических процессов.

Каждая практическая работа содержит ряд упражнений, при выполнении которых студенты приобретают опыт работы с программными средствами. Все выполненные упражнения должны быть сохранены на жестком диске компьютера в папке, именуемой фамилией студента. Сохраняемые студентами файлы являются формой отчета по выполненным лабораторным работам.

В конце каждой практической работы приводится список вопросов для самопроверки, способствующих закреплению полученных знаний. Устный опрос по приведенным вопросам проводится преподавателем при защите студентом отчета по практической работе.

ПРАКТИЧЕСКАЯ РАБОТА № 1

Тема работы: Функциональная методология *IDEF0*. Создание контекстной диаграммы в среде *CA Erwin Process Modeler*.

CASE-технологии представляют собой совокупность методов проектирования экономических информационных систем (ЭИС), набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать эту модель на всех стадиях разработки и сопровождения ЭИС и разрабатывать приложения в соответствии с информационными потребностями пользователей. Большинство существующих *CASE*-средств основано на методах структурного или объектно-ориентированного анализа и проектирования, использующих спецификации в виде диаграмм или текстов для описания внешних требований, связей между моделями системы, динамики поведения системы и архитектуры программных средств.

Инструментальное средство проектирования CA Erwin Process Modeler

Система *CA Erwin Process Modeler* является инструментальным средством для моделирования, анализа, документирования и оптимизации бизнес-процессов, которую можно использовать для графического представления схем выполнения работ, обмена информацией, документооборота.

Основные возможности *CA Erwin Process Modeler* следующие:

- поддержка различных технологий моделирования: *IDEF0* (функциональное моделирование), *DFD* (моделирование потоков данных) и *IDEF3* (моделирование потоков работ). Эти три основных метода позволяют комплексно описать предметную область;
- анализ показателей затрат и производительности с использованием методов расчета себестоимости по объему хозяйственной деятельности (функционально-стоимостной анализ, ABC);
- сбор дополнительной, существенной для бизнеса информации с помощью свойств, определенных пользователем. Введенная информация может быть отображена в отчетах, сгенерированных с помощью генератора отчетов, и экспортирована в другие программы, например в *Microsoft Word*, *Microsoft Excel*;

- контроль корректности модели достигается невозможностью создания в модели некорректных связей, автоматической миграцией граничных стрелок, отслеживанием дисбаланса граничных стрелок на дочерней и родительской диаграммах (туннели), проверкой наличия имен стрелок и имен функциональных блоков;

- интеграция с *CA Erwin Data Modeler* позволяет организовать многопользовательскую среду для работы целого коллектива разработчиков с моделями бизнес-процессов, сохранение моделей в единый репозиторий, обеспечивая аутентификацию и контроль доступа, разрешение конфликтов, возникающих при одновременной работе с одной моделью нескольких разработчиков, а также управление версиями моделей.

Функциональная методология IDEF0

Методология *IDEF0* предназначена для функционального моделирования, т.е. моделирования выполнения функций объекта путем создания описательной графической модели, показывающей что, как и кем делается в рамках функционирования предприятия. Функциональная модель представляет собой структурированное изображение функций производственной системы или среды, информации и объектов, связывающих эти функции.

Модель экономической системы в методологии *IDEF0* – это совокупность иерархически упорядоченных и взаимосвязанных диаграмм. Каждая диаграмма является единицей описания системы и располагается на отдельном листе. Модель может содержать четыре типа диаграмм:

1. *Контекстную диаграмму*, которая является вершиной древовидной структуры диаграмм и представляет собой общее описание системы и ее взаимодействие с внешней средой. Модель может иметь только одну контекстную диаграмму.

2. *Диаграммы декомпозиции*, которые получаются в результате разбиения контекстной диаграммы на крупные фрагменты. После декомпозиции контекстной диаграммы проводится декомпозиция каждой получившейся диаграммы и т.д. Синтаксис описания системы в целом и каждого ее фрагмента одинаков во всей модели.

3. *Диаграммы дерева узлов* показывают иерархическую зависимость работ, но не взаимосвязи между работами. Диаграмм деревьев узлов может быть в

модели несколько, поскольку дерево может быть построено на произвольную глубину и начиная с любой диаграммы (не обязательно с контекстной).

4. *Диаграммы только для экспозиции (FEO)* строятся для иллюстрации отдельных фрагментов модели, для иллюстрации альтернативной точки зрения, либо для хранения старых версий.

В *IDEF0* моделируемая система представляется как совокупность взаимосвязанных работ (функциональных блоков, активностей) (*Activity Box*). Работы обозначают поименованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. Работы изображаются в виде прямоугольников (рис. 1) и именуются отглагольным существительным, обозначающим действие (например, «Изготовление детали», «Прием заказа» и т.д.).

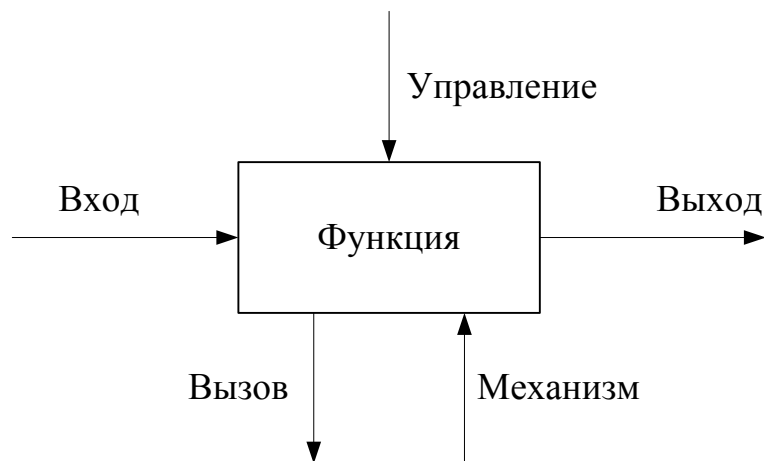


Рис. 1. Работа (Функциональный блок)

Все работы модели нумеруются. Номер состоит из префикса и числа. Может быть использован префикс любой длины, но обычно используют префикс *A*. Контекстная (корневая) работа дерева имеет номер *A0*. Работы декомпозиции *A0* имеют номера *A1*, *A2*, *A3* и т. д. Работы декомпозиции нижнего уровня имеют номер родительской работы и очередной порядковый номер, например, работы декомпозиции *A3* будут иметь номера *A31*, *A32*, *A33*, *A34* и т. д.

Взаимодействие работ с внешним миром и между собой описывается в виде стрелок. *Стрелки (Arrow)* отображают различные объекты, которые передаются между блоками, определяют правила обработки и механизмы обработки. Такими объектами могут быть элементы реального мира (детали, вагоны, сотрудники и т.д.) или потоки данных и информации (документы, данные, ин-

струкции и т.д.). Стрелки именуются существительными (например, «Заготовка», «Изделие», «Заказ»).

В *IDEFO* различают пять типов стрелок, каждый из которых подходит к определенной стороне работы, или выходит из нее (рис. 1):

1. *Вход (Input)* – материальные объекты или информация, которые используются или преобразуются работой для получения результата (выхода). Допускается, что работа может не иметь ни одной стрелки входа. Стрелка входа рисуется как входящая в левую грань работы.

2. *Управление (Control)* – правила, стратегии, процедуры, стандарты, которыми руководствуется работа. Каждая работа должна иметь хотя бы одну стрелку управления. Управление влияет на работу, но не преобразуется работой. В случае возникновения неопределенности в статусе стрелки (управление или вход) рекомендуется рисовать стрелку управления. Стрелка управления рисуется как входящая в верхнюю грань работы.

3. *Выход (Output)* – материальный объект или информация, которые производятся работой. Каждая работа должна иметь хотя бы одну стрелку выхода. Работа без результата не имеет смысла и не должна моделироваться. Стрелка выхода рисуется как исходящая из правой грани работы.

4. *Механизм (Mechanism)* – ресурсы, которые выполняют работу, например, персонал предприятия, станки, устройства и т.д. Стрелка механизма рисуется как входящая в нижнюю грань работы. По усмотрению аналитика стрелки механизма могут не изображаться в модели.

5. *Вызов (Call)* – специальная стрелка, указывающая на другую модель работы. Стрелка вызова используется при расщеплении модели и указывает, что некоторая работа представлена отдельной моделью. Расщепление модели необходимо для коллективной работы над моделью. Стрелка вызова рисуется как исходящая из нижней грани работы.

ICOM-коды (аббревиатура от *Input, Control, Output, Mechanism*) – это коды, предназначенные для идентификации граничных стрелок. *ICOM-коды* создаются автоматически и содержат префикс, соответствующий типу стрелок (*I, C, O* или *M*) и порядковый номер. Например, *I1, I2* и т.д.

Задание на лабораторную работу

Создать контекстную диаграмму модели деятельности некоторого предприятия, осуществляющего производство изделий на заказ. На основании полученных заказов формируется план выпуска готовой продукции на определенный период. В соответствии с этим планом определяются потребность в комплектующих изделиях и материалах, а также график загрузки производственного оборудования. После изготовления продукции и проведения платежей готовая продукция отправляется заказчику.

Начало работы в СА Erwin Process Modeler

Запуск программы *СА Erwin Process Modeler* осуществляется выбором в главном меню *Windows* пунктов *Все программы ⇒ СА ⇒ ERwin ⇒ ERwin Process Modeler ⇒ Process Modeler* или двойным щелчком мыши на соответствующем ярлыке на рабочем столе *Windows*.

Упражнение 1

1. Запустите программу *СА Erwin Process Modeler*. На экране появится диалоговое окно (рис. 2), в котором предлагается выбрать необходимое действие:

- *Create model* – создать модель;
- *Create model from template* – создать модель по шаблону;
- *Create model from ERwin MM template* – создать модель в среде групповой разработки;
- *Open model* – открыть модель;
- *Open model from ERwin MM template* – открыть модель из среды групповой разработки.

2. В диалоговом окне (рис. 2) установите переключатель в поле *Create model*, в поле *Name (Имя)* введите «Деятельность предприятия», выберите *Type – Business Process (IDEF0)* и нажмите *OK*. Появится окно *Properties for New Models (Свойства новой модели)* (рис. 3).

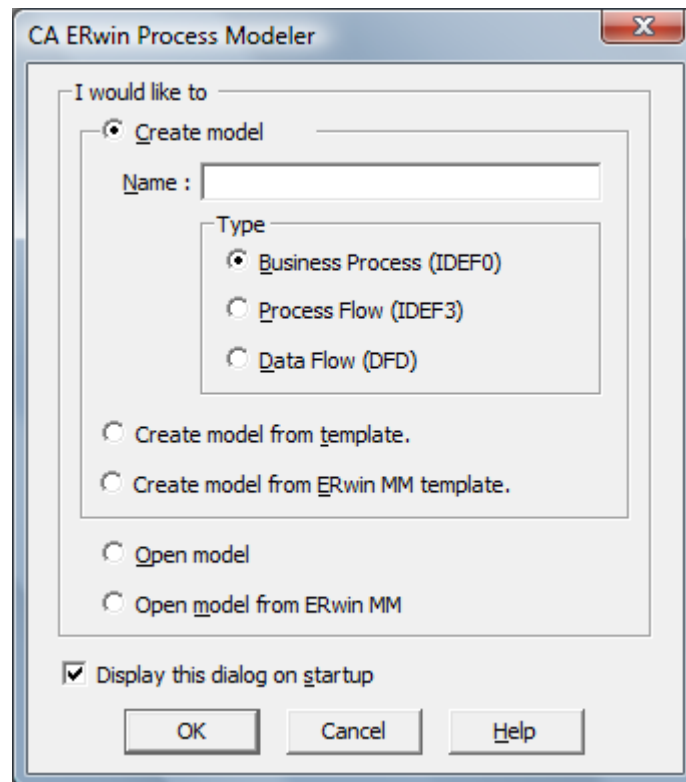


Рис. 2. Диалоговое окно создания/открытия модели

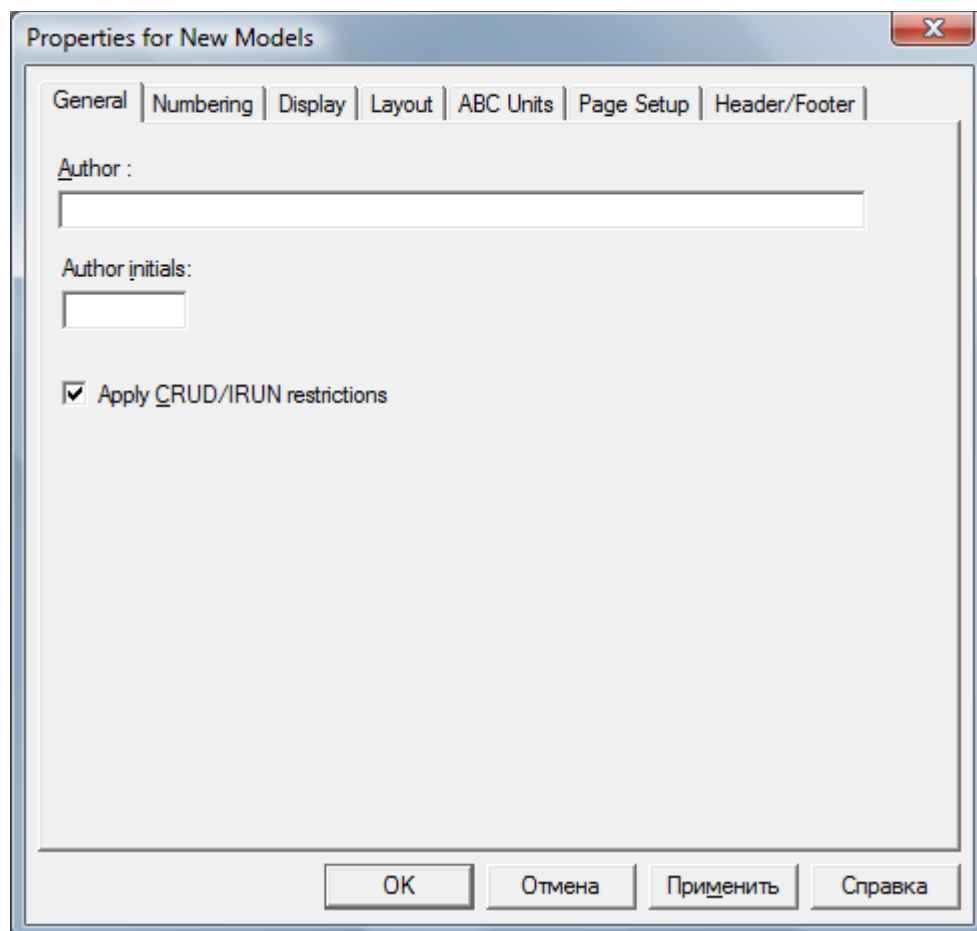


Рис. 3. Диалоговое окно задания свойств модели

3. В окне *Properties for New Models* (Свойства новой модели) (рис. 3) на вкладке *General* (Общее) введите свои фамилию и инициалы в поля *Author* и *Author initials* соответственно. Поставьте галочку в поле *Apply CRUD/IRUN restrictions* (Применить ограничения CRUD/IRUN) для связывания модели процессов и модели данных и нажмите *OK*. На экране появится основное окно программы, в рабочей области которого будет располагаться заготовка контекстной диаграммы (рис. 4).

В структуру окна программы *CA Erwin Data Modeler* входят стандартные элементы окна: заголовок, размерные кнопки, меню, панель инструментов и т.д. Основная область окна разделена на две части: слева расположен *Проводник модели*, содержащий иерархическое дерево, а справа представлена рабочая область диаграммы.

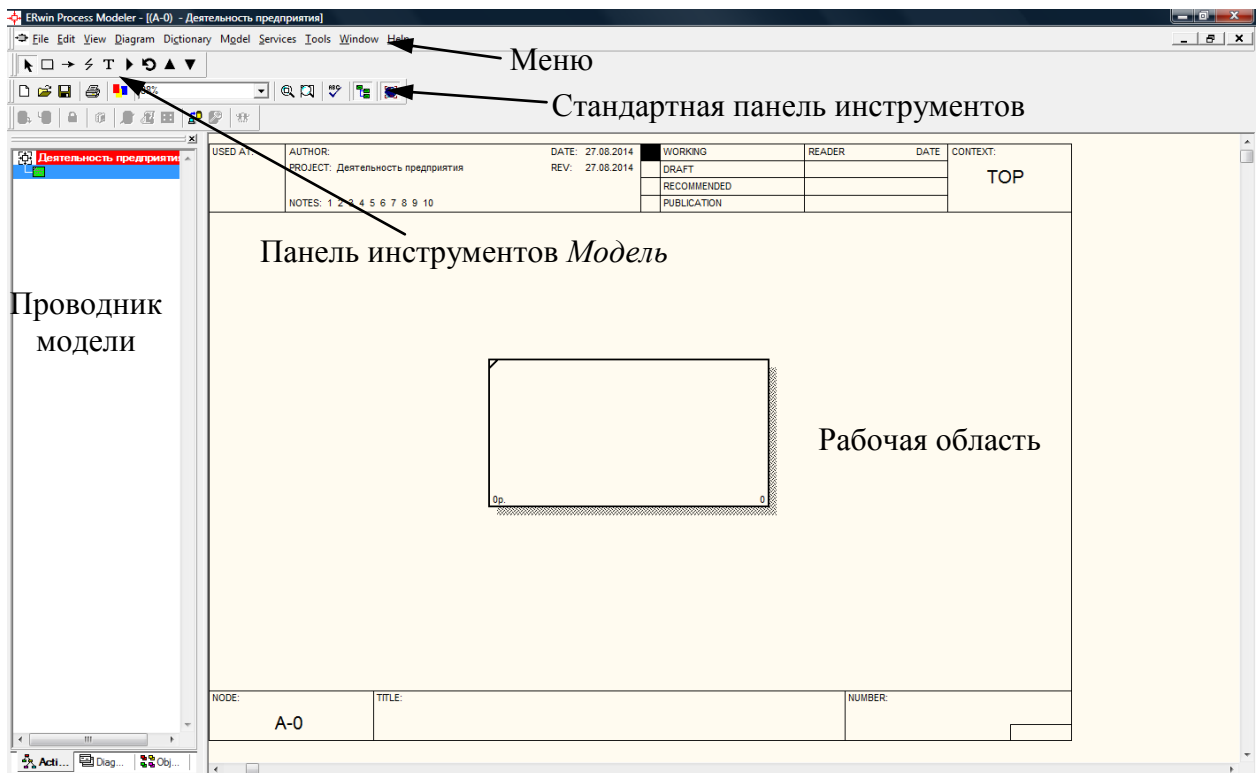


Рис. 4. Заготовка контекстной диаграммы

4. Для правильного отображения русских букв выполните команду меню *Model* \Rightarrow *Default Fonts* \Rightarrow *Parent Diagram Text* (Модель \Rightarrow Шрифты \Rightarrow Текст родительской диаграммы). В появившемся окне в поле *Script* (Рукописный шрифт) выберите *Кириллический*, установите галочку в поле *Change all occurrences of this font in the model* (Изменить все вхождения этого шрифта в модели) и нажмите *OK*. Если ничего не изменилось, то повторите эту операцию с

другим подменю, например, *Parent Diagram Title Text* (Текст заголовка родительской диаграммы).

5. Для задания свойств модели выполните команду меню *Model* \Rightarrow *Model Properties* (Модель \Rightarrow Свойства модели). На экране появится окно *Model Properties* (Свойства модели) (рис. 5), в котором установите:

– на вкладке *General* (Общее):

Model name (Имя модели) – Деятельность предприятия

Project (Проект) – Деятельность предприятия

Author (Автор) – Иванова (введите свою фамилию)

Author initials (Инициалы автора) – А.А. (введите свои инициалы)

Time Frame (Временные рамки) – AS-IS (Как есть)

– на вкладке *Purpose* (Цель):

Purpose (Цель) – Моделировать текущие бизнес-процессы предприятия

Viewpoint (Точка зрения) – Руководитель предприятия

The image shows a 'Model Properties' dialog box with the following elements:

- Tabs:** Layout, ABC Units, Page Setup, Header/Footer, Shapes, Draw Style, General (selected), Purpose, Definition, Source, Status, Numbering, Display.
- Model name:** Text field containing 'Деятельность предприятия'.
- Project:** Empty text field.
- Author:** Empty text field.
- Author initials:** Empty text field.
- Apply CRUD/IRUN restrictions:** Checked checkbox.
- Time Frame:** Radio buttons for 'AS-IS' (selected) and 'TO-BE'.
- Buttons:** OK, Отмена, Применить, Справка.

Рис. 5. Окно установки свойств модели

– на вкладке *Definition (Определение)*:

Definition (Определение) – Учебная модель, описывающая деятельность предприятия

Score (Область действия) – Общее управление деятельностью предприятия: прием заказов, закупка материалов и комплектующих, производство и продажа готовой продукции

– на вкладке *Status (Статус)* выберите *Working (Рабочий вариант)* и нажмите *OK*.

6. Выберите команду меню *File ⇒ Save (Файл ⇒ Сохранить)*, в появившемся окне *Сохранить как...* задайте имя файла (например, свою фамилию) и нажмите кнопку *Сохранить*.

Свойства созданных функциональных блоков (работ) можно изменить следующими способами:

– дважды щелкнуть на объекте и в появившемся диалоговом окне (рис. 6) задать требуемые значения;

– правой кнопкой мыши щелкнуть на объекте и в контекстном меню выбрать необходимое свойство;

– в проводнике модели на вкладке *Activities (Работы)* щелкнуть правой кнопкой мыши по строке с именем объекта и в появившемся контекстном меню выбрать необходимое свойство.

Упражнение 2

Задайте свойства работы контекстной диаграммы. Для этого дважды щелкните на объекте, и на экране появится диалоговое окно *Activity Properties (Свойства работы)* (рис. 6).

На вкладке *Name (Имя)* удалите надпись *Untitled Object ()*, введите *Деятельность предприятия*.

На вкладке *Definition (Определение)* внесите определение *Текущие бизнес-процессы предприятия* и нажмите *OK*.

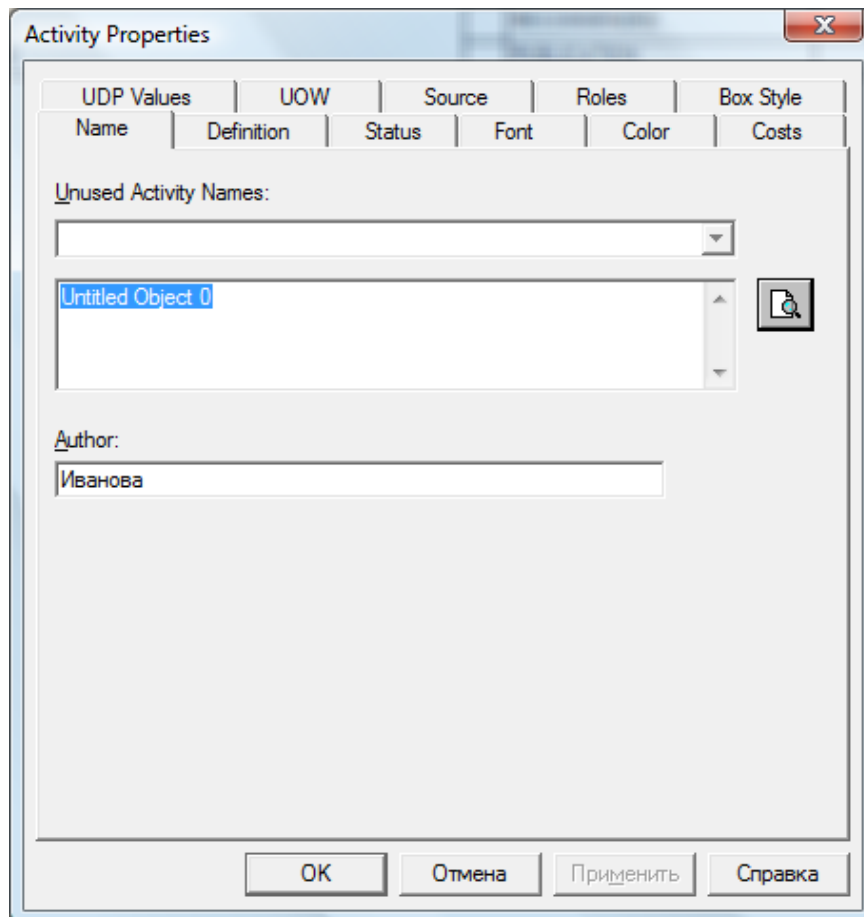


Рис. 6. Диалоговое окно *Activity Properties* (Свойства работы)

Стрелки на диаграмме рисуются с помощью кнопки *Precedence Arrow Tool* (Инструмент рисования стрелок) на панели инструментов *Model* (Модель) (рис. 7). Первым щелчком левой кнопкой мыши обозначается начало стрелки (граница области диаграммы или граница функционального блока), а вторым щелчком – конец.



Рис. 7. Панель *Model* (Модель)

Упражнение 3

1. Создайте стрелки на диаграмме в соответствии с рис. 8.
2. Для добавления имени стрелки дважды щелкните левой кнопкой мыши на стрелке (например, на стрелке входа) или в контекстном меню стрелки выберите пункт *Name* (Имя). В появившемся диалоговом окне *Arrow Properties* (Свойства стрелки) (рис. 9) в поле *Arrow Name* (Имя стрелки) введите *Материалы* и нажмите *Применить*.

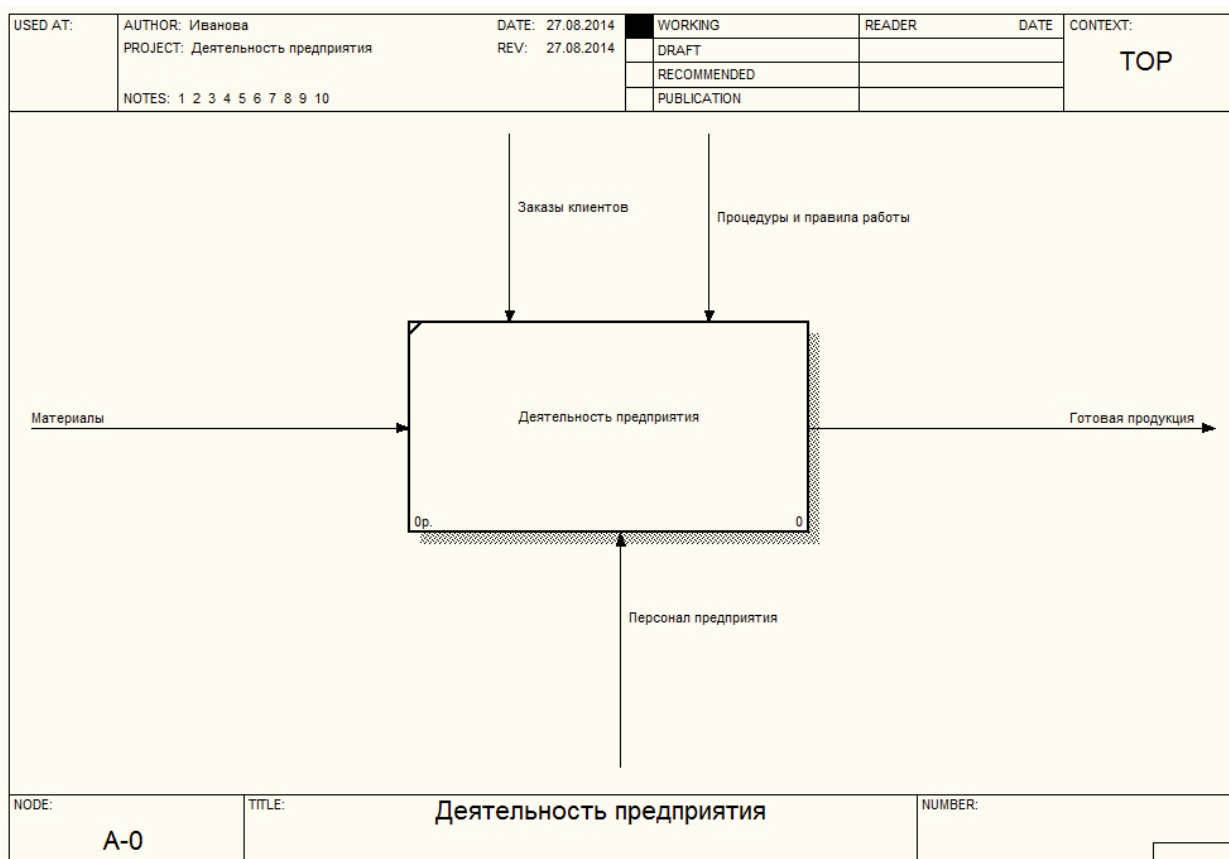


Рис. 8. Стрелки на контекстной диаграмме

Arrow Properties
X

Name
Style

Arrow Name:

☐ Replace all occurrences of this arrow name in model

Author:

OK
Отмена
Применить
Справка

Рис. 9. Окно *Arrow Properties* (Свойства стрелки)

3. Отобразится диалоговое окно полных свойств стрелки (рис. 10). На вкладке *Definition (Определение)* введите *Материалы, сырье, комплектующие изделия, которые необходимы для производства готовой продукции* и нажмите *OK*.

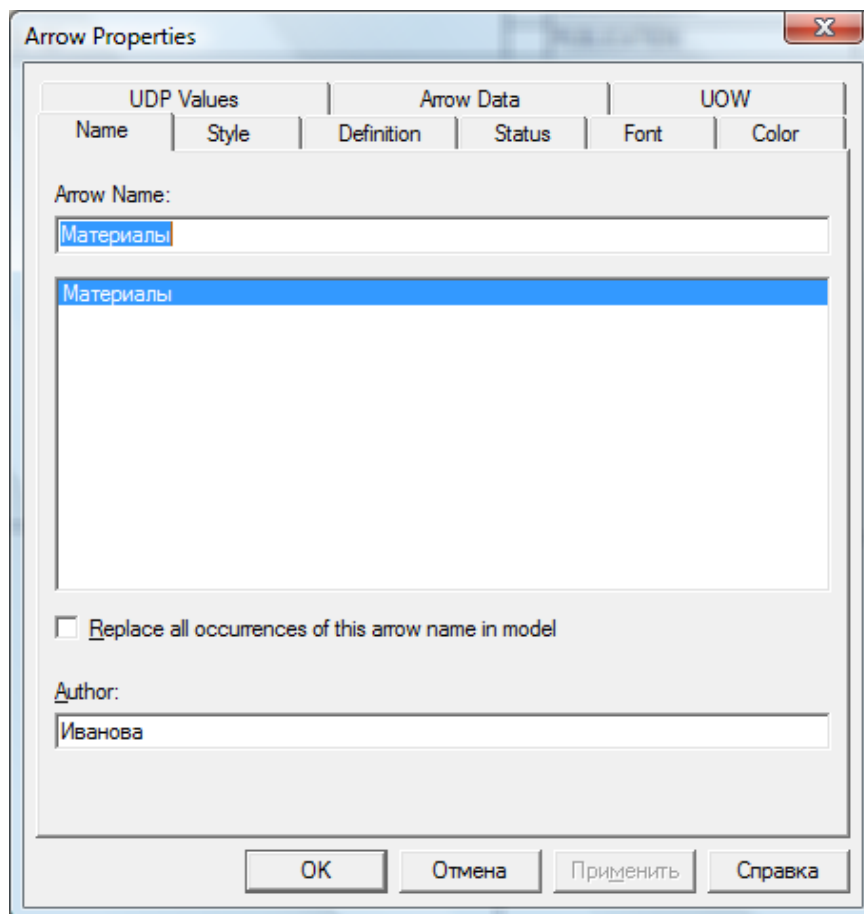


Рис. 10. Окно установки всех свойств стрелки

3. Аналогично задайте имена и определения другим стрелкам в соответствии с табл. 1.

Таблица 1

Стрелки контекстной диаграммы

<i>Arrow Name (Имя стрелки)</i>	<i>Arrow Definition (Определение стрелки)</i>
1	2
Материалы	Материалы, сырье, комплектующие изделия, которые необходимы для производства готовой продукции
Заказы клиентов	Заказы клиентов на изготовление готовой продукции

Окончание табл. 1

1	2
Процедуры и правила работы	Нормы расхода материалов, правила их закупки и расчетов с поставщиками, процедуры производства, критерии производительности и т.д.
Персонал предприятия	Все работники предприятия: руководитель, работники отделов снабжения, сбыта, производства, кладовщик
Готовая продукция	Готовые изделия, переданные клиентам

Свойства работ и стрелок автоматически помещаются в соответствующие словари. Для просмотра словарей функциональных блоков и стрелок следует выбрать соответственно меню *Dictionary* \Rightarrow *Activity...* (*Словарь* \Rightarrow *Работа*) и *Dictionary* \Rightarrow *Arrow...* (*Словарь* \Rightarrow *Стрелка*).

По модели можно создавать отчеты, содержащие выбранные пользователем свойства модели. Для этого следует выбрать команду меню *Tools* \Rightarrow *Reports* \Rightarrow *Model Report* (*Инструменты* \Rightarrow *Отчеты* \Rightarrow *Отчет по модели*).

Упражнение 4

1. Просмотрите словари работ и стрелок созданной диаграммы, выбрав команды меню *Dictionary* \Rightarrow *Activity...* (*Словарь* \Rightarrow *Работа*) и *Dictionary* \Rightarrow *Arrow...* (*Словарь* \Rightarrow *Стрелка*).

2. Для создания отчета вызовите команду меню *Tools* \Rightarrow *Reports* \Rightarrow *Model Report* (*Инструменты* \Rightarrow *Отчеты* \Rightarrow *Отчет модели*). В появившемся диалоговом окне (рис. 11) отметьте все свойства и нажмите кнопку *Preview* (*Предварительный просмотр*). Отобразится окно *Model Report Preview* (*Предварительный просмотр отчета модели*) (рис. 12).

3. Закройте окно отчета, нажав на кнопку *Close* (*Заккрыть*), и вернитесь к основному окну программы.

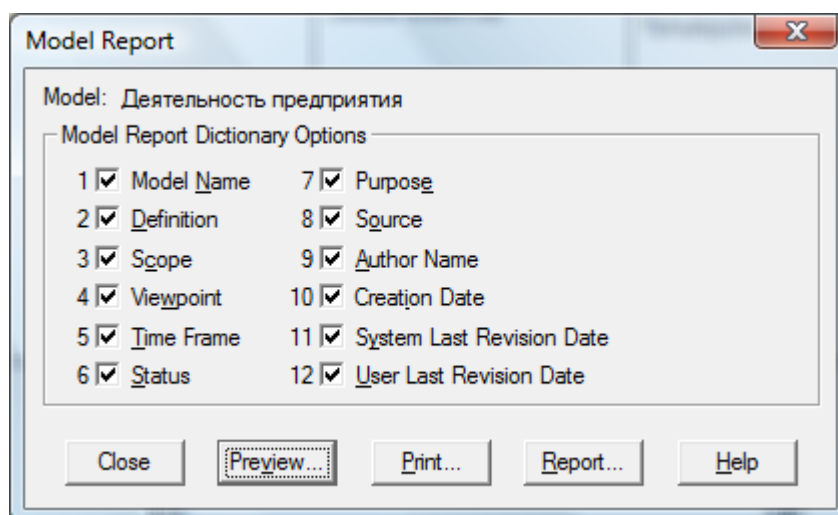
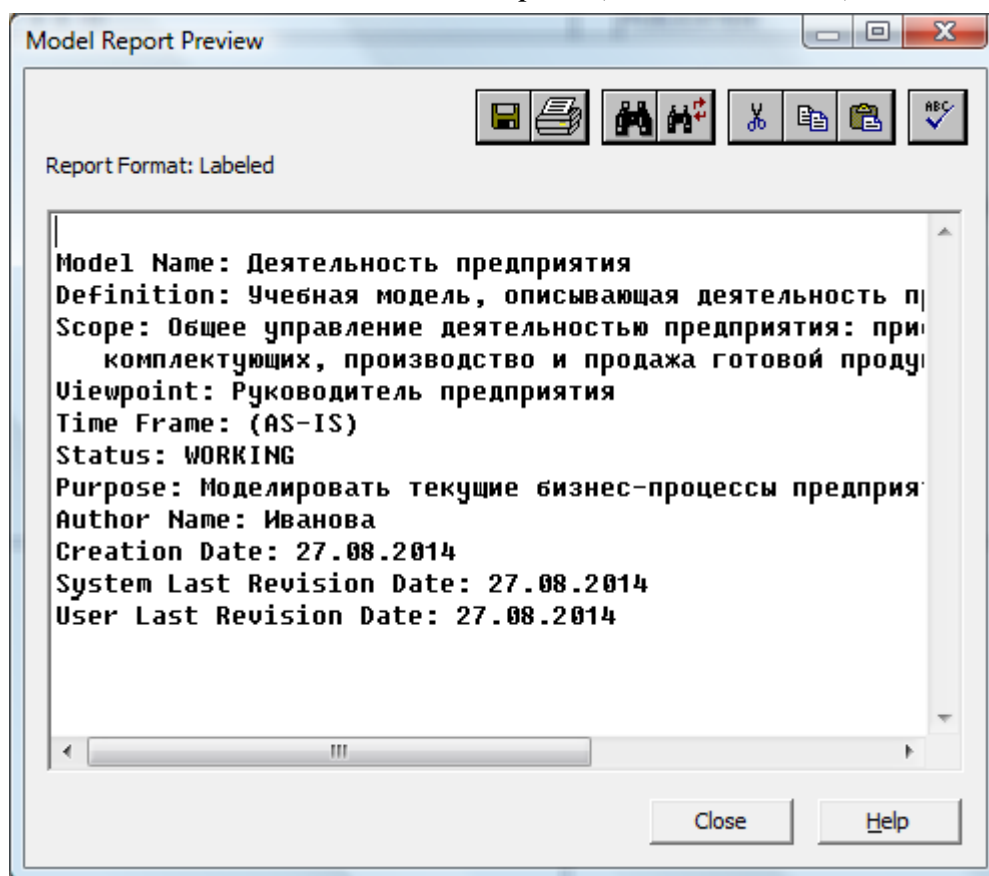
Рис. 11. Окно *Model Report* (Отчет модели)

Рис. 12. Окно предварительного просмотра отчета модели

Создание надписей на пространстве диаграммы

Надписи добавляются с помощью кнопки *Text Tool* (Инструмент Текст)

T на панели инструментов *Model* (Модель).

Упражнение 5

Выберите инструмент *Текст* на панели инструментов *Модель* и щелкните левой кнопкой мыши в левом нижнем углу диаграммы. На экране появится окно (рис. 13), в котором установите переключатель в положение *Purpose* (*Цель*). В поле ввода текста появится цель, которая была задана в окне свойств модели, т.е. строка «*Моделировать текущие бизнес-процессы предприятия*». Измените текст на «*Цель моделирования: Моделировать текущие бизнес-процессы предприятия*» и нажмите *ОК*. Результат приведен на рис. 14.

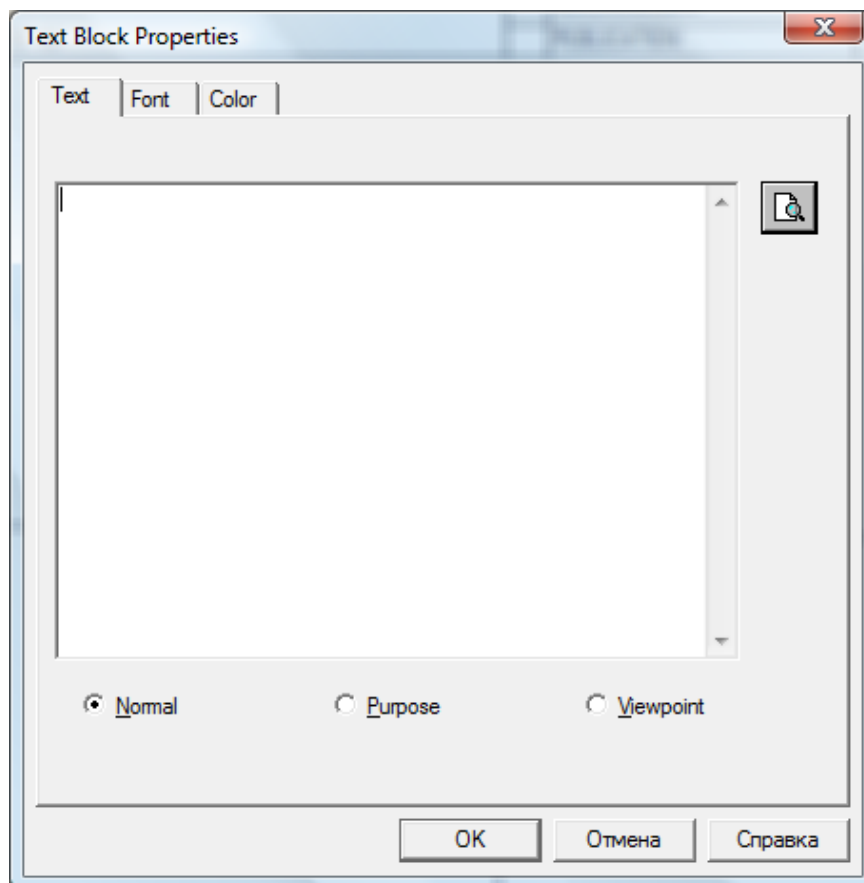


Рис. 13. Окно *Text Block Properties* (*Свойства текста*)

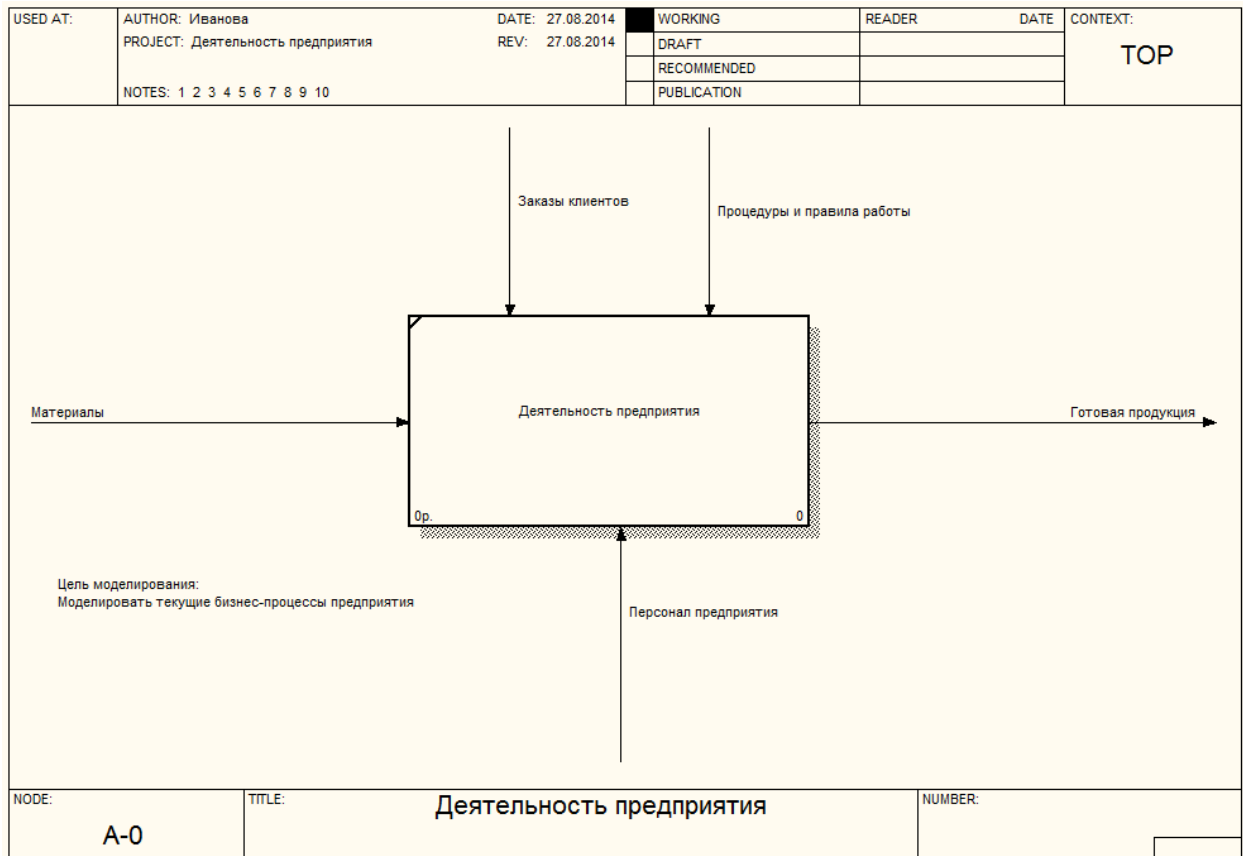


Рис. 14. Результат выполнения лабораторной работы

Вопросы для самопроверки

1. Что собой представляют *CASE*-технологии?
2. Назначение программы *CA ERwin Process Modeler*.
3. В чем заключается методология *IDEF0*?
4. Какие типы диаграмм может содержать модель в методологии *IDEF0*?
5. Основные составляющие модели *IDEF0* (работы, стрелки).
6. Как задаются свойства работ и стрелок в *CA ERwin Process Modeler*?
7. Как создать отчет модели в *CA ERwin Process Modeler*?

ПРАКТИЧЕСКАЯ РАБОТА № 2

Тема работы: Создание диаграмм декомпозиции в среде *CA Erwin Process Modeler*.

Диаграммы декомпозиции предназначены для детализации работы верхнего уровня и содержат дочерние работы, имеющие общую родительскую работу.

Работы на диаграммах декомпозиции обычно располагаются по диагонали от левого верхнего угла к правому нижнему (рис. 15). В левом верхнем углу располагается самая важная работа или работа, выполняемая по времени первой. Далее вправо вниз располагаются менее важные или выполняемые позже работы. Такое расположение облегчает чтение диаграмм. Каждая из работ на диаграмме декомпозиции может быть в свою очередь декомпозирована. На диаграмме декомпозиции работы нумеруются автоматически слева направо. Номер работы показывается в правом нижнем углу.

При декомпозиции работы входящие в нее и исходящие из нее стрелки (кроме стрелки вызова) автоматически появляются на диаграмме декомпозиции (миграция стрелок), но при этом не касаются работ. Такие стрелки называются несвязанными и воспринимаются как синтаксическая ошибка. Их следует довести до необходимой стороны работы.

Для связи работ между собой используются *внутренние стрелки*, т.е. стрелки, которые не касаются границы диаграммы, а начинаются у одной и кончаются у другой работы.

В *IDEF0* различают пять типов связей работ:

1. *Связь по входу* – стрелка выхода вышестоящей работы направляется на вход нижестоящей.

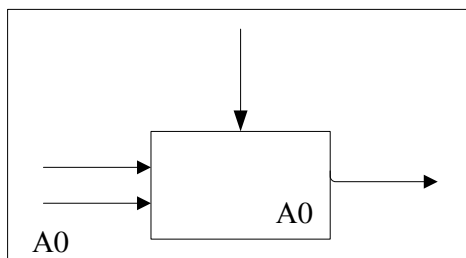
2. *Связь по управлению* – выход вышестоящей работы направляется на управление нижестоящей. Связь по управлению показывает доминирование вышестоящей работы. Данные или объекты выхода вышестоящей работы не меняются в нижестоящей.

3. *Обратная связь по входу* – выход нижестоящей работы направляется на вход вышестоящей. Такая связь, как правило, используется для описания циклов.

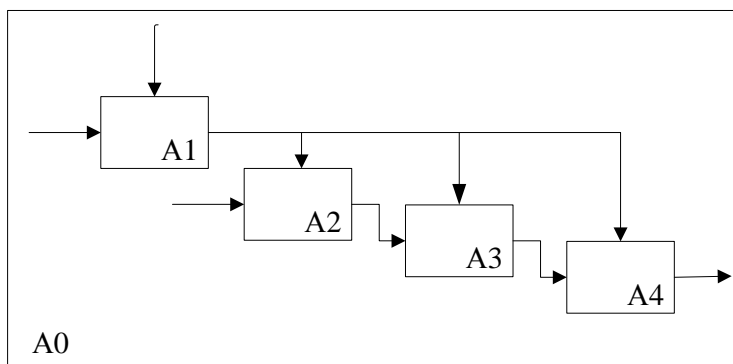
4. *Обратная связь по управлению* – выход нижестоящей работы направляется на управление вышестоящей. Обратная связь по управлению часто свидетельствует об эффективности бизнес-процесса.

5. *Связь выход-механизм* – выход одной работы направляется на механизм другой. Эта взаимосвязь используется реже остальных и показывает, что одна работа подготавливает ресурсы, необходимые для проведения другой работы.

Общее представление



Более детальное представление



Верхняя диаграмма является родительской для нижней диаграммы

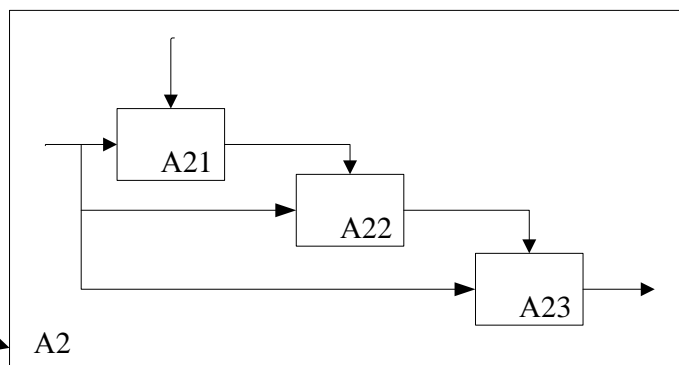


Рис. 15. Декомпозиция диаграмм

Разветвляющиеся и сливающиеся стрелки. Одни и те же данные или объекты, порожденные одной работой, могут использоваться сразу в нескольких других работах. С другой стороны, стрелки, порожденные в разных работах, могут представлять собой одинаковые или однородные данные или объекты, которые в дальнейшем используются или перерабатываются в одном месте.

Для моделирования таких ситуаций в методологии *IDEFO* используются разветвляющиеся и сливающиеся стрелки.

Смысл разветвляющихся и сливающихся стрелок передается именовани-ем каждой ветви стрелок. Если стрелка именована до разветвления, а после разветвления ни одна из ветвей не именована, то подразумевается, что каждая ветвь моделирует те же данные или объекты, что и ветвь до разветвления. Если при этом какая-либо ветвь после разветвления осталась неименованной, то подразумевается, что она моделирует те же данные или объекты, что и ветвь до разветвления.

Правила именования сливающихся стрелок полностью аналогичны – ошибкой будет считаться стрелка, которая после слияния не именована, а до слияния не именована какая-либо из ее ветвей. Для именования отдельной ветви разветвляющихся и сливающихся стрелок следует выделить на диаграмме только одну ветвь, после этого вызвать редактор имени и присвоить имя стрелке. Это имя будет соответствовать только выделенной ветви.

Туннельные стрелки. Граничные стрелки, созданные на диаграмме декомпозиции нижнего уровня, изображаются в квадратных скобках и автоматически не появляются на диаграмме верхнего уровня (рис. 16). Для того чтобы созданная стрелка стала отображаться на родительской диаграмме, следует правой кнопкой мыши щелкнуть по квадратным скобкам стрелки и в контекстном меню выбрать команду *Arrow Tunnel (Стрелка туннеля)*. Появится диалоговое окно *Border Arrow Editor (Редактор граничной стрелки)* (рис. 17), в котором в поле *How do you want to resolve this tunnel? (Как вы хотите решить этот туннель?)* можно выбрать один из двух вариантов:

- *Resolve it to border arrow (Разрешить перенос граничной стрелки)* – стрелка появится на диаграмме верхнего уровня;
- *Change it to resolved rounded tunnel (Изменить, разрешить закруглить туннель)* – стрелка станет туннельной и не попадет на диаграмму верхнего уровня. Туннельная стрелка изображается с круглыми скобками на конце.

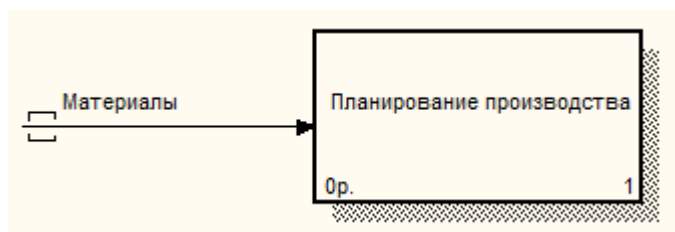


Рис. 16. Ненаследуемая входящая стрелка

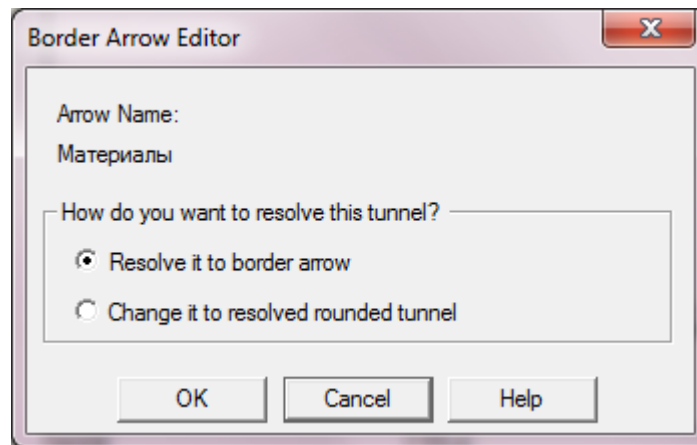


Рис. 17. Диалоговое окно выбора варианта туннеля стрелки

Туннельные стрелки используются для изображения малозначимых стрелок для родительской диаграммы. Если на какой-либо диаграмме нижнего уровня необходимо изобразить данные или объекты, которые не обрабатываются или не используются работами верхнего уровня, то их необходимо отображать туннельными стрелками.

Декомпозиция диаграмм в среде CA Erwin Process Modeler

Декомпозицию контекстной диаграммы в среде *CA Erwin Process Modeler* можно выполнить двумя способами:

1. Выделить функциональный блок на диаграмме и нажать кнопку *Go to Child Diagram* (Создать дочернюю диаграмму) ▼ на панели инструментов.
2. В проводнике модели щелкнуть правой кнопкой мыши на блоке и выбрать команду *Decompose* (Декомпозиция).

Для связывания стрелок входа, управления или механизма необходимо перейти в режим редактирования стрелок (кнопка → на панели инструментов модели), щелкнуть по наконечнику стрелки, а затем по соответствующему сегменту работы. Для связывания стрелки выхода необходимо перейти в режим редактирования стрелок, щелкнуть по сегменту выхода работы и по стрелке.

Для рисования внутренней стрелки необходимо в режиме рисования стрелок щелкнуть по сегменту (например, выхода) одной работы, а затем по сегменту (например, входа) другой.

Упражнение 1

Проведите декомпозицию контекстной диаграммы, созданной в предыдущей лабораторной работе. Диаграмма декомпозиции должна содержать 3 ра-

боты: *Планирование производства, Получение и хранение, Производство*. Для этого:

1. Запустите *CA Erwin Process Modeler* и откройте файл с контекстной диаграммой, созданной вами в предыдущей лабораторной работе.
2. На панели инструментов модели нажмите на кнопку *Go to Child Diagram* (Создать дочернюю диаграмму) ▼, появится диалоговое окно (рис. 18), в котором установите количество работ 3 и нажмите кнопку *OK*. Автоматически будет создана диаграмма декомпозиции (рис. 19).

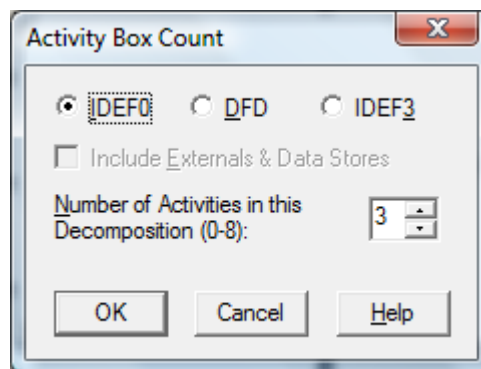


Рис. 18. Окно задания числа дочерних диаграмм



Рис. 19. Результат декомпозиции контекстной диаграммы



3. Правой кнопкой мыши щелкните по верхней левой созданной работе и в контекстном меню выберите *Name (Имя)*. В появившемся окне *Activity Properties (Свойства работы)* введите *Планирование производства* и нажмите на кнопку *Применить*. На вкладке *Definition (Определение)* введите *Обработка заказов клиентов, составление плана производства* и щелкните по кнопке *ОК*.

4. Аналогично задайте имена и определения другим работам в соответствии с табл. 2.

Таблица 2

Работы диаграммы декомпозиции A0

<i>Activity Name (Имя работы)</i>	<i>Definition (Определение)</i>
Планирование производства	Обработка заказов клиентов, составление плана производства
Получение и хранение	Получение материалов, проверка документов, хранение материалов и готовой продукции
Производство	Производство готовых изделий

5. Для связывания стрелки входа *Материалы* с работой *Планирование производства* (рис. 20) перейдите в режим редактирования стрелки (кнопка  на панели инструментов модели). Щелкните левой кнопкой мыши по наконечнику стрелки, а затем по необходимой стороне работы. Для выхода из режима редактирования стрелки нажмите на кнопку  на панели инструментов. Аналогично свяжите стрелку управления *Заказы клиентов* с работой *Планирование производства*.

6. Свяжите стрелку выхода *Готовая продукция* с работой *Получение и хранение* (рис. 20), вначале щелкнув в режиме редактирования стрелки по стороне работы, а затем – по стрелке.

7. Постройте разветвления стрелок *Процедуры и правила работы* и *Персонал предприятия* в соответствии с рис. 21. Для этого перейдите в режим редактирования стрелок, а затем щелкайте по существующей стрелке и необходимой работе.

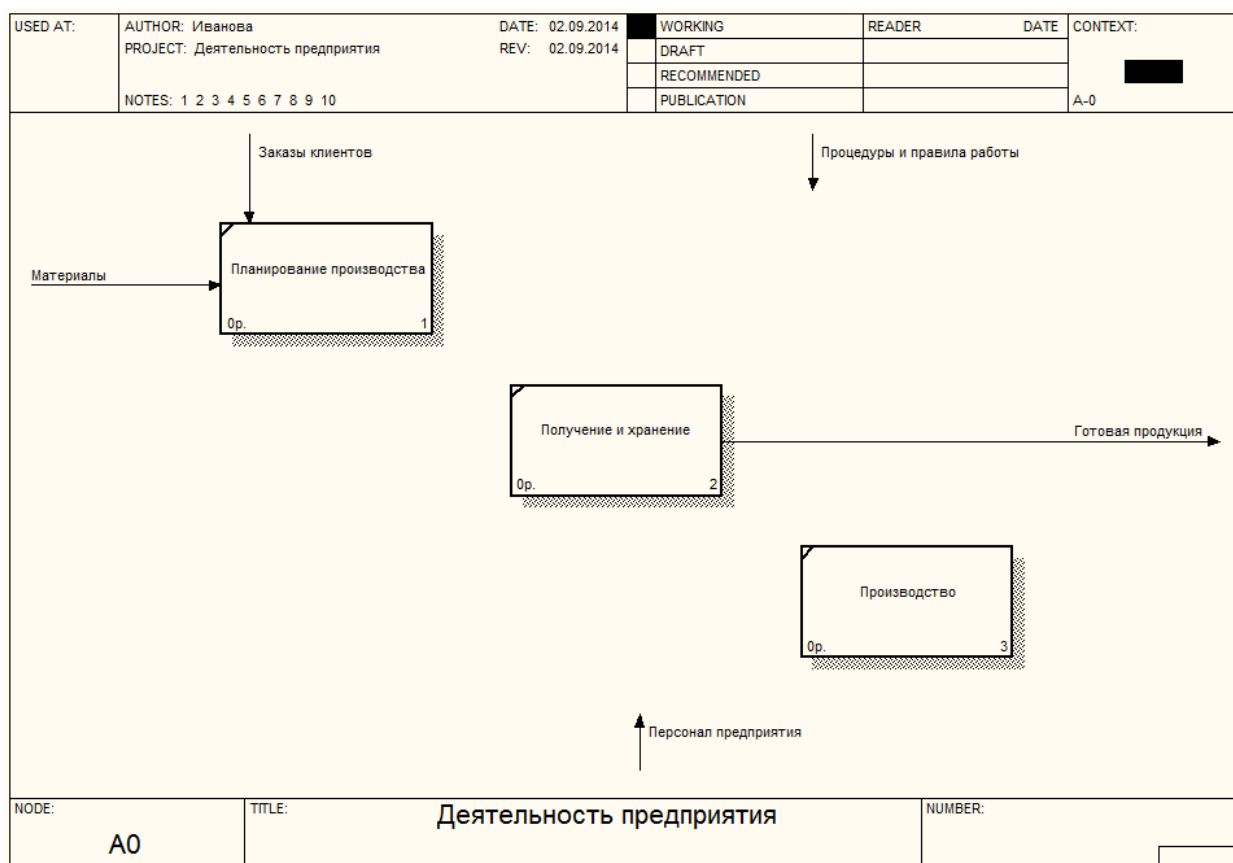


Рис. 20. Результат связывания стрелок

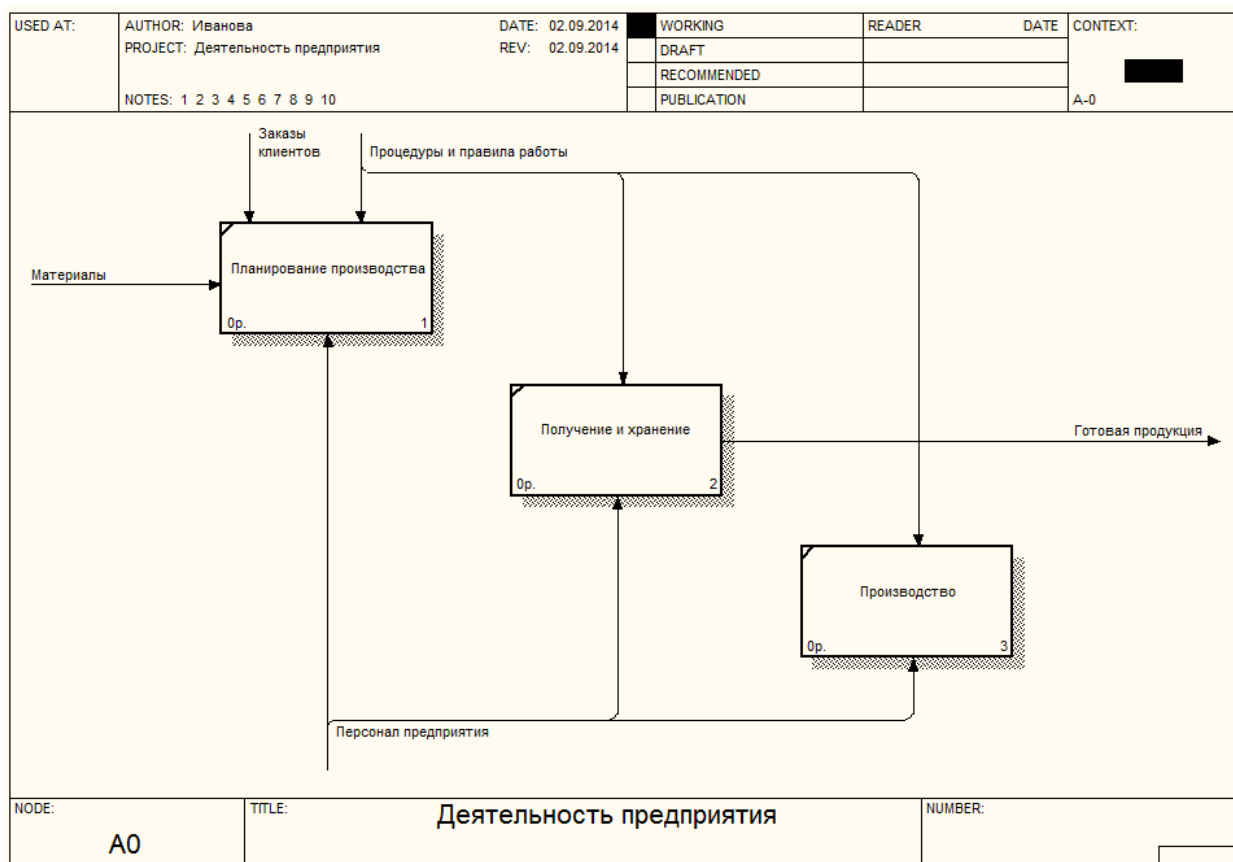


Рис. 21. Результат построения стрелок разветвления

8. Создайте внутреннюю стрелку *Поставки материалов*, идущую из блока *Планирование производства* к блоку *Получение и хранение* (рис. 22). Для нее в окне *Arrow Properties* (*Свойства стрелки*) на вкладке задайте *Definition* (*Определение*) – *Сырье и материалы, закупленные у поставщиков*. Щелкните правой кнопкой мыши по стрелке и в контекстном меню выберите команду *Squiggle* (*Волнистая ссылка*).

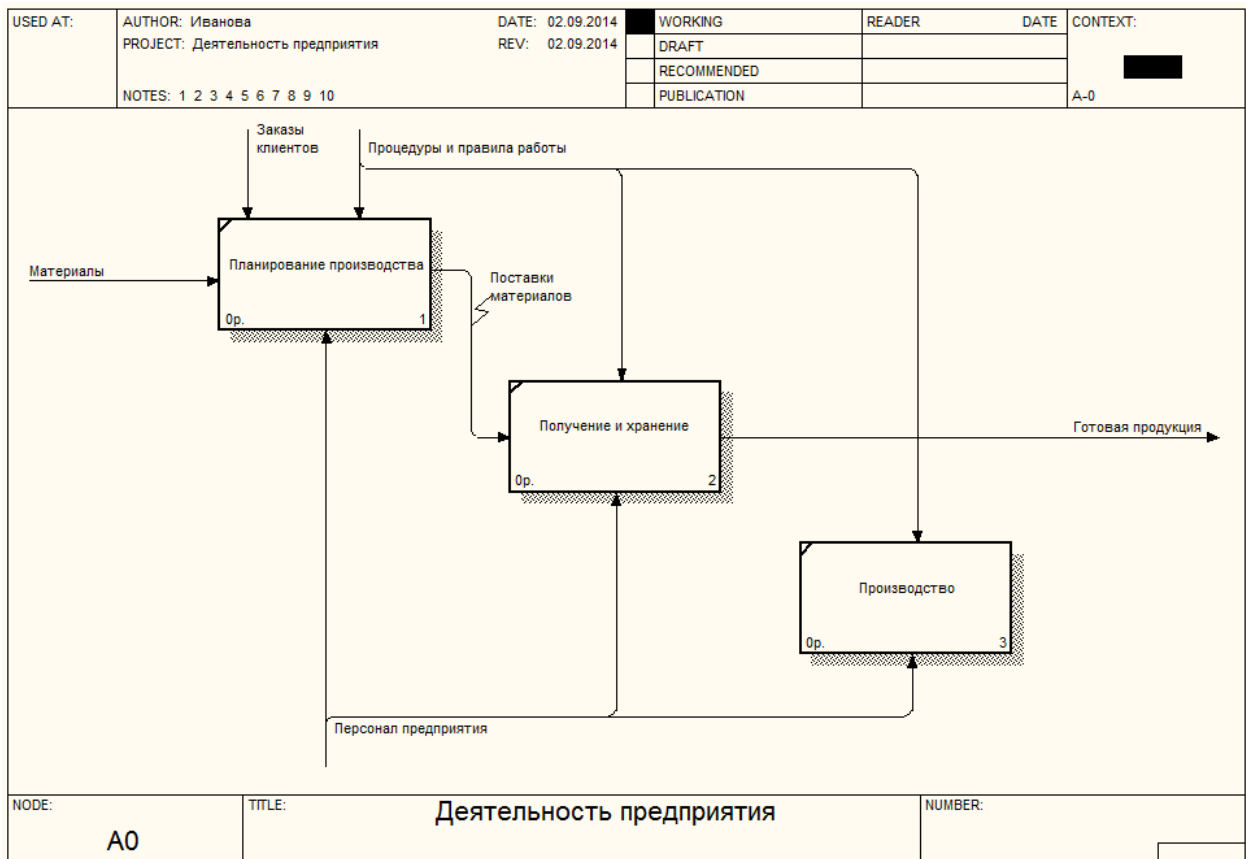


Рис. 22. Создание внутреннее стрелки *Поставки материалов*

9. Аналогично создайте внутренние стрелки в соответствии с табл. 3 и рис. 23.

10. Создайте стрелку обратной связи (по входу) *Готовые изделия*, идущую от блока *Производство* к блоку *Получение и хранение* (рис. 24). Для нее в окне *Arrow Properties* (*Свойства стрелки*) на вкладке *Definition* (*Определение*) задайте *Готовые изделия, переданные на склад*, на вкладке *Style* (*Стиль*) в поле *Thickness* (*Толщина*) выберите более толстый шаблон линии. Щелкните правой кнопкой мыши по созданной линии и установите опции *Extra Arrowhead* (*Дополнительный наконечник стрелки*), *Squiggle* (*Волнистая ссылка*).

Таблица 3

Внутренние стрелки диаграммы

<i>Arrow Name</i> (Имя стрелки)	Блок начала стрелки	Блок конца стрелки	<i>Arrow Definition</i> (Определение стрелки)
План производства	Планирование производства	Производство	Производственный план по выпуску готовой продукции
Материалы для производства	Получение и хранение	Производство	Сырье и материалы, необходимые в производстве

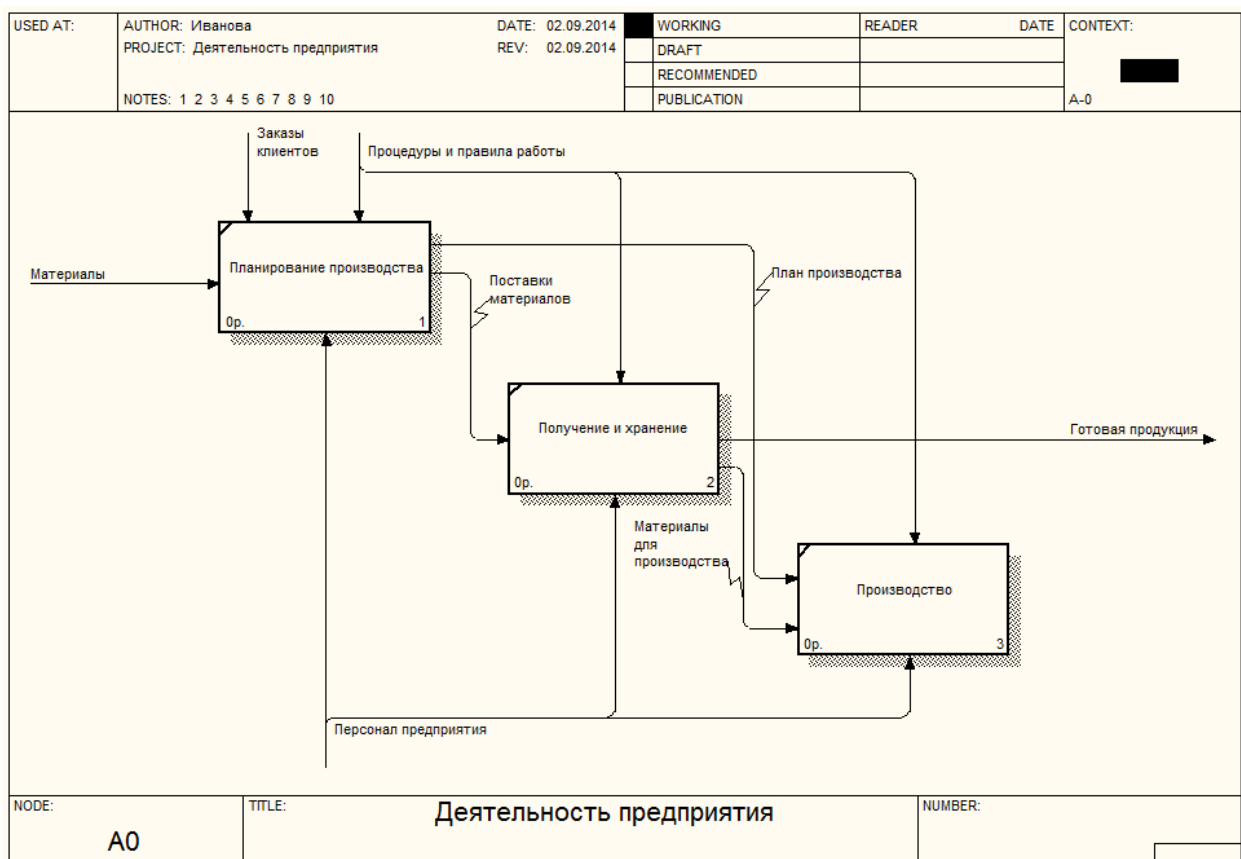
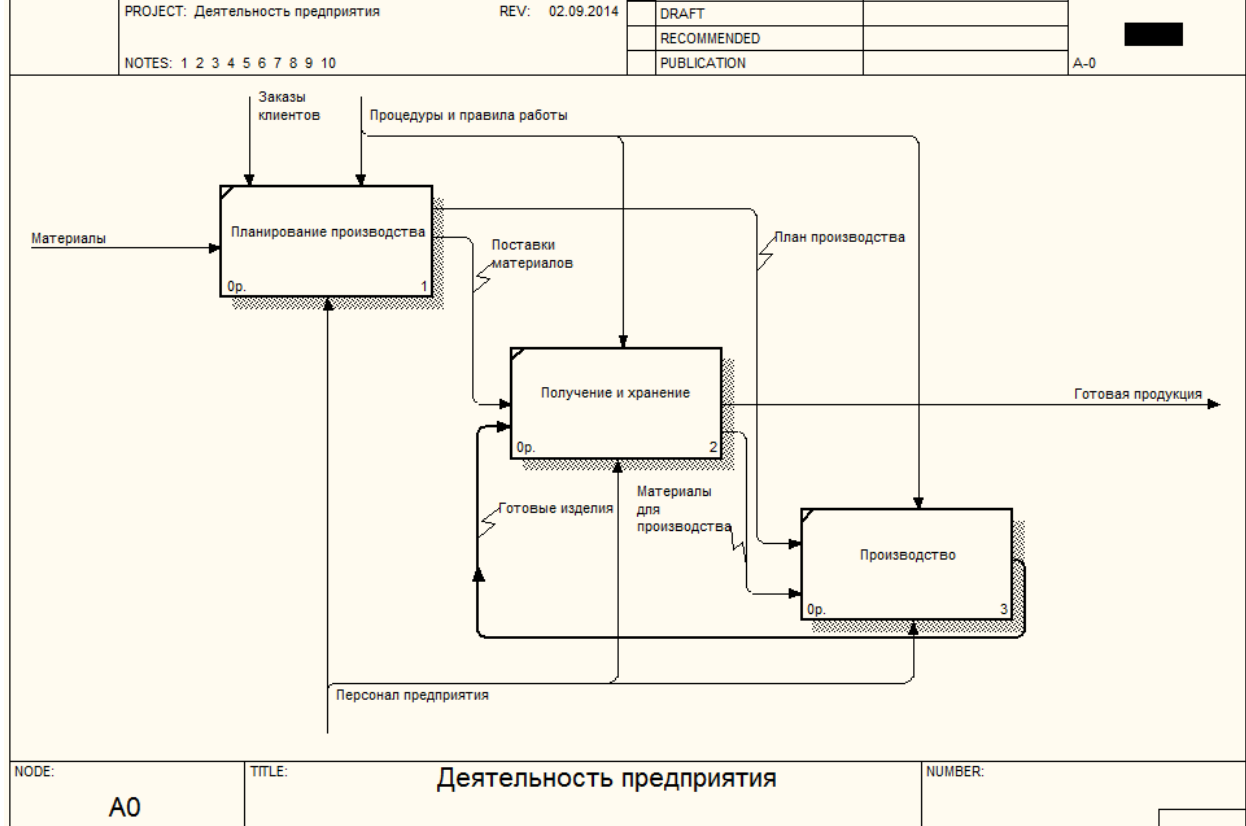


Рис. 23. Создание внутренних стрелок диаграммы

11. Аналогично создайте стрелку обратной связи *Информация о наличии материалов на складе*, идущую от работы *Получение и хранение* к работе *Планирование производства* (рис. 24). Для нее задайте *Definition (Определение)* –

USED AT:	AUTHOR: Иванова	DATE: 02.09.2014	WORKING	READER	DATE	CONTEXT:
----------	-----------------	------------------	---------	--------	------	----------



13. Аналогично задайте имена другим ветвям стрелки *Персонал предпри-*

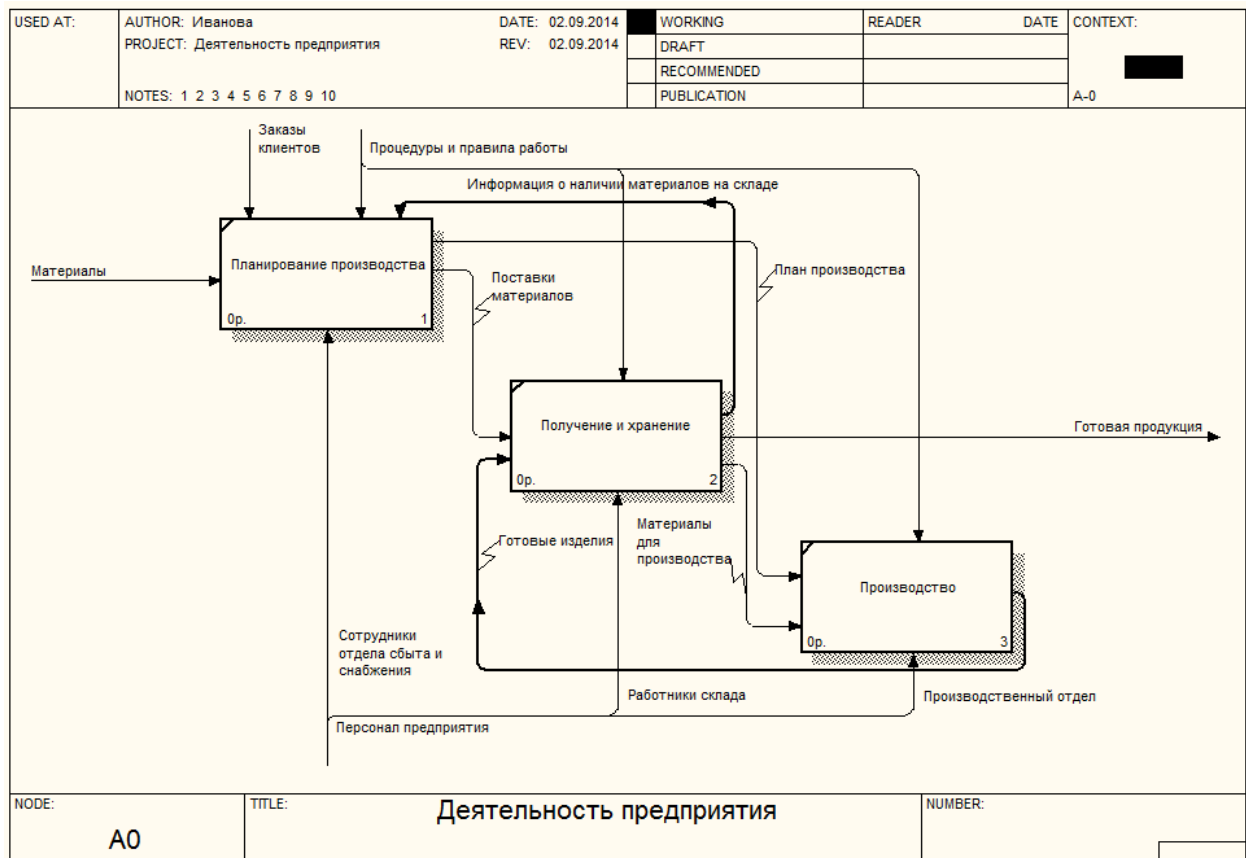


Рис. 25. Результат декомпозиции контекстной диаграммы

Упражнение 2

Проведите декомпозицию работы *Планирование производства* в соответствии с табл. 4, 5 и рис. 26.

Таблица 4

Работы диаграммы декомпозиции A1

<i>Name (Имя)</i>	<i>Definition (Определение)</i>
Обработка заказов	Прием заказов клиентов на изготовление изделий
Определение потребностей и закупка материалов	Определение объема необходимых материалов для выполнения заказов, закупка у поставщиков сырья и материалов
Планирование загрузки производственных мощностей	Составление плана загрузки производственных мощностей и производства

Таблица 5

Стрелки диаграммы декомпозиции *AI*

<i>Arrow Name</i> (Имя стрелки)	Тип стрелки	Начало стрелки	Конец стрелки
1	2	3	4
Материалы	Стрелка входа		Определение потребностей и закупка материалов
Заказы клиентов	Стрелка управления		Обработка заказов
Процедуры и правила работы	Стрелка управления		Обработка заказов Определение потребностей и закупка материалов Планирование загрузки производственных мощностей
Поставки материалов	Стрелка выхода	Определение потребностей и закупка материалов	
Информация о наличии материалов на складе	Стрелка управления		Определение потребностей и закупка материалов
Сотрудники отдела сбыта и снабжения	Механизм		Обработка заказов Определение потребностей и закупка материалов Планирование загрузки производственных мощностей
План производства	Стрелка выхода	Планирование загрузки производственных мощностей	

Окончание таблицы 5

1	2	3	4
Примерный план выпуска на период	Внутренняя	Обработка заказов	Определение потребностей и закупка материалов
Информация о наличии материалов для производства	Внутренняя стрелка	Определение потребностей и закупка материалов	Планирование загрузки производственных мощностей

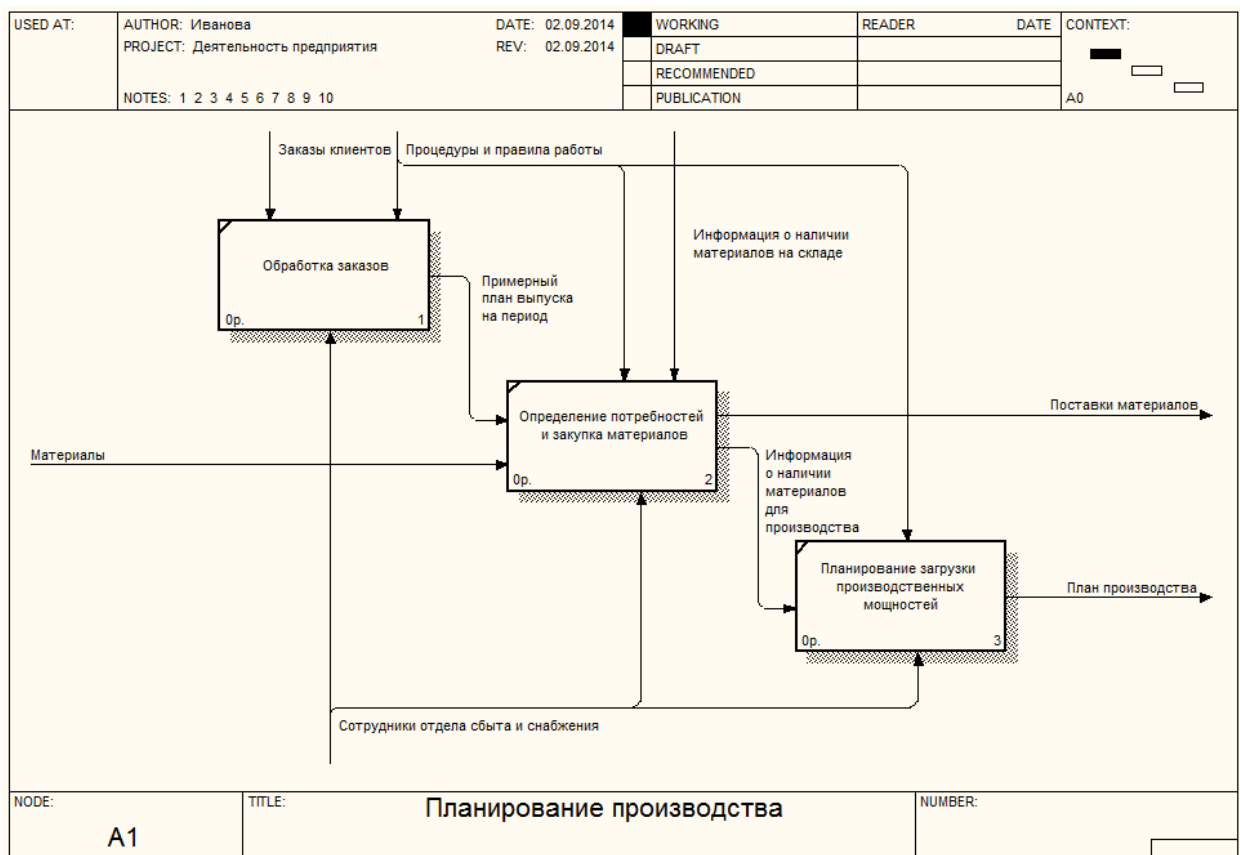


Рис. 26. Диаграмма декомпозиции A1

Упражнение 3

Проведите декомпозицию работы *Получение и хранение* в соответствии с табл. 6, 7 и рис. 27.

Таблица 6

Работы диаграммы декомпозиции A2

<i>Name (Имя)</i>	<i>Definition (Определение)</i>
Проверка накладных, получение материалов	Получение материалов и сырья от поставщиков
Хранение	Хранение материалов и готовых изделий

Таблица 7

Стрелки диаграммы декомпозиции A2

<i>Arrow Name (Имя стрелки)</i>	Тип стрелки	Начало стрелки	Конец стрелки
Поставки материалов	Стрелка входа		Проверка накладных, получение материалов
Готовые изделия	Стрелка входа		Хранение
Процедуры и правила работы	Стрелка управления		Проверка накладных, получение материалов Хранение
Информация о наличии материалов на складе	Стрелка выхода	Хранение	
Готовая продукция	Стрелка выхода	Хранение	
Материалы для производства	Стрелка выхода	Хранение	
Работники склада	Механизм		Проверка накладных, получение материалов Хранение
Полученные материалы	Внутренняя	Проверка накладных, получение материалов	Хранение



Рис. 27. Диаграмма декомпозиции A2

Вопросы для самопроверки

1. Что такое диаграмма декомпозиции?
2. Как располагаются и именуется работы на диаграммах декомпозиции?
3. Назовите типы связей работ на диаграммах декомпозиции.
4. Расскажите правила именования разветвляющихся и сливающихся стрелок.
5. Для чего используются туннельные стрелки? Как их создать?
6. Как в среде *CA Erwin Process Modeler* выполнить декомпозицию работы?

ПРАКТИЧЕСКАЯ РАБОТА № 3

Тема работы: Создание диаграммы дерева узлов и FEO-диаграммы.

Диаграмма дерева узлов показывает иерархию работ в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между работами (стрелки). Процесс создания модели работ является итерационным, работы могут менять свое расположение в дереве узлов многократно. Чтобы не запутаться и проверить способ декомпозиции, следует после каждого изменения создавать диаграмму дерева узлов. При создании дерева узлов следует задавать имя диаграммы, так как в нескольких диаграммах в качестве корня дерева узлов может использоваться одна и та же работа. Тогда все эти диаграммы получат одинаковый номер (номер узла + постфикс *N*, например, *AON*) и в списке открытых диаграмм (меню *Windows*) их можно будет различить только по имени.

Упражнение 1

1. Запустите *CA Erwin Process Modeler* и откройте файл, созданный в предыдущих работах.
2. В главном меню выберите команду *Diagram ⇒ Add Node Tree* (*Диаграмма ⇒ Добавить дерево узлов*). На экране появится окно *Мастера создания дерева узлов* (рис. 28), в котором укажите имя диаграммы *Деятельность предприятия*, выберите корневую диаграмму *A0*, задайте *Number of levels* (*Количество уровней*) – 3 и нажмите *Далее*.

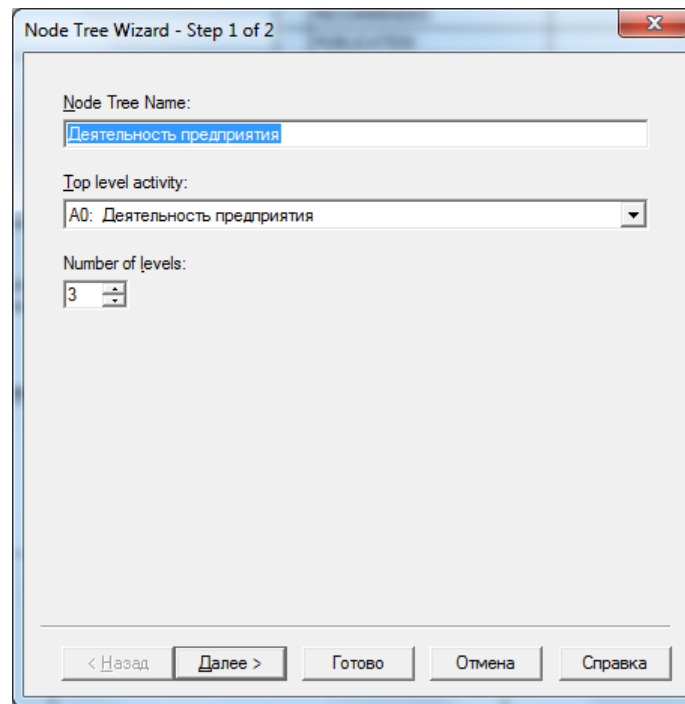


Рис. 28. Первый шаг Мастера создания дерева узлов

3. Во втором окне Мастера создания дерева узлов оставьте опции, заданные по умолчанию и нажмите *Готово*. Результат представлен на рис. 29.

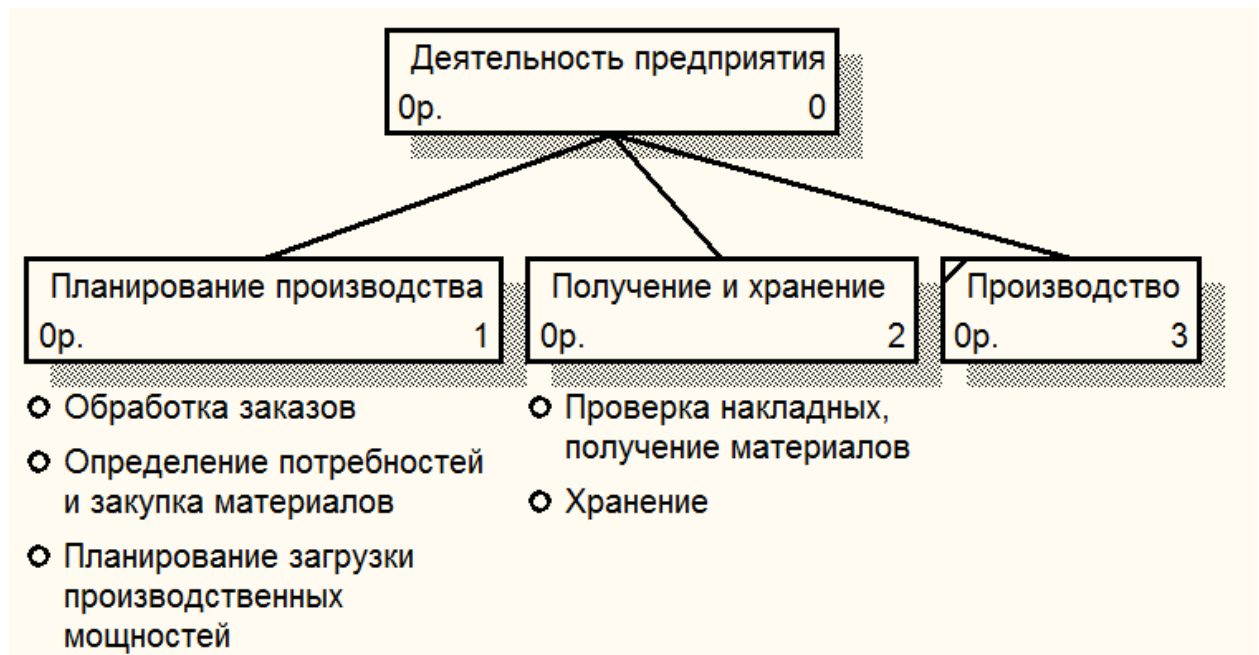


Рис. 29. Диаграмма дерева узлов

4. Создайте новую диаграмму дерева узлов. Для ее изменения щелкните правой кнопкой мыши на свободном месте диаграммы и в контекстном меню выберите команду *Node tree Diagram Properties* (Свойства диаграммы дерева узлов).

5. В окне *Node Tree Properties* (Свойства дерева узлов) на вкладке *Style* (Стиль) (рис. 30) уберите галочку в поле *Bullet last level* (Маркер последнего уровня) и нажмите *OK*. Результат представлен на рис. 31.

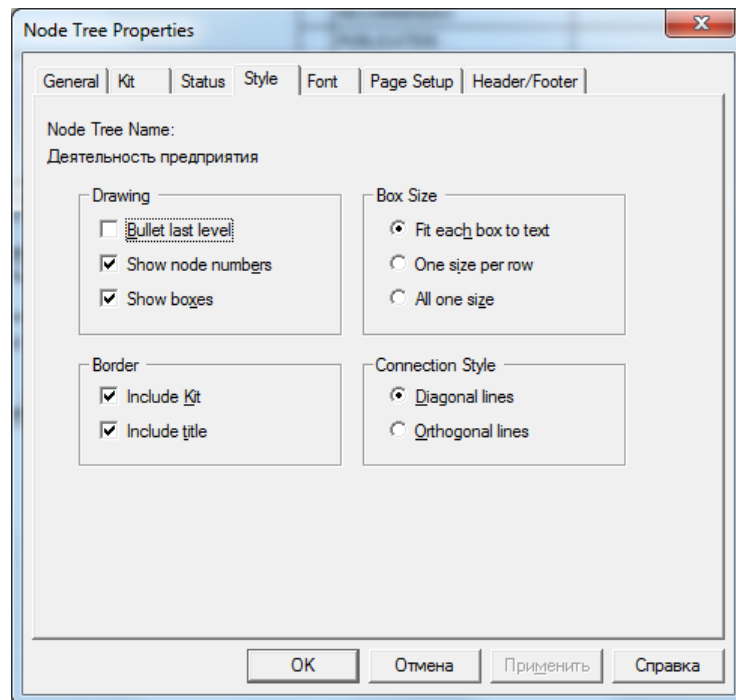


Рис. 30. Диалоговое окно *Node Tree Properties* (Свойства дерева узлов)

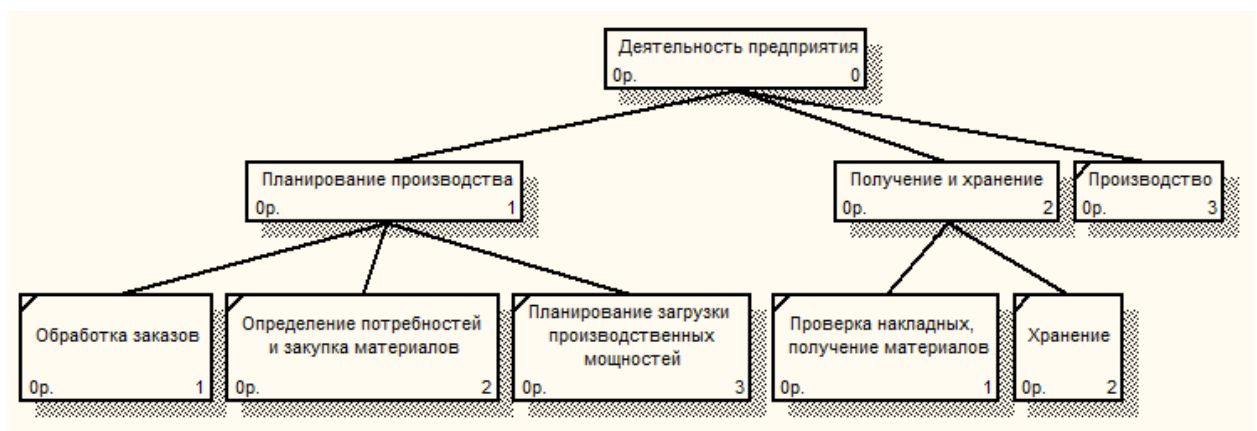


Рис. 31. Диаграмма дерева узлов

Диаграммы «только для экспозиции» (*FEO*-диаграммы) часто используются в модели для иллюстрации других точек зрения, для отображения отдельных деталей, которые не поддерживаются явно синтаксисом *IDEF0*. Диаграммы *FEO* позволяют нарушить любое синтаксическое правило поскольку являются просто картинками – копиями стандартных диаграмм и не включаются в анализ синтаксиса. Созданная диаграмма *FEO* получает номер, который генерируется автоматически (номер родительской диаграммы по узлу + постфикс *F*, напри-

мер, *AIF*), а далее следует заданное имя диаграммы и тип родительской диаграммы.

Упражнение 2

Создайте *FEO*-диаграмму работы *Планирование производства*, на которой будут отображены только стрелки данной работы. Для этого:

1. В главном меню выберите команду *Diagram* \Rightarrow *Add FEO diagram* (*Диаграмма* \Rightarrow *Добавить FEO-диаграмму*). В появившемся окне (рис. 32) введите имя – *FEO-диаграмма Планирование производства*, установите переключатель в положение *Decomposition Diagram*, в поле *Source Diagram Name* (*Имя диаграммы источника*) – *A1: Планирование производства* и нажмите *OK*. На *FEO*-диаграмму скопируется выбранная диаграмма модели.

2. Удалите стрелки в соответствии с рис. 33. Для удаления стрелки выделите ее левой кнопкой мыши и нажмите кнопку *Delete* (*Удалить*) на клавиатуре. В появившемся окне подтверждения удаления выберите *Да*.

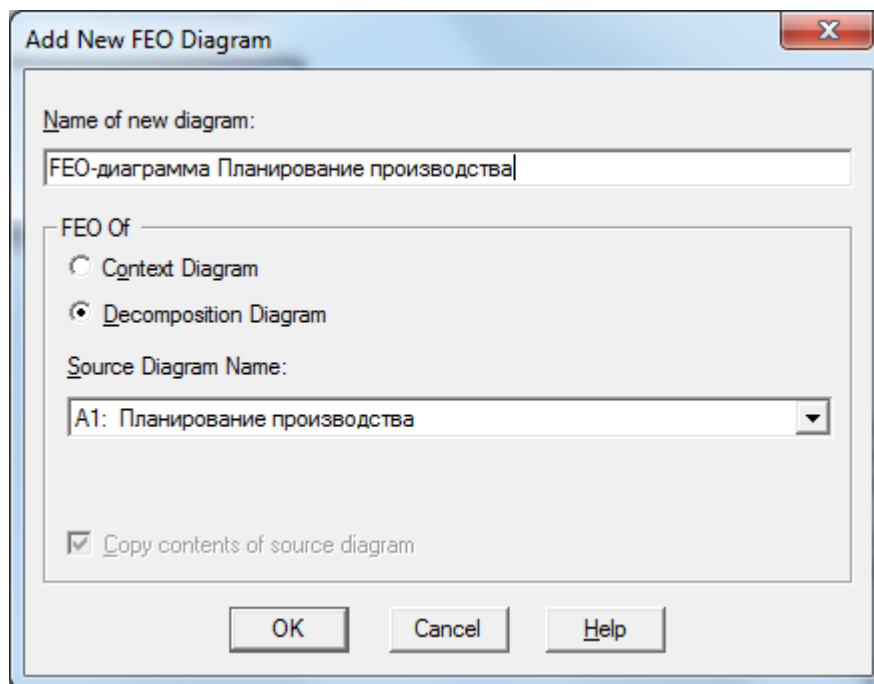


Рис. 32. Диалоговое окно *Add New FEO Diagram* (*Добавление новой FEO-диаграммы*)

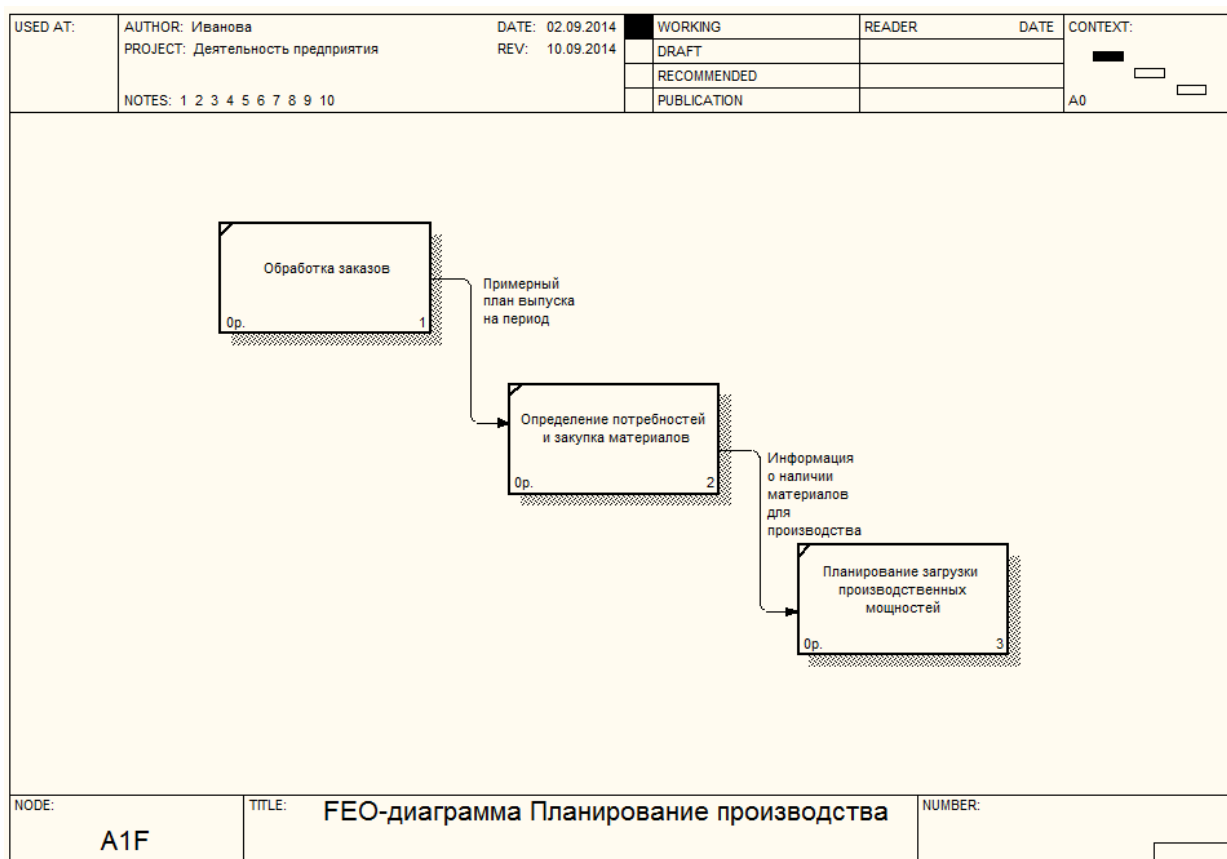
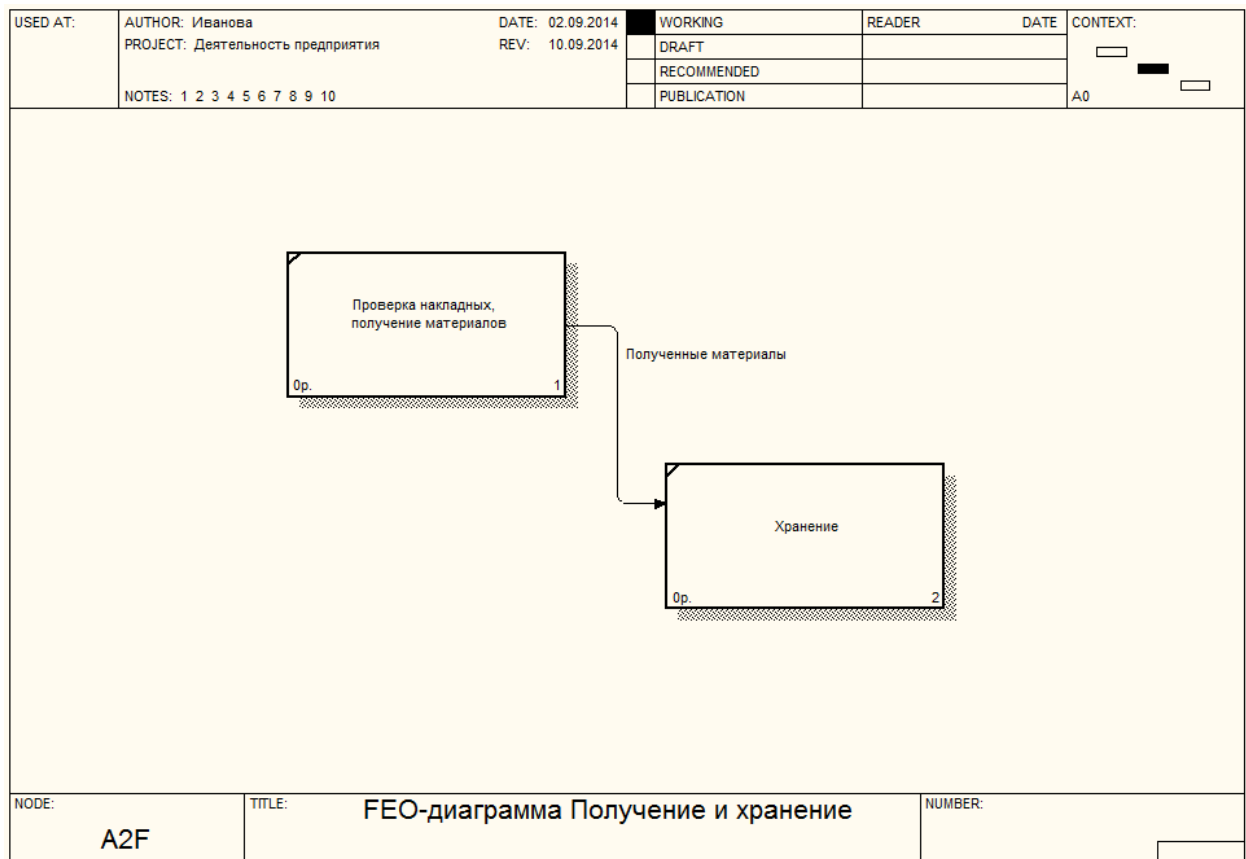


Рис. 33. FEO-диаграмма Планирование производства

Упражнение 3

Создайте FEO-диаграмму работы *Получение и хранение*, на которой оставьте только стрелки данной работы. Результат приведен на рис. 34.

Рис. 34. FEO-диаграмма *Получение и хранение***Упражнение 4**

С помощью методологии *IDEF0* представьте процесс учета товарно-материальных ценностей (ТМЦ) на складе и в бухгалтерии. Для этого:

1. Создайте новую модель *Учет ТМЦ* со следующими свойствами:

Model name (Имя модели) – Учет ТМЦ

Project (Проект) – Учет ТМЦ

Author (Автор) – Иванова (своя фамилия)

Author initials (Инициалы автора) – А.А. (свои инициалы)

Time Frame (Временные рамки) – AS-IS (Как есть)

Purpose (Цель) – Моделировать текущие бизнес-процессы учета товарно-материальных ценностей

Viewpoint (Точка зрения) – Бухгалтер

Definition (Определение) – Учебная модель, описывающая учет товаров на складе и в бухгалтерии

Scope (Область действия) – Учет товарно-материальных ценностей на складе и в бухгалтерии

2. В соответствии с табл. 8, 9 и рис. 35 создайте контекстную диаграмму.

Таблица 8

Работа контекстной диаграммы

<i>Name (Имя)</i>	<i>Definition (Определение)</i>
Учет ТМЦ	Учет товарно-материальных ценностей на складе и в бухгалтерии

Таблица 9

Стрелки контекстной диаграммы

<i>Arrow Name (Имя стрелки)</i>	Тип стрелки
Счет-фактура/накладная	Вход
Контроль за движением продукции	Управление
Персонал предприятия	Механизм
Составление отчетности	Выход

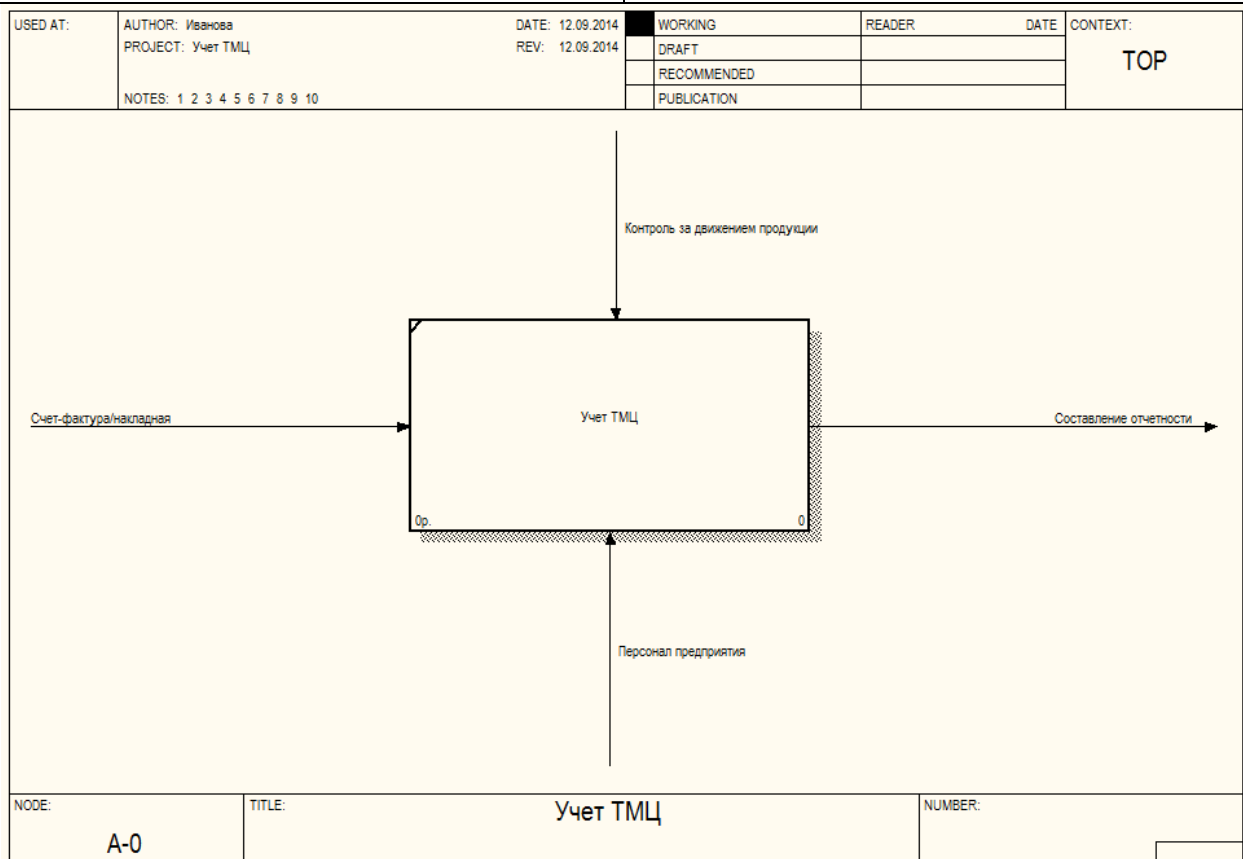


Рис. 35. Контекстная диаграмма Учет ТМЦ

3. В соответствии с табл. 10, 11 и рис. 36 создайте диаграмму декомпозиции.

4. Создайте диаграмму дерева узлов.

5. Создайте *FEO*-диаграмму работы *Учет ТМЦ*, на которой будут отображены только стрелки данной работы.

Таблица 10

Работы диаграммы декомпозиции *A0*

<i>Name (Имя)</i>	<i>Definition (Определение)</i>
Складской учет	Учет товаров на складе ведется материально-ответственным лицом в количественных и стоимостных измерениях
Бухгалтерский учет	На основе первичных документах, поступающих со склада, ведется бухгалтерский учет ТМЦ
Инвентаризация	Инвентаризация ТМЦ проводится в соответствии с рекомендациями МИНФИНА РФ

Таблица 11

Стрелки диаграммы декомпозиции *A0*

<i>Arrow Name (Имя стрелки)</i>	Тип стрелки	Начало стрелки	Конец стрелки
1	2	3	4
Счет-фактура/ накладная	Вход	Складской учет	
Товары	Вход	Складской учет	
Контроль за движением продукции	Управление		Складской учет Бухгалтерский учет Инвентаризация
Персонал предприятия	Механизм		Складской учет Бухгалтерский учет Инвентаризация
Реализованные товары	Выход	Складской учет	
Первичные документы	Внутренняя	Складской учет	Бухгалтерский учет

Окончание таблицы 11

1	2	3	4
Ведомость учета остатков	Обратная связь по входу	Бухгалтерский учет	Складской учет
Составление отчетности	Выход	Инвентаризация	
Данные отчетности	Внутренняя	Бухгалтерский учет	Инвентаризация
Инвентаризационная опись или акт	Обратная связь по входу	Инвентаризация	Бухгалтерский учет

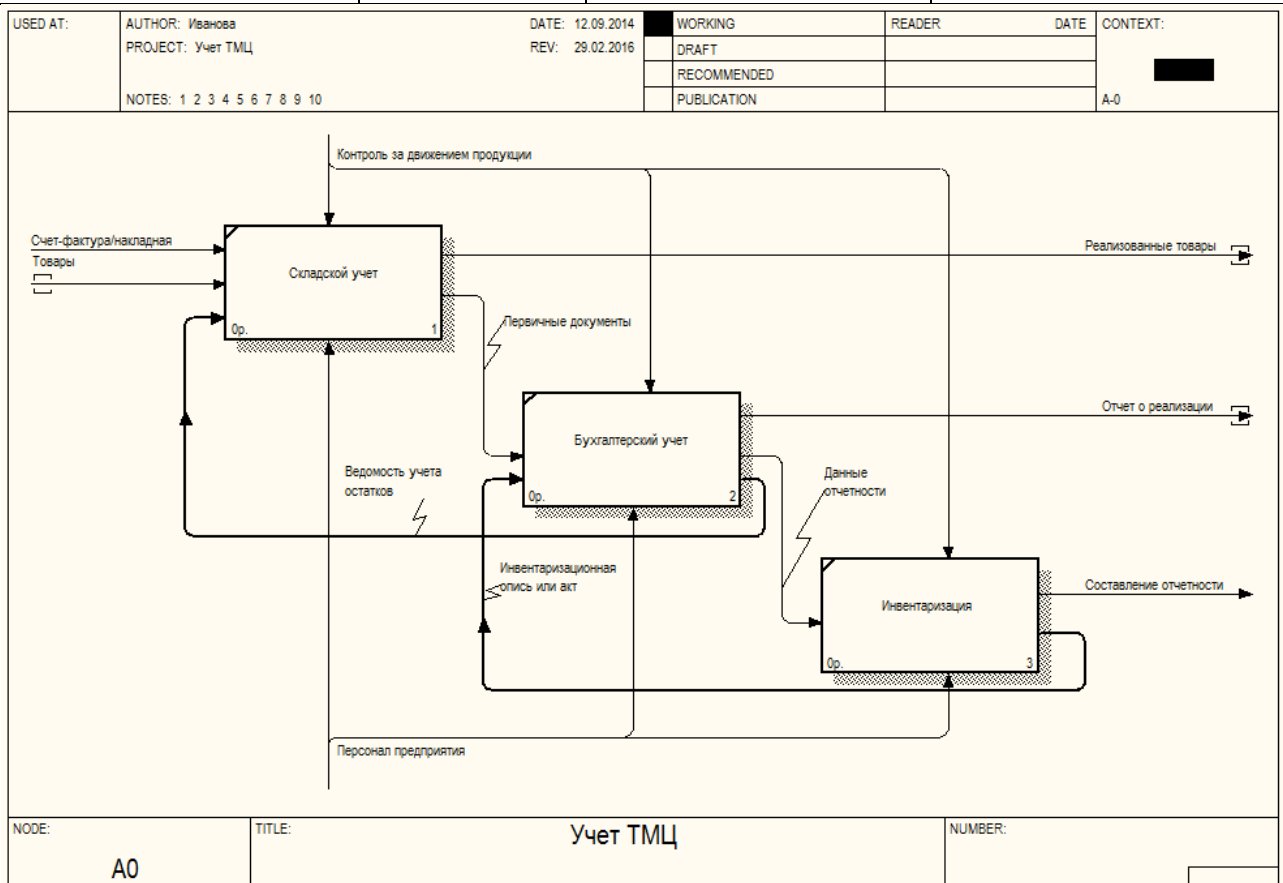


Рис. 36. Диаграмма декомпозиции A0

Вопросы для самопроверки

1. Что такое диаграмма дерева узлов?
2. Как создать диаграмму дерева узлов в среде *CA ERwin Process Modeler*?
3. Что собой представляют диаграммы для экспозиции (*FEO*-диаграммы)?
4. Как создать *FEO*-диаграмму в *CA ERwin Process Modeler*?

ПРАКТИЧЕСКАЯ РАБОТА № 4

Тема работы: Создание моделей в стандарте *DFD* (диаграммы потоков данных).

Диаграммы потоков данных (Data Flow Diagram – DFD) используются для описания документооборота и обработки информации. Модель информационной системы определяется как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи пользователю.

Диаграммы верхних уровней иерархии (контекстные диаграммы) определяют основные процессы или подсистемы ИС с внешними входами и выходами. Они детализируются при помощи диаграмм нижнего уровня. Такая декомпозиция продолжается до тех пор, пока не будет достигнут такой уровень декомпозиции, на котором процессы становятся элементарными и детализировать их далее невозможно.

DFD можно использовать как дополнение к модели *IDEF0* для более наглядного отображения текущих операций документооборота.

Основными компонентами диаграмм потоков данных являются:

- системы/подсистемы;
- процессы (функции обработки информации, работы);
- внешние сущности (*External references*);
- накопители данных (хранилище данных (*Data store*));
- потоки данных.

При построении модели сложной ИС она может быть представлена в самом общем виде на так называемой контекстной диаграмме в виде одной *системы* как единого целого, либо может быть декомпозирована на ряд *подсистем*.

В *DFD работы* представляют собой функции системы, преобразующие входные потоки данных в выходные. Работы изображаются прямоугольниками со скругленными углами (рис. 37), смысл их совпадает со смыслом работ *IDEF0*, они имеют входы и выходы, но не поддерживают управления и механизмы.

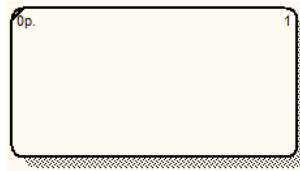


Рис. 37. Графическое изображение работы (процесса)

Внешняя сущность представляет собой материальный объект или физическое лицо, являющееся источником или приемником информации. Например, заказчики, персонал, поставщики, клиенты, склад. Внешние сущности изображаются в виде прямоугольника с тенью (рис. 38) и обычно располагаются по краям диаграммы. Одна внешняя сущность может быть использована многократно на одной или нескольких диаграммах.

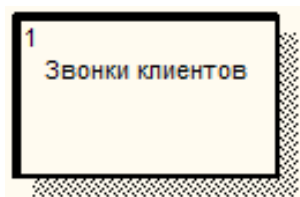


Рис. 38. Графическое изображение внешней сущности

Хранилище (накопитель) данных моделирует данные, которые будут сохраняться в памяти между процессами. Информация, которую содержит хранилище, может использоваться в любое время после ее получения, при этом данные могут выбираться в любом порядке. Имя хранилища должно определять его содержимое и быть существительным (рис. 39). Накопитель данных в общем случае является прообразом будущей базы данных, и описание хранящихся в нем данных должно быть связано с информационной моделью.

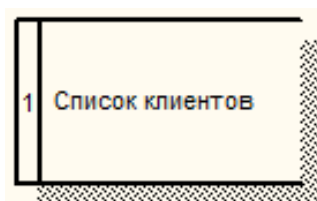


Рис. 39. Графическое изображение хранилища данных


Поток данных определяет информацию, передаваемую через некоторое соединение от источника к приемнику. Поток данных на диаграммах изображаются именованными стрелками, ориентация которых указывает направление движения информации. В *DFD* также используются двунаправленные стрелки для описания диалогов типа «команда-ответ» между работами, между работой и внешней сущностью и между внешними сущностями.

В отличие от стрелок *IDEF0*, которые представляют собой жесткие взаимосвязи, стрелки *DFD* показывают, как объекты (включая данные) двигаются от одной работы к другой. Это представление потоков совместно с хранилищами данных и внешними сущностями делает модели *DFD* более похожими на физические характеристики системы – движение объектов, хранение объектов, поставка и распространение объектов.

Источники информации (внешние сущности) порождают информационные потоки (потоки данных), переносящие информацию к подсистемам или процессам. Те, в свою очередь, преобразуют информацию и порождают новые потоки, которые переносят информацию к другим процессам или подсистемам, накопителям данных или внешним сущностям – потребителям информации.

Упражнение 1

1. Запустите *CA ERwin Process Modeler* и откройте модель *Учет ТМЦ*, созданную в предыдущей работе.

2. На диаграмме декомпозиции *A0* выделите работу *Складской учет* и на панели инструментов модели нажмите кнопку *Go to Child Diagram* (Создать дочернюю диаграмму) . В появившемся диалоговом окне *Activity Box Count* (Число работ) установите тип методологии *DFD*, количество работ 2 и нажмите *OK*. Будет создана диаграмма декомпозиции, и будет отображена панель инструментов модели *DFD* (рис. 40).

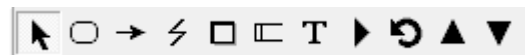




Рис. 40. Панель инструментов модели *DFD*

3. На новой диаграмме задайте имена работам: *Количественный учет*; *Стоимостной учет*.

4. Удалите имеющиеся граничные стрелки с диаграммы *DFD A1*.

5. Используя кнопку *External Reference Tool* (Внешняя сущность)  на панели инструментов модели, создайте внешнюю сущность *Материально ответственное лицо* (рис. 41).

6. Используя кнопку *Data Store Tool* (Хранилище данных)  на панели инструментов модели, создайте хранилища данных (рис. 41): *Ведомость учета остатков*; *Записи в БД*; *Первичные документы*.

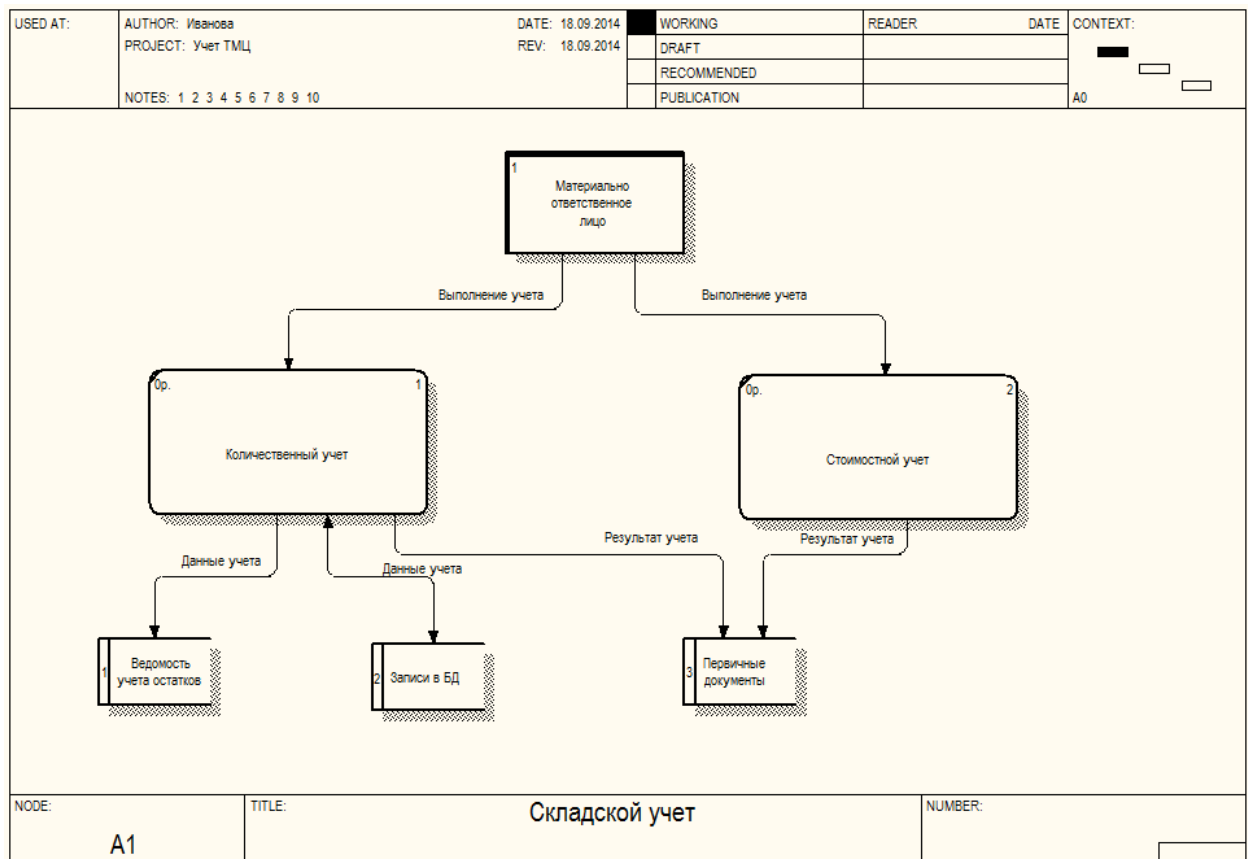


Рис. 41. DFD-диаграмма A1 Складской учет

7. Создайте внутренние стрелки согласно табл. 12 и рис. 41. Обратите внимание, что стрелка *Данные учета*, идущая к хранилищу *Записи в БД*, двунаправленная. Чтобы сделать стрелку двунаправленной, щелкните правой кнопкой по стрелке, выберите в контекстном меню команду *Style (Стиль)* и в диалоговом окне *Arrow Properties (Свойства стрелки)* установите опцию *Bidirectional (Двунаправленная)*.

Таблица 12

Потоки данных DFD-диаграммы A1 Складской учет

Имя стрелки	Начало стрелки	Конец стрелки
Выполнение учета	Материально ответст- венное лицо	Количественный учет
Выполнение учета	Материально ответст- венное лицо	Стоимостной учет
Данные учета	Количественный учет	Ведомость учета остатков
Данные учета	Количественный учет	Записи в БД
Результат учета	Количественный учет	Первичные документы
Результат учета	Стоимостной учет	Первичные документы

Упражнение 2.

В соответствии с табл. 12, 13 и рис. 42 проведите декомпозицию работы *Инвентаризация*.

Таблица 13

Объекты *DFD*-диаграммы *А3 Инвентаризация*

Вид объекта	Имя объекта
Работа	Выявление наличия продукции
Работа	Сопоставление с данными бухгалтерского учета
Работа	Формирование БД персонала
Внешняя сущность	Инвентаризационная комиссия
Хранилище данных	Записи в БД
Хранилище данных	Бухгалтерская отчетность
Хранилище данных	БД сотрудников

Таблица 14

Потоки данных *DFD*-диаграммы *А3 Инвентаризация*

Имя стрелки	Начало стрелки	Конец стрелки
1	2	3
Выполнение инвентаризации	Инвентаризационная комиссия	Выявление наличия продукции
Выполнение инвентаризации	Инвентаризационная комиссия	Сопоставление с данными бухгалтерского учета
Внесение записи	Инвентаризационная комиссия	Формирование БД персонала
Информация о продукции	Записи в БД	Выявление наличия продукции
Данные отчетности	Бухгалтерская отчетность	Сопоставление с данными бухгалтерского учета
Выбор записи	БД сотрудников	Формирование БД персонала

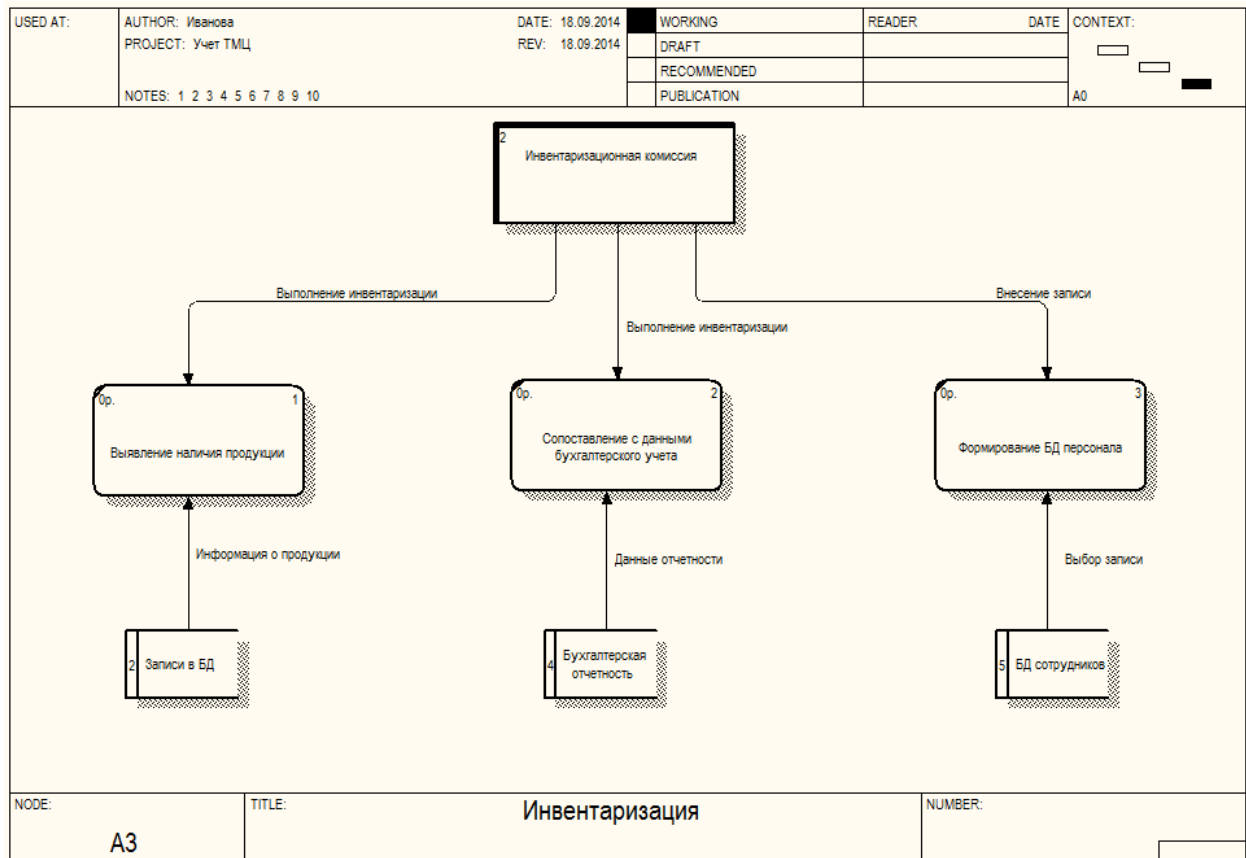


Рис. 42. DFD-диаграмма A3 Инвентаризация

Упражнение 3

1. В соответствии с табл. 15, 16 и рис. 43 проведите декомпозицию работы *Количественный учет*.
2. На основе табл. 17, 18 и рис. 44 проведите декомпозицию работы *Стоимостной учет*.

Таблица 15

Объекты DFD-диаграммы A11 Количественный учет

Вид объекта	Имя объекта
Работа	Регистрация в БД поступления нового товара
Работа	Регистрация в БД выбытия товара
Работа	Формирование БД клиентов
Внешняя сущность	Материально ответственное лицо, бухгалтер
Хранилище данных	БД готовой продукции
Хранилище данных	БД контрагентов

Таблица 16

Потоки данных *DFD*-диаграммы *A11 Количественный учет*

Имя стрелки	Начало стрелки	Конец стрелки
Накладная на передачу готовой продукции в места хранения	Материально ответственное лицо, бухгалтер	Регистрация в БД поступления нового товара
Чек/накладная	Материально ответственное лицо, бухгалтер	Регистрация в БД выбытия товара
Внесение записи	Материально ответственное лицо, бухгалтер	Формирование БД клиентов
Занесение наименования, количества и т.д.	Регистрация в БД поступления нового товара	БД готовой продукции
Учет выбытия товара (двунаправленная стрелка)	Регистрация в БД выбытия товара	БД готовой продукции
Выбор записи	БД контрагентов	Регистрация в БД выбытия товара
Обновление данных (двунаправленная стрелка)	Формирование БД клиентов	БД контрагентов

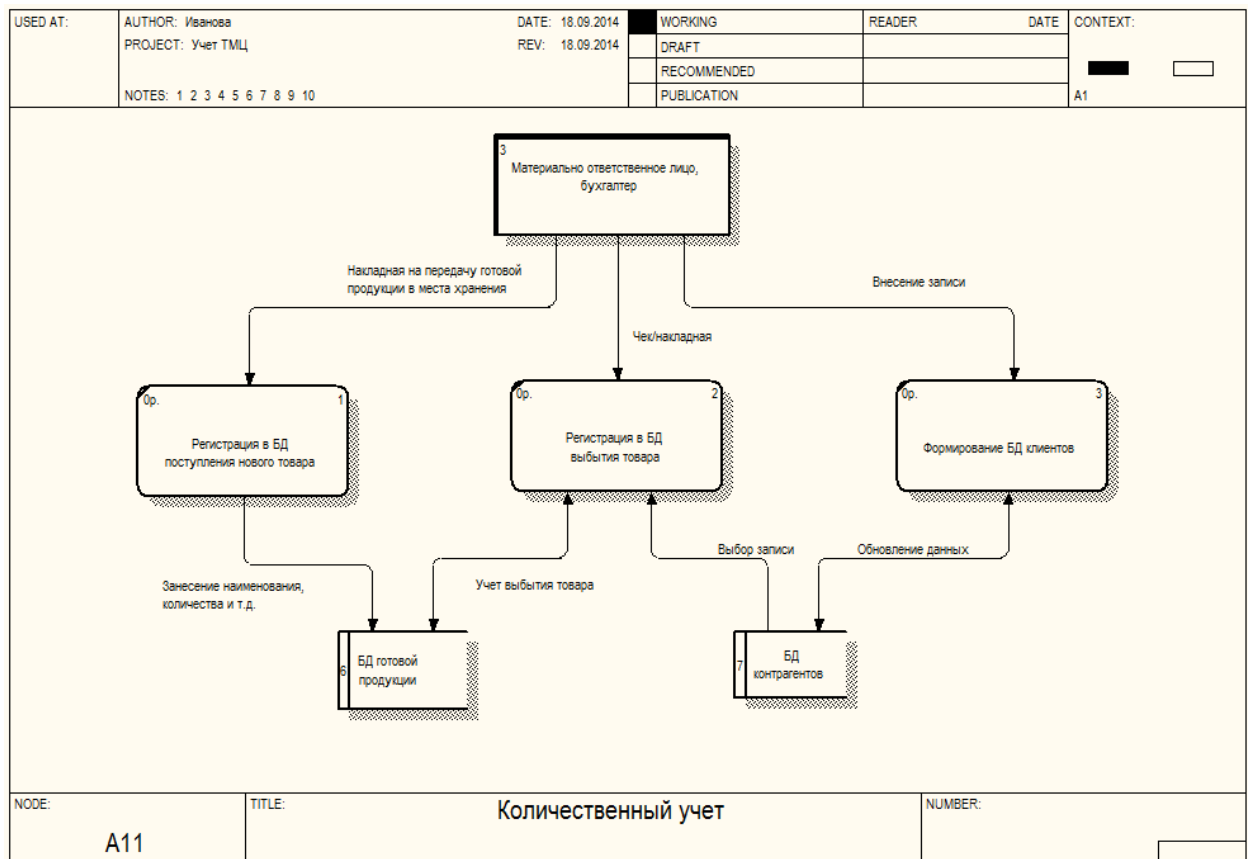


Рис. 43. DFD-диаграмма A11 Количественный учет

Таблица 17

Объекты DFD-диаграммы A12 Стоимостной учет

Вид объекта	Имя объекта
Работа	Регистрация в БД_ поступления нового товара
Работа	Регистрация в БД_ выбытия товара
Работа	Формирование БД_ клиентов
Внешняя сущность	Материально ответственное лицо, бухгалтер
Хранилище данных	БД готовой продукции
Хранилище данных	БД контрагентов

Таблица 18

Потоки данных DFD-диаграммы A12 Стоимостной учет

Имя стрелки	Начало стрелки	Конец стрелки
1	2	3
Накладная на передачу готовой продукции в места хранения	Материально ответственное лицо, бухгалтер	Регистрация в БД_ поступления нового товара

Окончание таблицы 18

1	2	3
Чек/накладная	Материально ответст- венное лицо, бухгалтер	Регистрация в БД_ вы- бытия товара
Внесение записи	Материально ответст- венное лицо, бухгалтер	Формирование БД_ кли- ентов
Занесение наименова- ния, стоимости и т.д.	Регистрация в БД_ по- ступления нового товара	БД готовой продукции
Учет выбытия товара	Регистрация в БД_ вы- бытия товара	БД готовой продукции
Выбор записи	БД контрагентов	Регистрация в БД_ вы- бытия товара
Обновление данных (двунаправленная стрелка)	Формирование БД_ кли- ентов	БД контрагентов

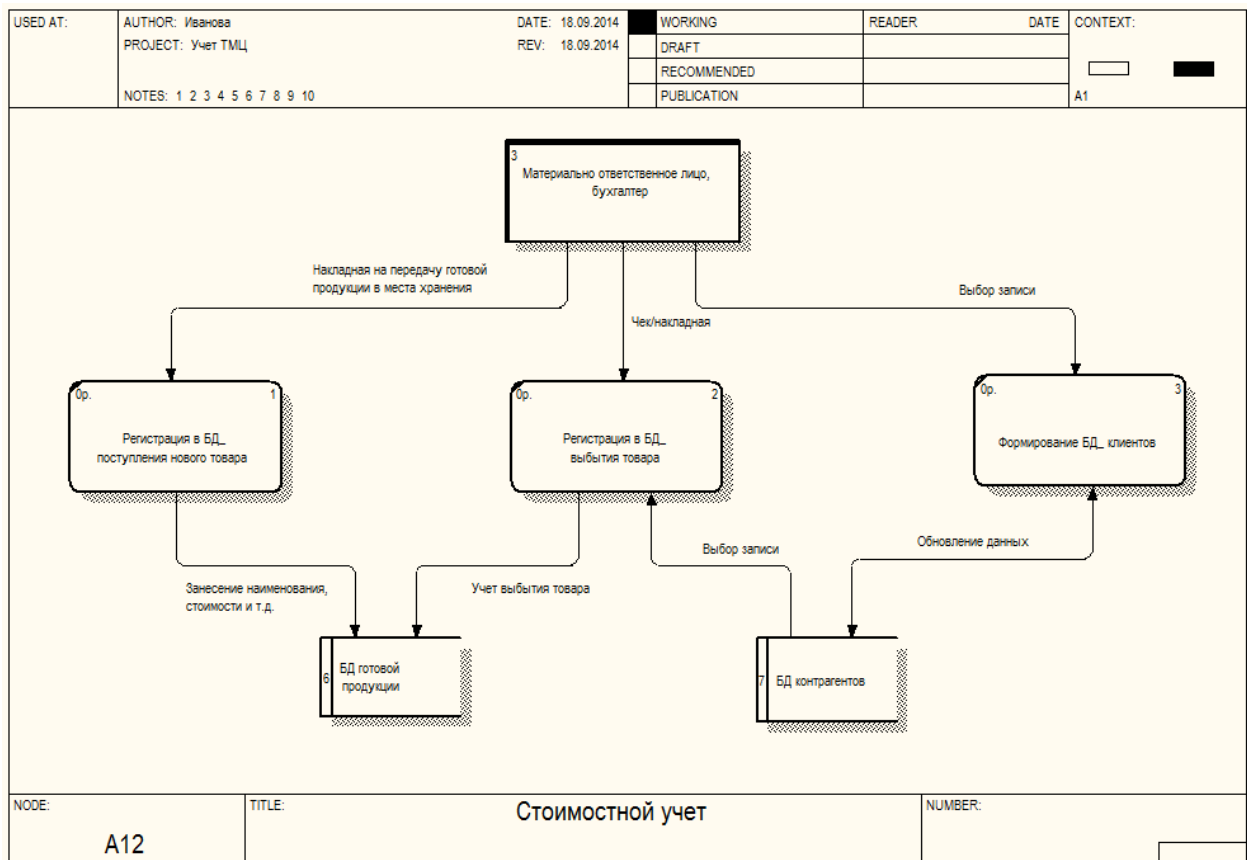


Рис. 44. DFD-диаграмма A12 Стоимостной учет

Вопросы для самопроверки

1. Для чего предназначены диаграммы *DFD*? Чем они отличаются от диаграмм *IDEF0*?
2. Какие основные компоненты составляют диаграммы потоков данных?
3. Как графически представляются компоненты диаграммы *DFD*?

ПРАКТИЧЕСКАЯ РАБОТА № 5

Тема работы: Создание моделей в стандарте *IDEF3* (метод описания процессов – *workflow diagramming*).

Методология построения моделей *IDEF3* используется для графического описания информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов. Диаграммы *IDEF3* могут применяться в моделировании бизнес-процессов для анализа завершенности процедур обработки информации. С их помощью можно описывать сценарии действий сотрудников организации (например, последовательность обработки заказа или события), которые необходимо обработать за конечное время. Каждый сценарий сопровождается описанием процесса и может быть использован для документирования каждой функции.

В отличие от других методик описаний процессов *IDEF3* не ограничивает аналитика жесткими рамками синтаксиса, что может привести к созданию неполных или противоречивых моделей.

Центральными компонентами *IDEF3*-диаграмм являются *работы*, которые изображаются прямоугольниками (рис. 45) и имеют имя, выраженное отглагольным существительным, обозначающим процесс действия (например, *Изготовление детали*), и номер (идентификатор). Идентификатор работы присваивается при создании работы и никогда не меняется.

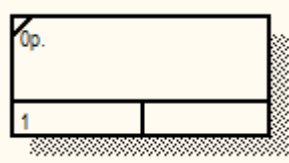


Рис. 45. Графическое изображение работы

Связи (стрелки) показывают взаимоотношения работ и могут быть направлены в любую сторону, но обычно диаграммы *IDEF3* стараются построить


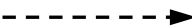

так, чтобы связи были направлены слева направо. В *IDEF3* различают три типа стрелок (табл. 19).

Старшая связь показывает, что работа-источник заканчивается ранее, чем начинается работа-цель. Отношение показывает, что стрелка является альтернативой старшей стрелке или потоку объектов в последовательности выполнения работ – работа-источник не обязательно должна закончиться, прежде чем работа-цель начнется. Работа-цель может закончиться прежде, чем закончится работа-источник.

Окончание одной работы может служить сигналом к началу нескольких работ, или же одна работа для своего запуска может ожидать окончание нескольких работ.

Таблица 19

Типы связей на диаграмме *IDEF3*

Название связи	Изображение связи	Определение
Старшая связь		Показывает, что работа-источник должна заканчиваться прежде, чем работа-цель начнется
Отношения		Используется для изображения связей между работами и между работами и объектами ссылок
Потоки объектов		Применяется для описания факта, что объект используется в двух или более работ

Для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы, используются *перекрестки*. Различают перекрестки для слияния (*Fan-in Junction*) и разветвления (*Fan-out Junction*) стрелок. Перекресток не может использоваться одновременно для слияния и для разветвления. Смысл каждого типа перекрестков приведен в табл. 20.

Типы перекрестков

Соединение	Имя	Значение <i>Fan-in Junction</i>	Значение <i>Fan-out Junction</i>
1	2	3	4
	Асинхронное И (AND)	Все предшествующие процессы должны быть завершены	Все следующие процессы должны быть запущены
	Синхронное И (AND)	Все предшествующие процессы завершены одновременно	Все следующие процессы запускаются одновременно
	Асинхронное ИЛИ (OR)	Один или несколько предшествующих процессов должны быть завершены	Один или несколько последующих процессов должны быть запущены
	Синхронное ИЛИ (OR)	Один или более предшествующих процессов завершаются одновременно	Один или более последующих процессов запускаются одновременно
	Исключающее ИЛИ (XOR)	Только один предшествующий процесс завершен	Только один следующий процесс запускается

Объект ссылки в *IDEF3* выражает некоторую идею, концепцию или данные, которые нельзя связать со стрелкой, перекрестком или работой. Объект ссылки изображается в виде прямоугольника (рис. 46), в качестве имени можно использовать имя какой-либо стрелки с других диаграмм или имя сущности из модели данных. Объекты ссылки должны быть связаны с работами или перекрестками пунктирными линиями. При внесении объектов ссылок кроме имени следует указывать тип объекта ссылки. Типы объектов ссылок приведены в табл. 21.

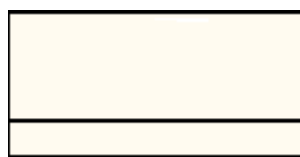


Рис. 46. Графическое изображение объекта ссылки

Типы объектов ссылок

Тип объекта ссылки	Цель описания
<i>OBJECT</i>	Описывает участие важного объекта в работе
<i>GOTO</i>	Инструмент циклического перехода (в повторяющейся последовательности работ).
<i>UOB (Unit of behavior)</i>	Применяется, когда необходимо подчеркнуть множественное использование какой-либо работы, но без цикла.
<i>NOTE</i>	Используется для документирования важной информации, относящейся к каким-либо графическим объектам на диаграмме.
<i>ELAB (Elaboration)</i>	Используется для усовершенствования графиков или их более детального описания. Обычно употребляется для детального описания разветвления и слияния стрелок на перекрестках

Декомпозиция в *IDEF3* используется для детализации работ. Методология *IDEF3* позволяет декомпонировать работу многократно, т.е. работа может иметь множество дочерних работ. Это позволяет в одной модели описать альтернативные потоки.

Упражнение 1

1. В соответствии с табл. 22, 23 и рис. 47 проведите декомпозицию работы *Бухгалтерский учет* по методологии *IDEF0*.

Таблица 22

Работы диаграммы декомпозиции *A2 Бухгалтерский учет*

<i>Name (Имя)</i>	<i>Definition (Определение)</i>
Аналитический учет	Аналитический учет ведется в количественных и стоимостных показателях по видам изделий, по сортам изделий и по ценам
Синтетический учет	Синтетический учет ведется на счете 40, который включает в себя учет фактической себестоимости и оприходование по учетным ценам

Таблица 23

Стрелки диаграммы декомпозиции *A2 Бухгалтерский учет*

<i>Arrow Name</i> (Имя стрелки)	Тип стрелки	Начало стрелки	Конец стрелки
Первичные документы	Вход		Аналитический учет
Инвентаризационная опись или акт	Вход		Аналитический учет
Контроль за движением готовой продукции	Управление		Аналитический учет Синтетический учет
Оборотная ведомость	Внутренняя	Аналитический учет	Синтетический учет
Ведомость товарного выпуска	Вход		Синтетический учет
Отчет о реализации	Вход		Синтетический учет
Персонал предприятия	Механизм		Аналитический учет Синтетический учет
Составление отчетности	Выход	Синтетический учет	
Данные отчетности	Выход	Синтетический учет	
Ведомость учета остатков	Выход	Синтетический учет	

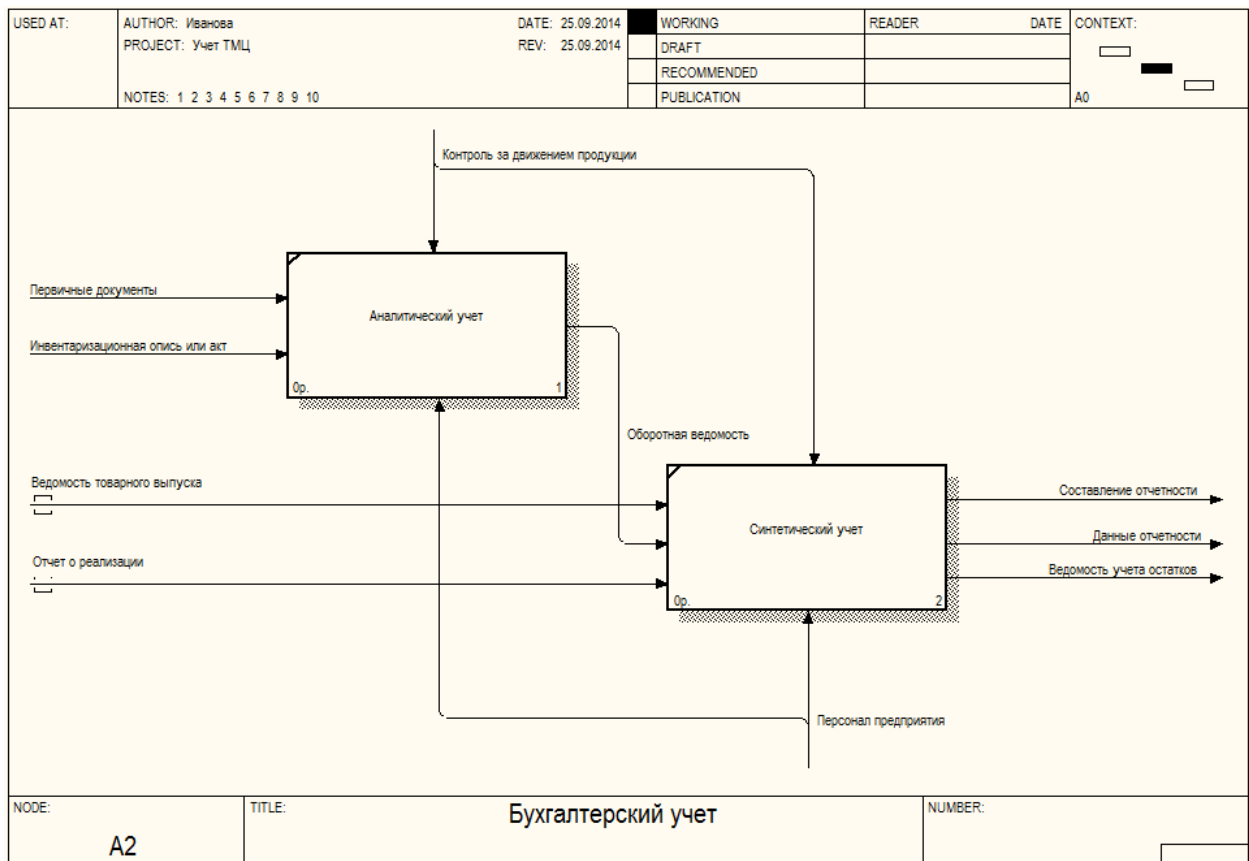


Рис. 47. Декомпозиция A2 Бухгалтерский учет

2. На диаграмме декомпозиции A2 выделите работу *Аналитический учет* и на панели инструментов модели нажмите кнопку *Go to Child Diagram* (Создать дочернюю диаграмму) . В появившемся диалоговом окне *Activity Box Count* (Число работ) установите тип методологии *IDEF3*, количество работ 5 и нажмите *OK*. Будет создана диаграмма декомпозиции, и отобразится панель инструментов модели *IDEF3* (рис. 48).

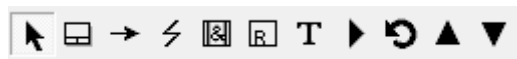



Рис. 48. Панель инструментов модели IDEF3

3. На новой диаграмме задайте имена работам:

- 1 – Учет в количественных и стоимостных показателях
- 2 – Учет по видам изделий
- 3 – Учет по сортам изделий
- 4 – Учет по ценам
- 5 – Обобщение данных учета в оборотных ведомостях

4. С помощью кнопки *Junction Tool (Перекресток)*  на панели инструментов модели создайте два перекрестка типа *Асинхронное И (AND)* и свяжите работы с перекрестками, как показано на рис. 49.

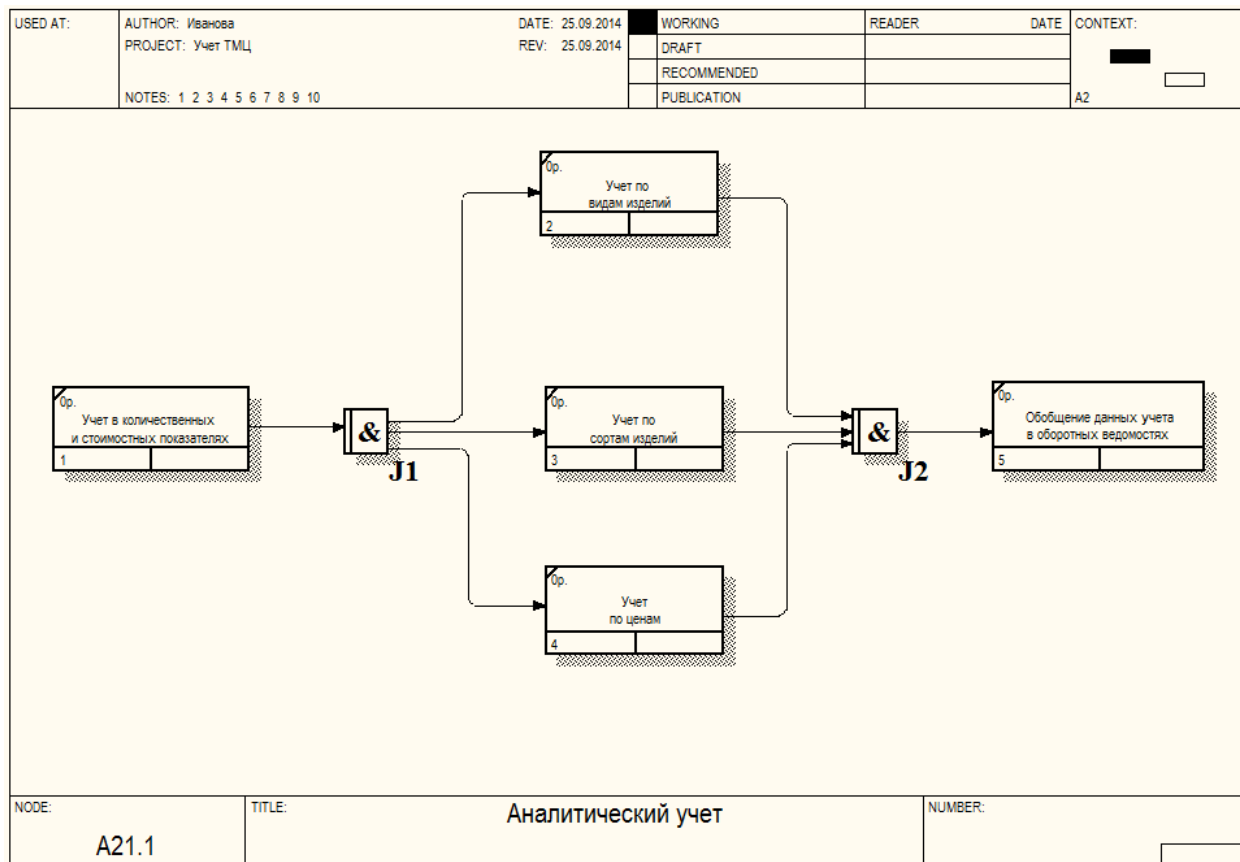


Рис. 49. Декомпозиция A21.1 Аналитический учет

Упражнение 2

Создайте *IDEF3*-диаграмму для работы *Синтетический учет*, которая содержит 4 работы:

6 – Учет продукции на счете 40

7 – Учет по фактической себестоимости

8 – Оприходование по учетным ценам

9 – Вычисление фактической себестоимости

и два перекрестка типа *Асинхронное И (AND)* (рис. 50.)

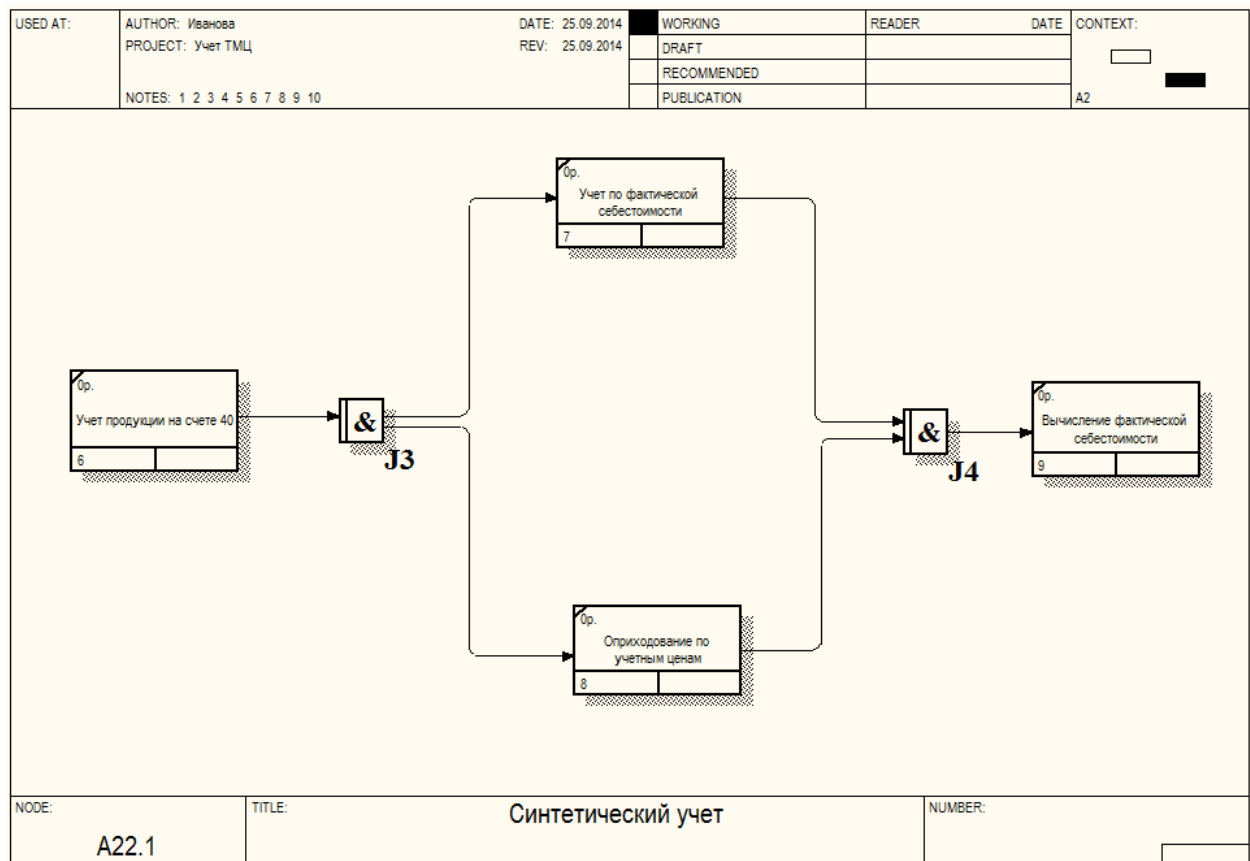


Рис. 50. Декомпозиция A22.1 Синтетический учет

Упражнение 3

Создайте *IDEF3*-диаграмму для работы *Учет продукции на счете 40*, которая содержит 4 работы:

10 – Счет 40

11 – Дебет

12 – Кредит

13 – Закрытие в каждом отчетном месяце

один перекресток типа *Асинхронное И* (AND) и один перекресток *Синхронное И* (AND) (рис. 51).

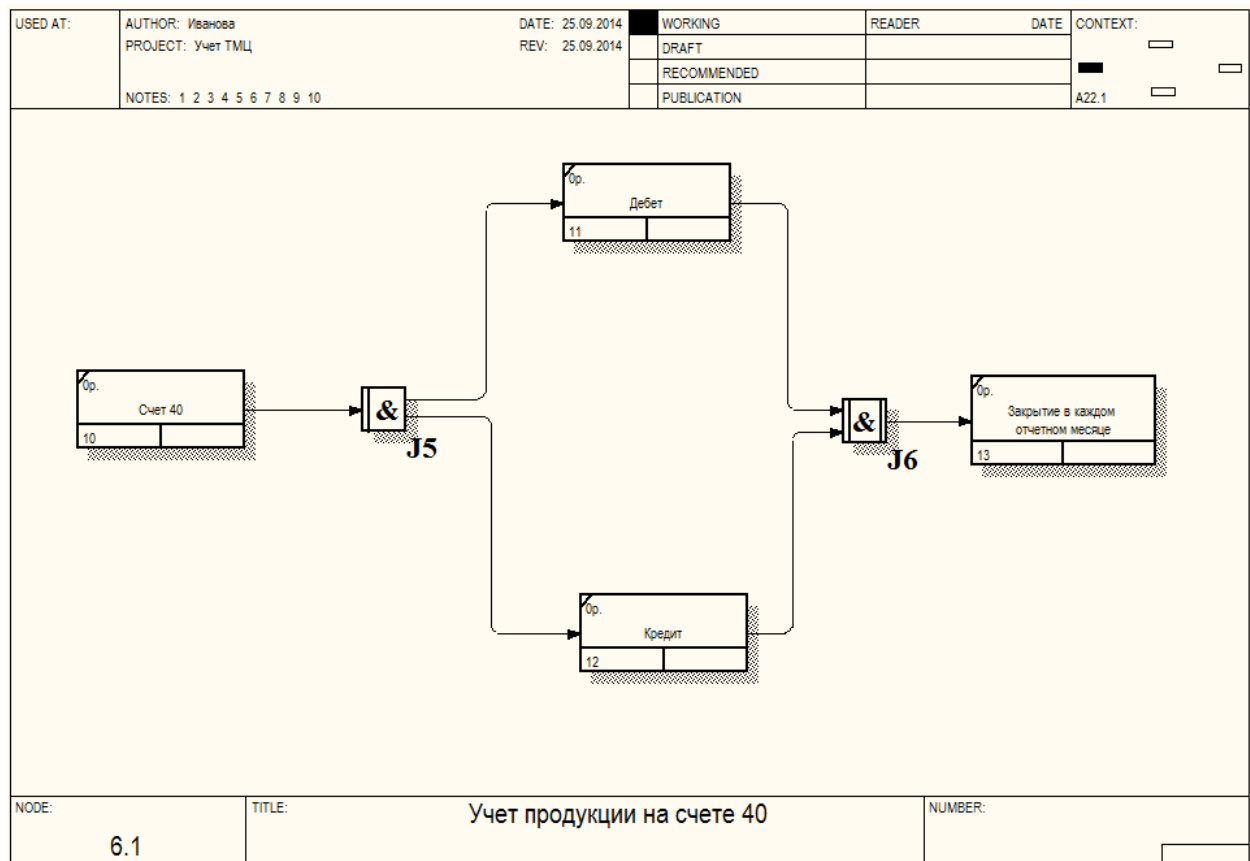


Рис. 51. Декомпозиция 6.1 Учет продукции на счете 40

Упражнение 4

Создайте *IDEF3*-диаграмму для работы *Учет фактической себестоимости*, которая содержит 8 работ:

- 14 – Зарплата персоналу
 - 15 – Отчисления в разные фонды и бюджеты
 - 16 – Затраты на содержание, обслуживание, управление организацией
 - 17 – Затраты, связанные с производством и реализацией продукции
 - 18 – Затраты на подготовку и освоение новой продукции
 - 19 – Платежи сторонним организациям за оказание услуг по производству
 - 20 – Прочие затраты
 - 21 – Формирование фактической себестоимости
- и один перекресток типа *Синхронное И (AND)* (рис. 52).

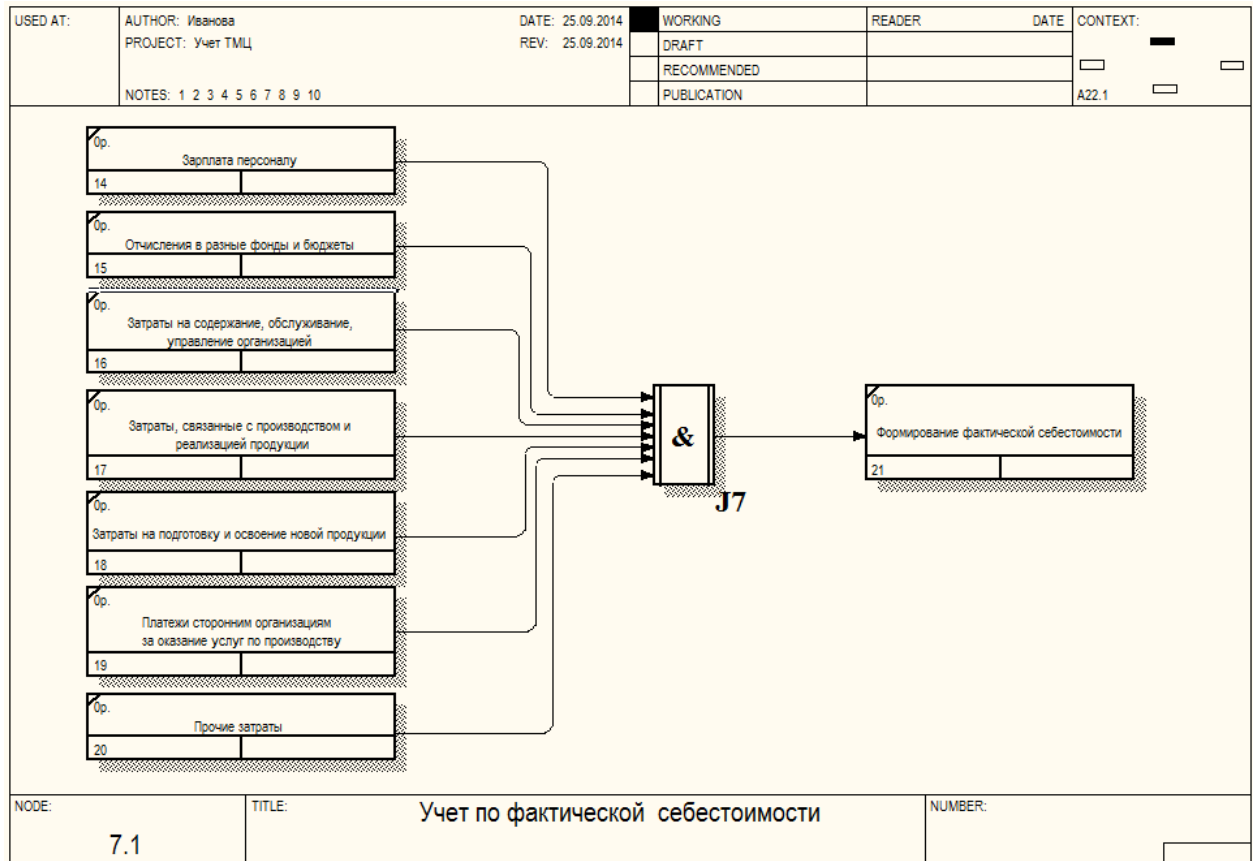


Рис. 52. Декомпозиция 7.1 Учет по фактической себестоимости

Вопросы для самопроверки

1. В чем состоит назначение метода описания процессов *IDEF3*?
2. Какие функциональные элементы может содержать диаграмма *IDEF3*?
3. Что такое работа на диаграмме *IDEF3*? Как она изображается?
4. Какие типы связей между работами существуют?
5. Для чего предназначены перекрестки на диаграмме? Какова их классификация?
6. Что такое объект ссылки? Как он отображается?

ПРАКТИЧЕСКАЯ РАБОТА № 6

Тема работы: Методология *ARIS*. Создание организационной и функциональной моделей компании в *График-студии Лайт*.

Одной из современных методологий бизнес-моделирования, получившей широкое распространение в России, является методология *ARIS*, которая расшифровывается как *Architecture of Integrated Information System* – проектирование интегрированных информационных систем.

Методика моделирования *ARIS* основывается на разработанной профессором Августом Вильгельмом Шеером теории построения интегрированных информационных систем (ИС), определяющей принципы визуального отображения всех аспектов функционирования анализируемой организации (рис. 53). *ARIS* поддерживает четыре типа моделей, отражающих различные аспекты исследуемой системы:

- *организационная модель* представляет структуру системы – иерархию организационных подразделений, должностей и конкретных лиц, связи между ними, а также территориальную привязку структурных подразделений;
- *функциональная модель* содержит стратегические цели компании, функции, которые должны быть выполнены, и их взаимосвязи. Эта модель включает описание самих функций, а также перечень отдельных подфункций (операций) и их связей как с основной функцией, так и между собой;
- *модель ресурсов* отражает используемые ресурсы (оборудование и программы). Эта модель анализируемого бизнес-процесса необходима для описания компонент, связанных с самим бизнес-процессом. Компоненты других типов моделей рассматриваются с точки зрения их привязанности к ресурсам;
- *управленческая модель* описывает взаимосвязи между моделями различных типов. Интеграция этих взаимосвязей с помощью модели специального типа позволяет вводить дополнительные взаимосвязи без какой-либо избыточности.

ARIS не накладывает ограничений на последовательность разработки четырех видов моделей. Процесс анализа и проектирования можно начинать с любого из них, в зависимости от конкретных условий и целей, предусмотренных разработчиками.

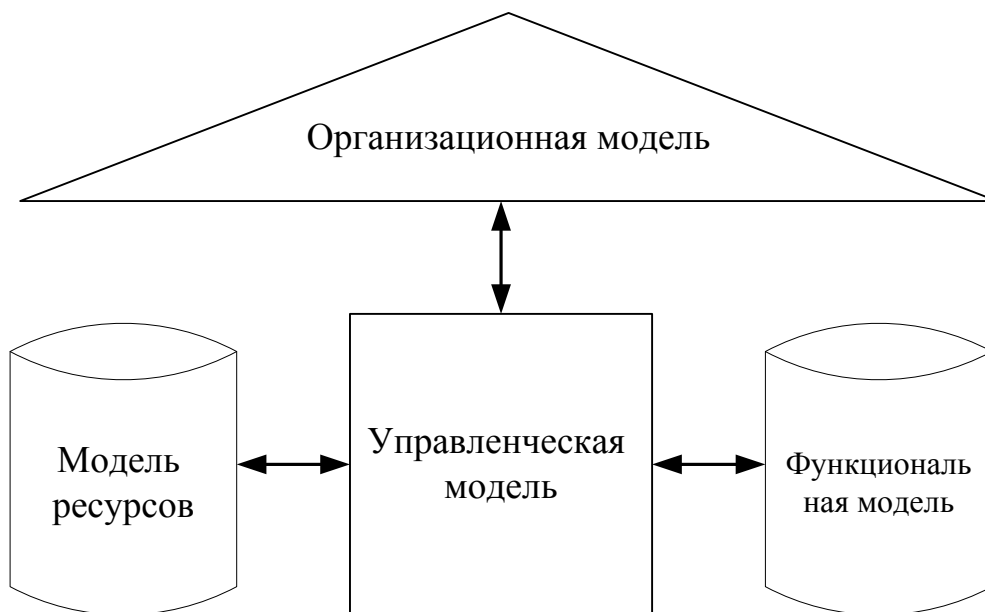


Рис. 53. Взаимосвязь типов моделей, используемых ARIS

Другой особенностью методологии ARIS является использование различных уровней описания, т.е. каждая из моделей делится еще на три подуровня (рис. 54):

- *уровень описания требований* – разрабатываются модели, описывающие то, что должна делать система – как она организована, какие деловые процессы в ней присутствуют, какие данные при этом используются;
- *уровень описания спецификаций проекта* – описываются уже не функции, а пользовательские или модульные итерации, которые выполняют функции. Рассматривается отображение сформулированных требований в категории и методы описания, связанные непосредственно с ИС и выраженные в терминах информационных технологий;
- *уровень описания реализации* – происходит преобразование спецификации в физическое описание конкретных программных и технических средств. Это заключительный этап проектирования систем, за которым следует этап физической реализации (программирования). Уровень описания реализации порождает документы, на основе которых обеспечивается процесс разработки программных модулей (или подбора готовых программных компонент, отвечающих поставленным требованиям), а также выбор и организация технических средств реализации системы.

Создание различных типов моделей и проработка каждой из них по уровням в сочетании с формулировкой проблем бизнеса и составляет процесс работы в архитектуре *ARIS*.



Рис. 54. Архитектура *ARIS*

Рассмотрим правила выделения процессов в организации. Для этого определим несколько понятий.

Процесс – это устойчивая, целенаправленная совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входы в выходы, представляющие ценность для потребителя.

Процессы делятся на три основные группы:

- сквозные процессы, проходящие через несколько подразделений организации или через всю организацию, пересекающие границы функциональных подразделений. Сквозные процессы часто называют межфункциональными процессами;

– процессы (подпроцессы) подразделений, деятельность которых ограничена рамками одного функционального подразделения организации. Такие процессы называют внутрифункциональными процессами;

– операции (функции) самого нижнего уровня декомпозиции деятельности организации, которые, как правило, выполняются одним человеком.

Названия процессов, подпроцессов (или функций) должно быть выражено глаголом или отглагольным существительным.

Для управления процессом необходимо назначить должностное лицо, ответственное за выполнение процесса и его результат. Чтобы должностное лицо могло управлять процессом, в его распоряжение должны быть выделены ресурсы, необходимые для проведения процесса, делегированы права и полномочия. Каждый процесс существует не сам по себе, а выполняет какие-либо функции в организации и является подконтрольным высшему руководству организации.

Владелец процесса – это должностное лицо, имеющее в своем распоряжении ресурсы, необходимые для выполнения процесса, и несущее ответственность за результат процесса.

Вход бизнес-процесса – продукт, который в ходе выполнения процесса преобразуется в выход. Например, сырье, материалы, информация, услуги и т.д. Вход всегда должен иметь своего поставщика.

Выход процесса (продукт) – материальный или информационный объект или услуга, являющийся результатом выполнения процесса и потребляемый внешними по отношению к процессу клиентами. Выход процесса всегда имеет потребителя.

Ресурс бизнес-процесса – материальный или информационный объект, постоянно используемый для выполнения процесса, но не являющийся входом процесса. Например, информация, персонал, оборудование, программного обеспечение, транспорт, связь и т.д. Владелец процесса в ходе планирования и управления процессом производит распределение и перераспределение ресурсов для достижения наилучшего результата процесса.

Деление объектов, необходимых для выполнения процесса, на входы и ресурсы, является достаточно условным. Более важным для выполнения процесса, является точное определение того, что должно поступать в распоряжение владельца процесса, чтобы процесс состоялся и был выполнен успешно.

Входы, выходы и ресурсы обозначаются существительным.

Таким образом, процесс включает:

- *владельца процесса* – должностное лицо, имеющее в своем распоряжении ресурсы процесса, с определенными правами, зоной ответственности и полномочиями;
- *технология процесса* – порядок выполнения деятельности по преобразованию входов в выходы;
- *систему показателей процесса* – показателей продукта, показателей эффективности процесса, показателей удовлетворенности потребителей;
- *управление процессом* – деятельность владельца процесса по анализу данных о процессе и принятию управленческих решений;
- *ресурсы процесса* – информация и материальные средства, которые владелец распределяет в ходе планирования работ по процессу и учитывает при расчете эффективности процесса, как соотношение затраченных ресурсов на полученный результат процесса.

Процессы организации могут быть разделены на три основных типа по характеру деятельности и создаваемому продукту (табл. 24).

Таблица 24

Классификация процессов предприятия

Типы процессов	Характерные признаки	Клиенты
Основные процессы (например, закупка, производство, хранение, поставка продукции и т.д.)	1. Назначение процессов – создание основных продуктов 2. Результат – основной продукт или полуфабрикат для его изготовления 3. Процессы лежат на пути создания основных продуктов 4. Процессы добавляют к продукту ценность для потребителя	1. Внешние клиенты 2. Конечные потребители 3. Внутренние клиенты – другие процессы организации

Типы процессов	Характерные признаки	Клиенты
Обеспечивающие процессы (например, подготовка кадров, сервисное обслуживание оборудования, административно-хозяйственное обеспечение, обеспечение безопасности и т.д.)	1. Назначение процессов – обеспечение деятельности основных процессов 2. Результат – ресурсы для основных процессов 3. Деятельность процессов не касается основных продуктов 4. Процессы добавляют продукту стоимость	Внутренние клиенты – другие процессы организации
Процесс управления организацией (например, стратегическое управление, управление финансами, управление персоналом и т.д.)	1. Назначение процесса – управление деятельностью всей организации 2. Результат – деятельность всей организации	1. Собственники (инвесторы) 2. Потребители (клиенты) 3. Персонал (сотрудники) 4. Поставщики и субподрядчики 5. Общество (внешняя среда)

График-студิโอ Лайт

Бизнес-инженер – профессиональное инструментальное средство бизнес-моделирования, управления эффективностью организации и разработки регламентирующих документов. *Бизнес-инженер* позволяет разрабатывать взаимосвязанные модели стратегических целей, ключевых показателей, бизнес-процессов, организационной структуры и других элементов бизнес-архитектуры организации. В состав *Бизнес-инженера* входит приложение *График-студิโอ Лайт*.

График-студิโอ Лайт – это программный продукт бизнес-моделирования, который позволяет разрабатывать графические диаграммы стратегических це-

лей, бизнес-процессов, организационной структуры и других аспектов деятельности организации.

График-студิโอ Лайт прост в использовании, и поэтому с ним могут работать не только специалисты по бизнес-моделированию, но и другие сотрудники компании, которым поставлена задача описать процессы своих структурных подразделений.

В *График-студิโอ Лайт* включено несколько десятков графических диаграмм, соответствующих наиболее часто применяемым на практике методологиям и нотациям процессного описания: *IDEF0*, *IDEF3*, диаграмма процесса (*DFD*, *WFD* и *Swimmer lanes*), *BPMN* (*Business Process Model and Notation*), *ARIS EPC*, *ARIS Value-added chain diagram*, *ARIS Function tree*, *ARIS Material flow diagram* и другие, а также диаграммы, которые используются для моделирования стратегических целей, показателей – *KPI*, системы *BSC*, организационной структуры, продуктов и услуг, ИТ-системы, бизнес-анализа и моделирования других элементов бизнес-архитектуры организации.

Для разработки графических диаграмм *График-студิโอ Лайт* использует функциональные возможности мощного и гибкого редактора деловой графики *Microsoft Visio*.

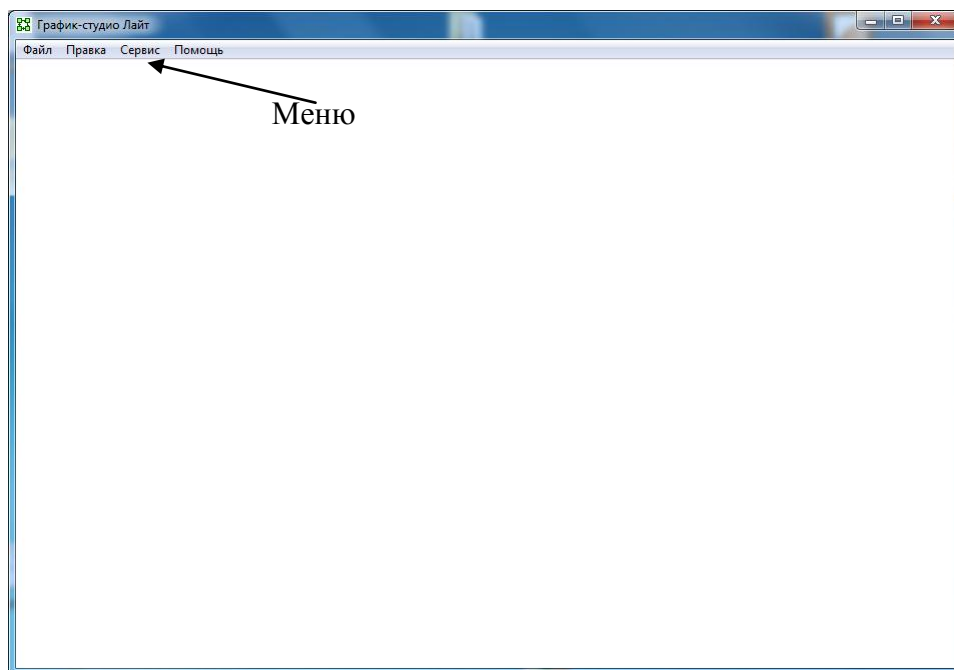
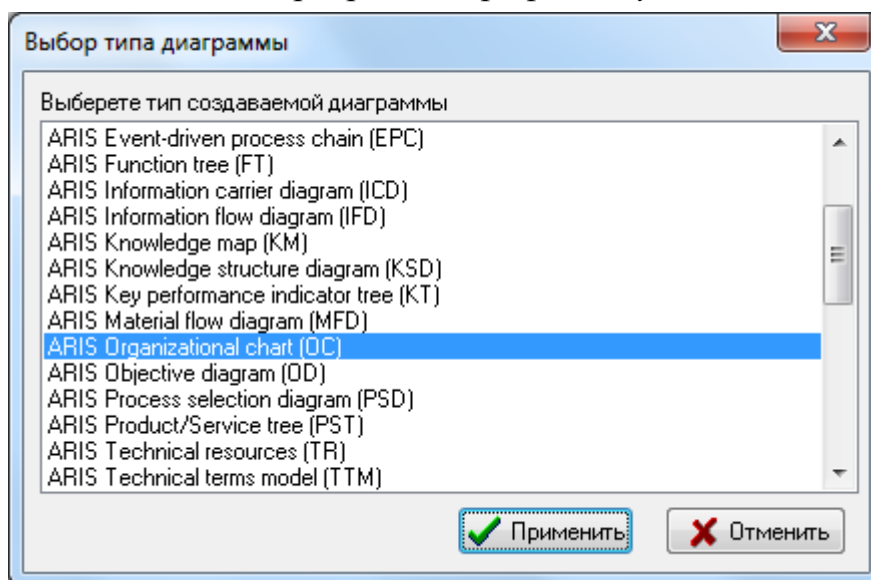
Задание.

При выполнении упражнений будет рассматриваться деятельность издательской компании, которая занимается изданием газет, журналов и рекламной продукции.

Упражнение 1

1. Запустите программу *График-студิโอ Лайт* 8. На экране появится окно программы (рис. 55), в котором в меню вызовите команду *Файл* ⇒ *Новая диаграмма*.

2. В появившемся окне *Выбор типа диаграммы* (рис. 56) выберите *ARIS Organizational chart* (*OC*) и нажмите на кнопку *Применить*.

Рис. 55. Окно программы *График-студия Лайт 8*Рис. 56. Окно *Выбор типа диаграммы*

Окно программы *График-студия Лайт 8* изменится (рис. 57). В центральной части окна будет располагаться страница диаграммы, в которой осуществляется ее построение. Вверху появится панель инструментов *Редактирование*, которая отображается при создании любого типа диаграммы и используется для редактирования страницы и элементов диаграммы. В правой части окна находится панель *Моделирование*, содержащая элементы, которые используются для построения выбранного типа диаграммы. При наведении указателя мыши на элементе будет появляться его название.

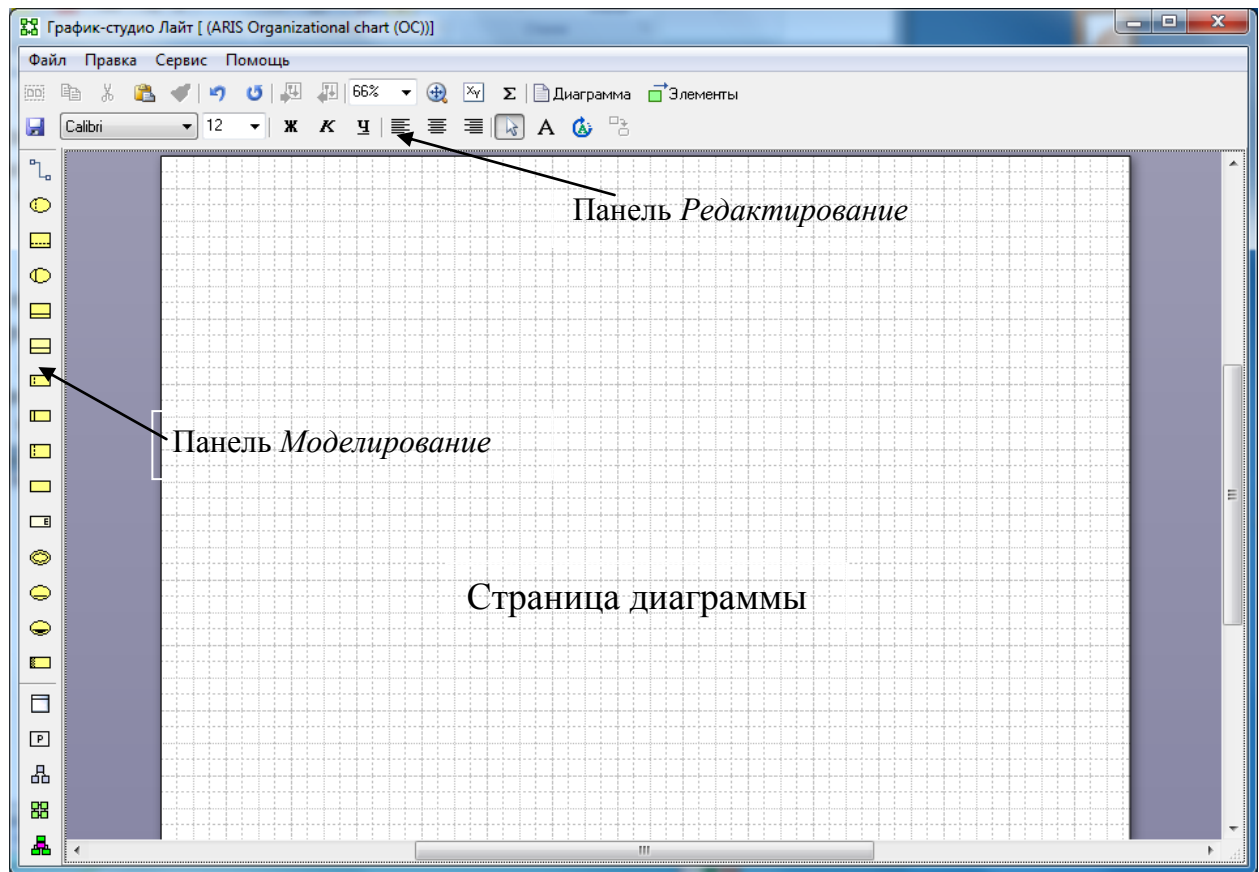
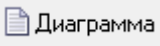
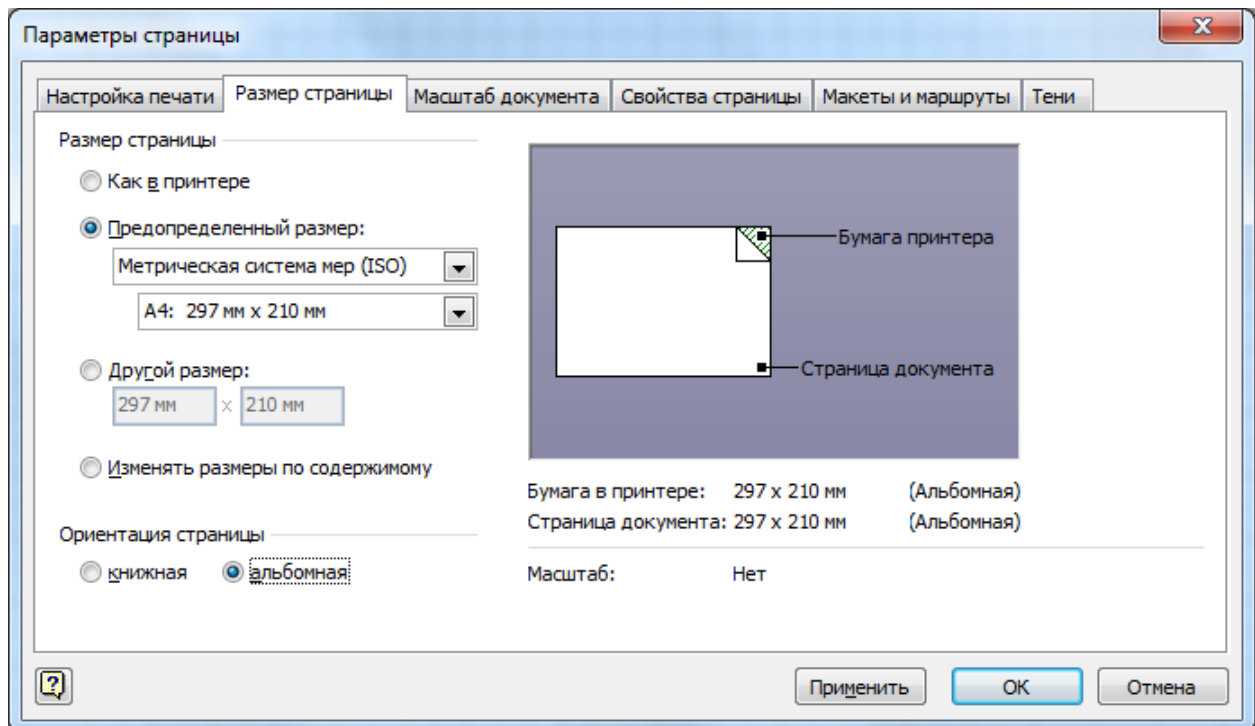



Рис. 57. Окно программы *График-студิโอ Лайт* [ARIS Organizational chart (OC)]

3. Для установки параметров страницы на панели *Редактирование* нажмите на кнопку *Диаграмма* , а затем выберите команду *Параметры страницы*. На экране появится соответствующее окно (рис. 58), в котором на вкладке *Размер страницы* установите альбомную ориентацию страницы, размер 210 мм × 297 мм и нажмите *ОК*.

4. Для сохранения диаграммы в меню вызовите команду *Файл* ⇒ *Сохранить диаграмму*. В появившемся окне *Сохранить* как в поле *Имя файла* задайте имя и нажмите *Сохранить*.

Рис. 58. Окно *Параметры страницы*

Построение диаграмм

Построение диаграмм происходит путем размещения элементов с панели *Моделирование* на страницу диаграммы. Для того чтобы поместить элемент необходимо нажать на соответствующий значок на панели *Моделирование*, а затем на странице диаграммы щелкнуть левой кнопкой мыши в том месте, в котором должен располагаться элемент. При наведении курсора мыши на элемент будет появляться подсказка в виде имени. Фигуры на панели *Моделирование* разделены с помощью знака  на две части, выше которого расположены элементы, свойственные только для заданного типа диаграммы, а ниже – общие для всех.







Рассмотрим основные виды диаграмм.

Модель *ARIS Organizational chart (OC)* используется для описания статических отношений между различными структурными элементами – участниками бизнес-процесса, ответственными за выполнение функций на предприятии. На данной модели изображаются структурные подразделения, группы, должности, роли и другие элементы организационной структуры и связи между ними. Низшим уровнем является описание подразделений на уровне должности – штатных единиц, занимаемых конкретными сотрудниками.


При построении диаграммы типа *ARIS Organizational chart (OC)* наиболее часто используются фигуры, приведенные в табл. 25.

Таблица 25

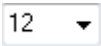
Фигуры диаграммы *ARIS Organizational chart (OC)*


Изображение фигуры	Название	Описание элемента
	Должность	Должность в организации
	Организационная единица (вид 1)	Отдельное штатное подразделение
	Организационная единица (вид 2)	Отдельное штатное подразделение
	Внутренняя персона	Конкретный сотрудник из штата предприятия
	Внешняя персона	Конкретное лицо, не состоящее в штате предприятия
	Установить связь	


Упражнение 2

1. На панели инструментов *Моделирование* нажмите на кнопку *Титульная рамка* , а затем щелкните на странице диаграммы. По границе листа появится рамка, а в верхней части – на красном фоне строка заголовка *Титульная рамка*.

2. Для редактирования текста заголовка дважды щелкните мышью по нему, удалите текст и введите *Организационная структура издательской компании верхнего уровня*.

3. Щелкните мышью по строке заголовка и на панели *Редактирование* в поле *Выбрать размер шрифта*  установите 16.

4. Вновь выделите строку заголовка, справа появится желтый маркер , который переместите нажатой левой кнопкой мыши вниз для увеличения ширины строки заголовка.

5. На панели инструментов *Моделирование* нажмите на кнопку *Должность* , а затем щелкните вверху в центре. Появится фигура, в которой удалите имеющийся текст и введите *Генеральный директор*. Если необходимо, измените размер шрифта и размеры фигуры.

6. Аналогично в соответствии с рис. 59 добавьте другие фигуры типов *Должность* и *Организационная единица (вид 1)*. Обратите внимание, что размеры элементов типа *Должность* отличаются (фигуры верхнего уровня больше).

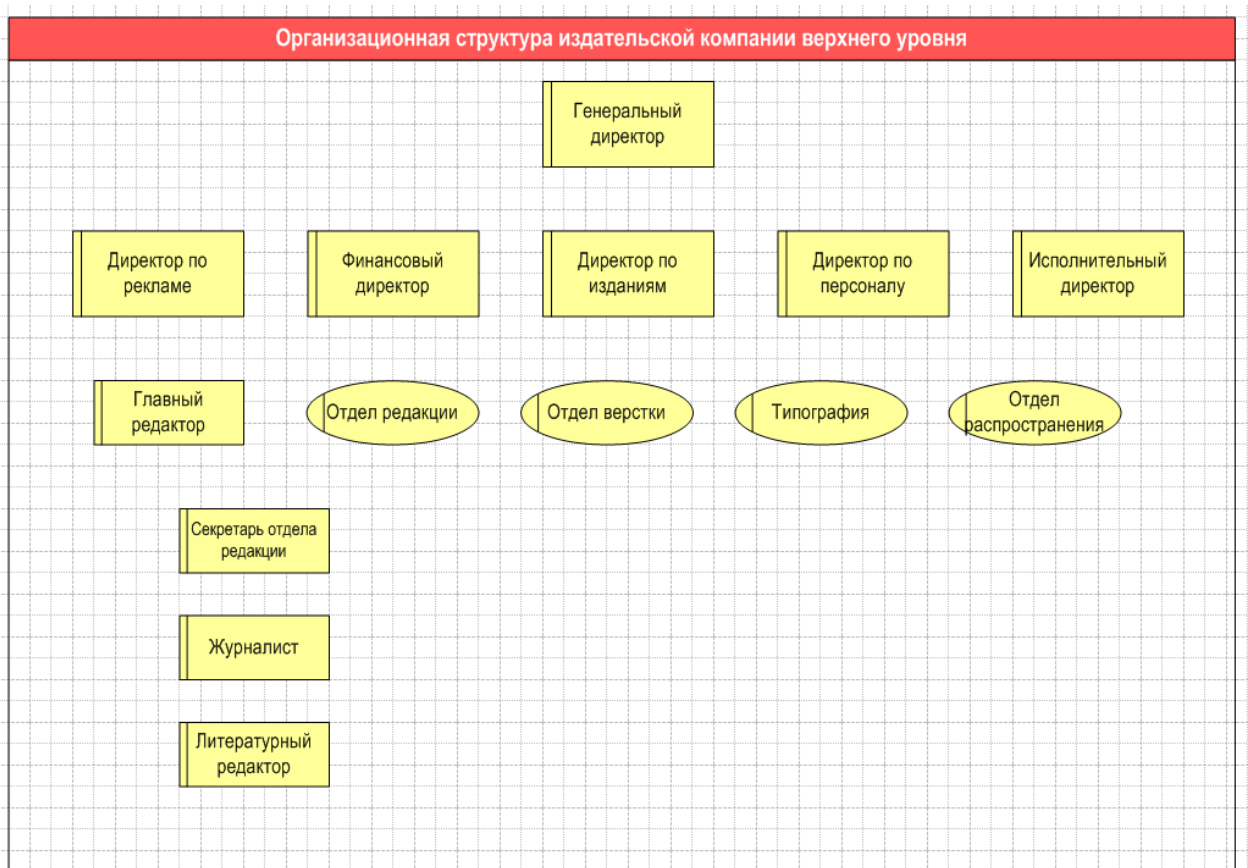



Рис. 59. Размещение на диаграмме элементов типа *Должность* и *Организационная единица (вид 1)*

Упражнение 3

Нарисуйте связи между элементами диаграммы, выполнив следующие действия:

1. На панели инструментов *Моделирование* нажмите на кнопку *Установить связь*  и на странице диаграммы, удерживая левую кнопку мыши, переместите курсор мыши от середины нижней границы элемента *Генеральный директор* к верхней границе *Директор по изданиям*.

2. В появившемся окне *Выбор связи* (рис. 60) выделите *Иерархическая связь*, установите галочку в поле *Выбирать ту же самую связь в дальнейшем* и нажмите кнопку *Применить*.

При проведении связи между двумя элементами на диаграмме необходимо контролировать, чтобы связь присоединилась к границам фигур.

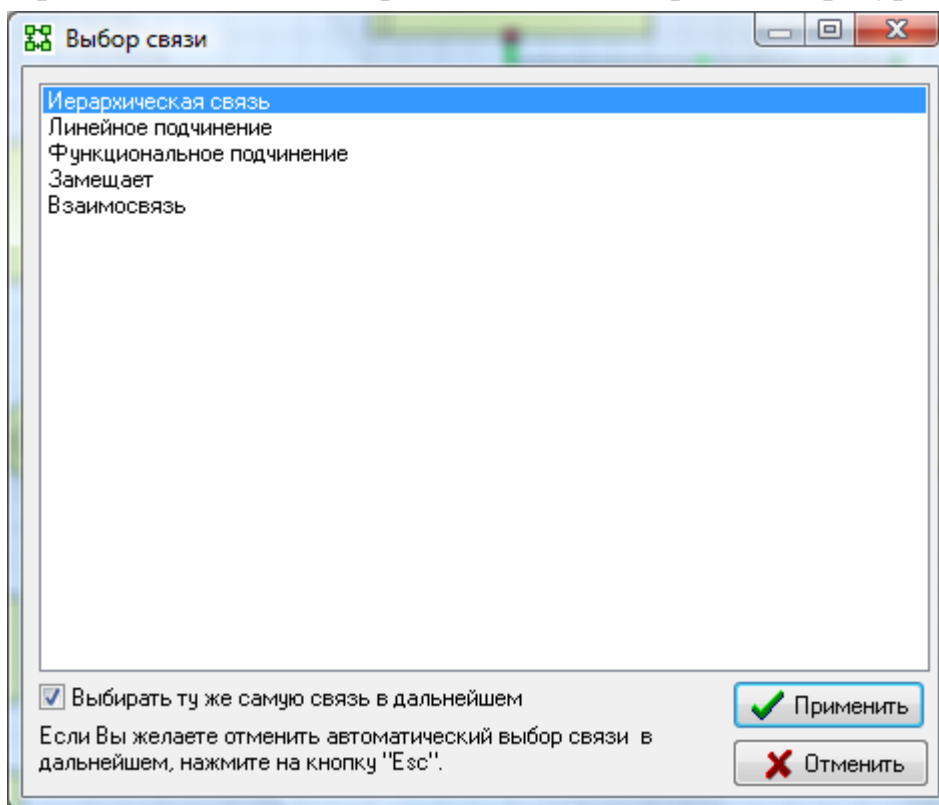


Рис. 60. Окно Выбор связи

3. Выделите созданную связь. На диаграмме точки присоединения к элементами должны стать красными (рис. 61).

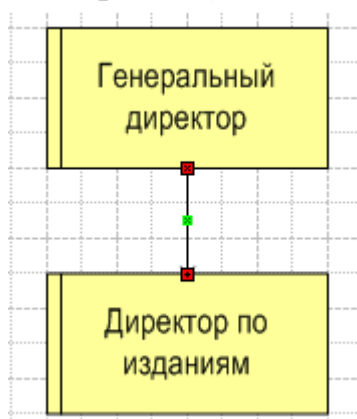



Рис. 61. Точки контроля присоединения связи к элементам

4. Вновь выделите созданную связь и на панели инструментов нажмите на кнопку *Элементы* , а затем выберите команду *Установить фор-*

мат линии. В появившемся окне измените толщину линии и форму конца линии, сделав ее более четкой.

5. Создайте другие связи между элементами в соответствии с рис. 62.

6. Сохраните диаграмму.

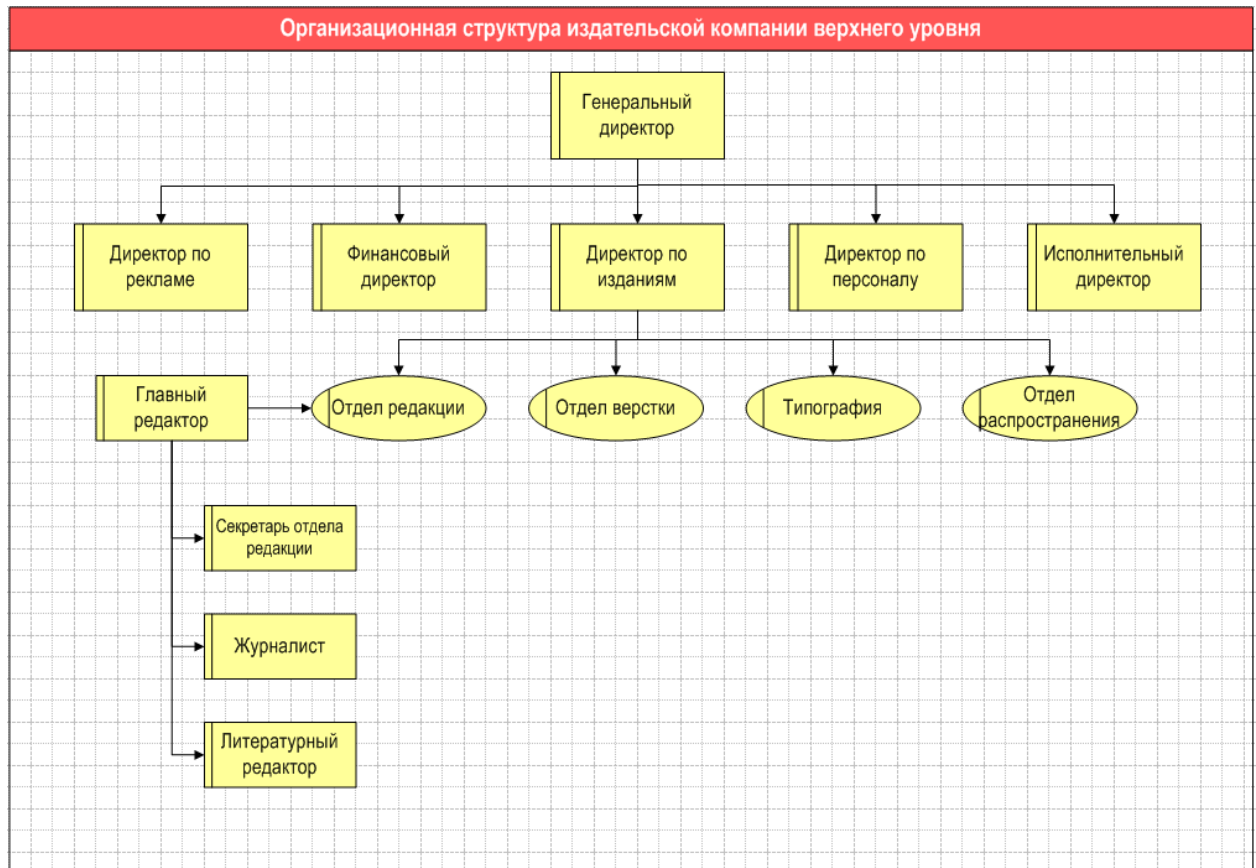


Рис. 62. Связи между элементами на диаграмме

Модель *ARIS Product/Service tree (PST)* (*Дерево продуктов и услуг*) описывает продукты и услуги, производимые компанией, а также связи со стратегическими целями компании, бизнес-процессами, поддерживающими их производство.

При построении диаграммы типа *ARIS Product/Service tree (PST)* наиболее часто используются фигуры, приведенные в табл. 26.

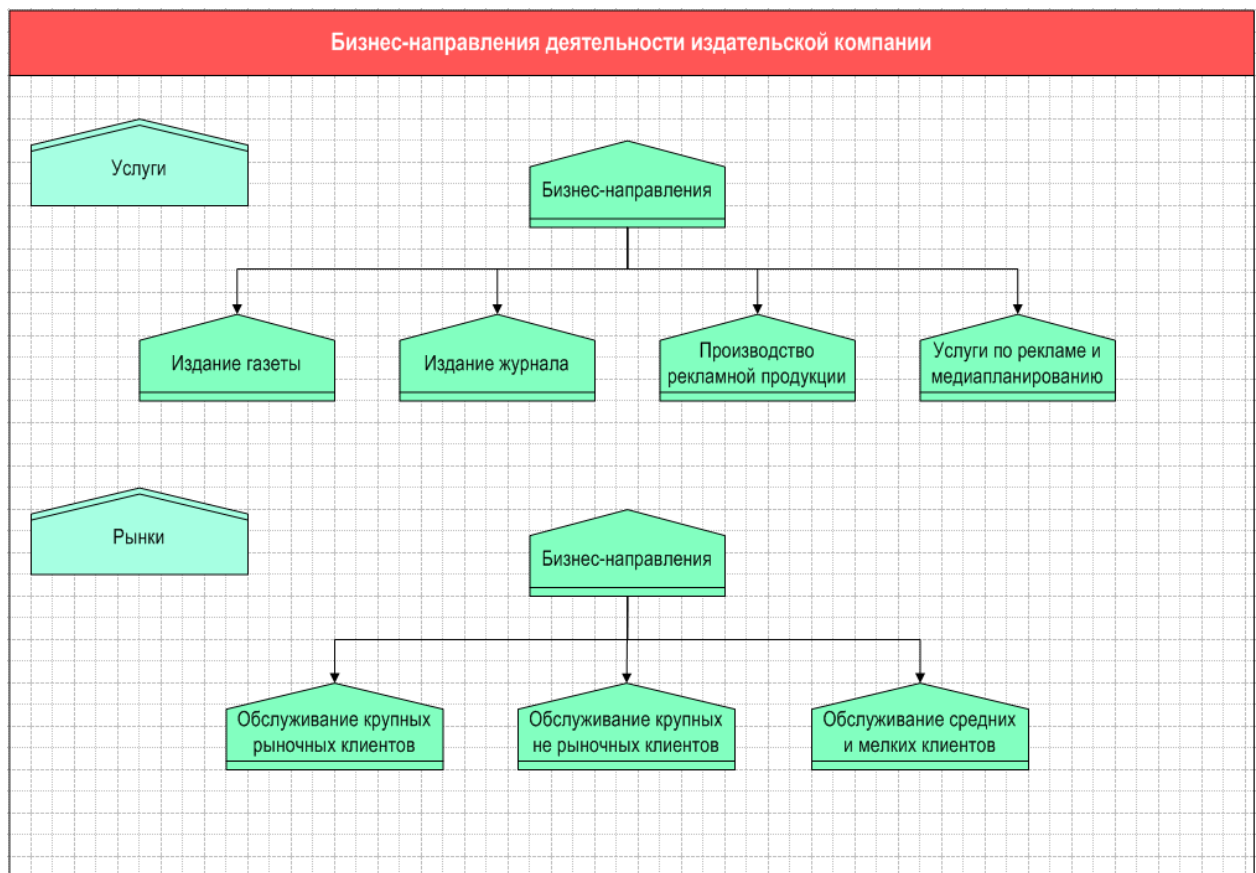
Таблица 26

Фигуры диаграммы *ARIS Product/Service tree (PST)*

Изображение фигуры	Название
	Продукт/Услуга
	Услуга
	Информационная услуга


Упражнение 4

1. Создайте новую диаграмму типа *ARIS Product/Service tree (PST)* (*Дерево продуктов и услуг*).
2. В соответствии с рис. 63 создайте диаграмму *Бизнес-направления деятельности издательской компании*.

Рис. 63. Диаграмма типа *ARIS Product/Service tree (PST)* (*Дерево продуктов и услуг*)

Модель *ARIS Function tree (FT)* – это граф, показывающий взаимоотношения между функциями. Модель описывает функции, выполняемые в компании и их иерархию. Данная модель часто используется для построения дерева бизнес процессов компании.

В этом типе модели функции могут быть описаны с различными уровнями детализации. При этом функции представляются не обязательно в хронологическом порядке. На самом верхнем уровне описываются наиболее сложные функции, представляющие собой отдельный бизнес-процесс или процедуру. Детализация функций образует иерархическую структуру их описаний. Разделение функций на элементы может происходить на нескольких иерархических уровнях. Базовые функции представляют самый нижний уровень в семантическом дереве функций. Базовая функция – это функция, которая уже не может быть разделена на составные элементы с целью анализа бизнес-процесса.

Основной фигурой, которая изображается на диаграмме, является *Функция* . В модели *ARIS Function tree* между блоками *Функция* допустимы *Процессно-ориентированные иерархические связи*.

Упражнение 5

1. Создайте новую диаграмму типа *ARIS Function tree (FT)* (*Дерево функций*).
2. Установите книжную ориентацию страницы.
3. В соответствии с рис. 64 создайте диаграмму *Бизнес-процессы компании первого уровня в форме дерева*.



Рис. 64. Диаграмма типа *ARIS Function tree (FT)* (Дерево функций)

Диаграмма процессов включает методологии *Data Flow Diagram (DFD)*, *Work Flow Diagram (WFD)* и *Swimmer Lanes*.

Стандарт описания бизнес-процессов *DFD - Data Flow Diagram* переводится как диаграмма потоков данных и используется для описания процессов верхнего уровня. На диаграмме потоков данных показываются работы, которые

входят в состав описываемого бизнес-процесса, а также – входы и выходы каждой из работ, представляющие из себя информационные или материальные потоки. При этом выходы одной работы могут являться входами для других.

При описании бизнес-процессов нижнего уровня используются процессные схемы, под названием *WFD – Work Flow Diagram*, что переводится как диаграмма потоков работ. На этой схеме появляются дополнительные объекты, с помощью которых описывается процесс: логические операторы, события начала и окончания процесса, а также элементы, показывающие временные задержки.

Подход *Swimmer lanes* представляет из себя смесь классических *DFD* и *WFD* стандартов и имеет одну отличительную особенность. Диаграмма, на которой рисуется схема бизнес-процесса, разделена по горизонтали на дорожки. Каждая дорожка принадлежит определенному структурному подразделению или должности, участвующей в бизнес-процессе. Те операции бизнес-процесса, которые выполняются этим структурным подразделением, размещаются в зоне соответствующей дорожки. Такой подход позволяет наглядно показать распределение ответственности в бизнес-процессе и продемонстрировать степень его организационной фрагментарности.





При построении диаграммы процессов наиболее часто используются фигуры, приведенные в табл. 27.

Таблица 27

Фигуры Диаграммы процессов

Изображение фигуры	Название
1	2
	Группа процессов (в форме горизонтальной дорожки)
	Группа процессов (в форме вертикальной дорожки)
	Процесс
	Организационная единица

Окончание таблицы 27

1	2
	Должность
	Решение
	Логический оператор «Исключающее ИЛИ»
	Логический оператор «И»
	Логический оператор «ИЛИ»

Упражнение 6


1. Создайте новую диаграмму типа *Диаграмма процесса (расширенная)*.
2. Установите книжную ориентацию страницы.
3. На панели инструментов *Моделирование* нажмите кнопку *Титульная рамка*, а затем щелкните на странице диаграммы. В рамку введите заголовок: *Бизнес-процессы компании первого уровня в форме сети и распределение ответственности*.
4. На панели инструментов *Моделирование* нажмите кнопку *Группа процессов (в форме горизонтальной дорожки)*  и щелкните на странице диаграммы.
5. Удалите стандартное название дорожки и введите *Бизнес-процессы управления*.
6. Выделите вставленную горизонтальную дорожку и путем перемещения зеленых маркеров разместите их таким образом, чтобы левый край дорожки совпадал с краем титульной рамки (рис. 65).



Рис. 65. Размещение группы процессов *Бизнес-процессы управления*

7. Разместите еще две горизонтальные группы процессов: *Основные бизнес-процессы* и *Обеспечивающие бизнес-процессы* (рис. 66).

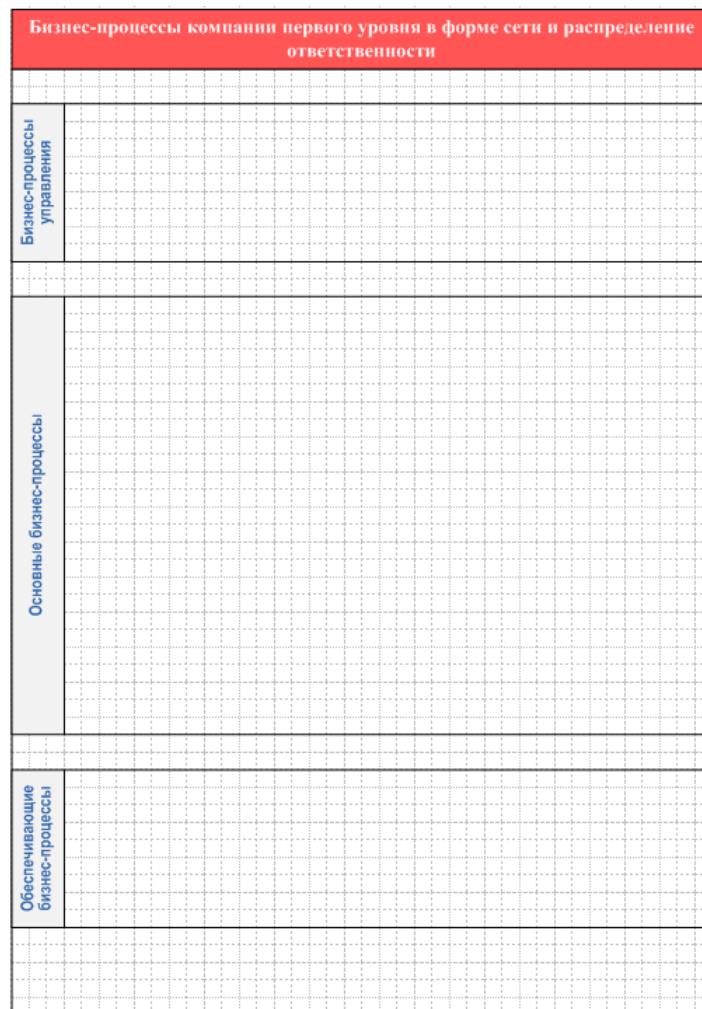


Рис. 66. Размещение на диаграмме горизонтальных групп процессов

8. Согласно рис. 67 и 68 в группах процессов *Бизнес-процессы управления* и *Обеспечивающие бизнес-процессы* нарисуйте фигуры типа *Процесс* и *Должность*, а между ними установите связь *Роль в процессе*.

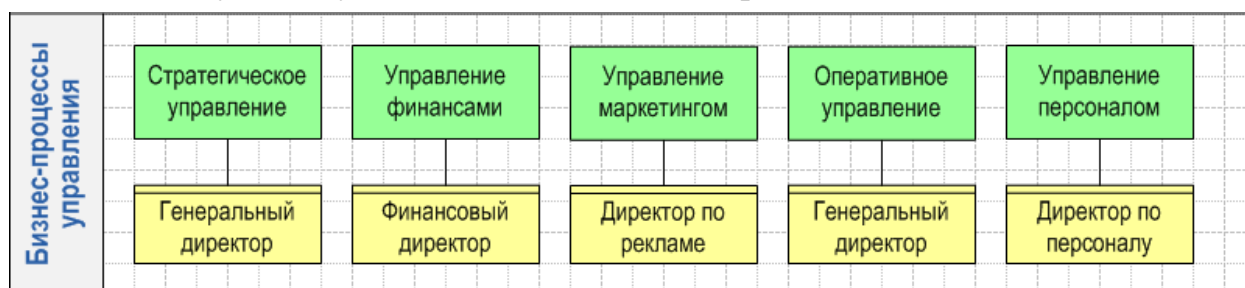


Рис. 67. Группа процессов *Бизнес-процессы управления*

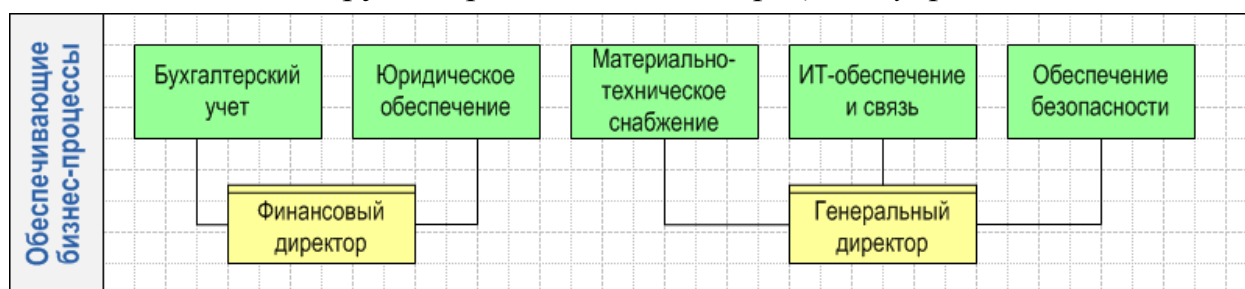


Рис. 68. Группа процессов *Обеспечивающие бизнес процессы*

9. Согласно рис. 69 в группе *Основные бизнес-процессы* нарисуйте фигуры типа *Процесс* и *Должность*, между ними установите связь *Роль в процессе*, а между однотипными элементами *Процесс* – *Связь предшествования* (пунктирные линии).

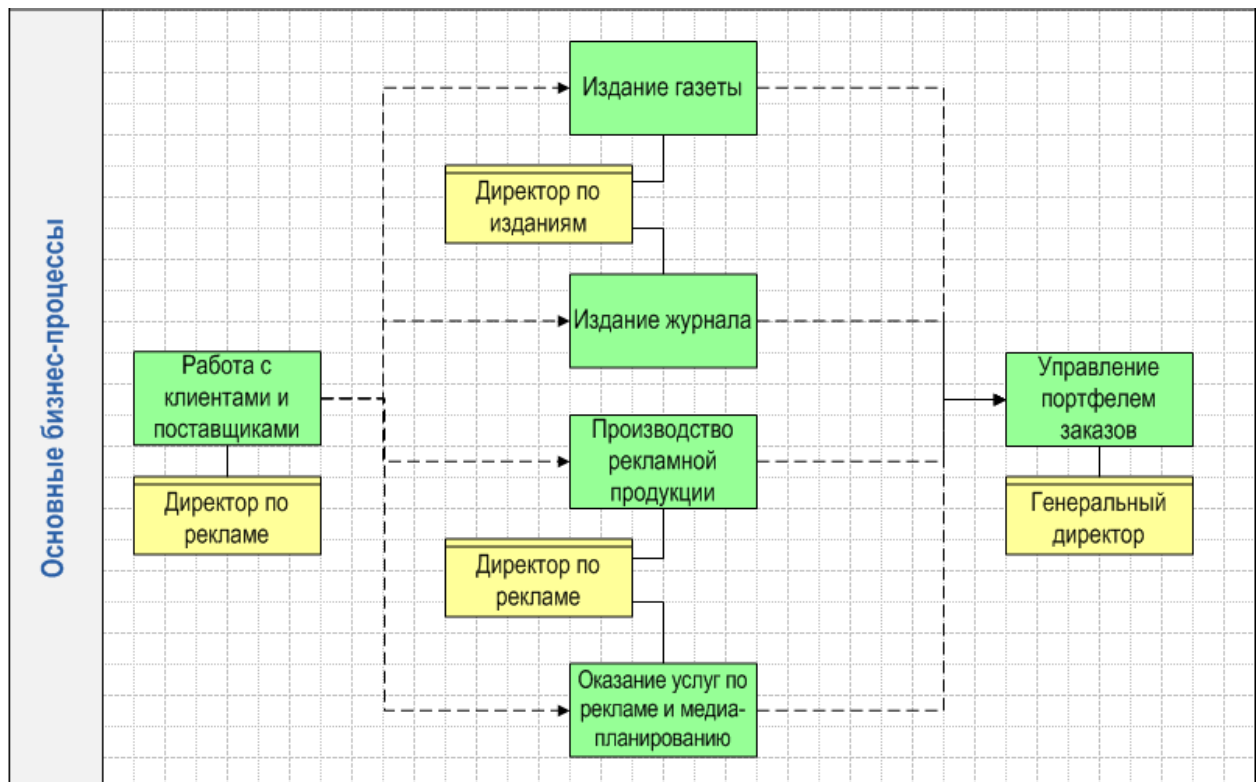



Рис. 69. Основные бизнес-процессы

10. На панели инструментов *Моделирование* нажмите на кнопку *Фигурная стрелка* , нарисуйте стрелку между группами бизнес процессов и поверните ее вниз. Результат представлен на рис. 70.

Вопросы для самопроверки

1. В чем заключается сущность методологии *ARIS*?
2. Расскажите о четырех типах моделей *ARIS*, отражающих различные аспекты предприятия.
3. Какие уровни описания поддерживают модели *ARIS*?
4. Что такое процесс? Какие основные группы процессов выделяют в организации при их декомпозиции?
5. Расскажите об основных составляющих процесса: владелец, вход, выход, ресурс.
6. Охарактеризуйте типы классификации процессов: основные, обеспечивающие, процессы управления организацией.
7. Для чего предназначена программа *График-студия Лайт*?
8. Дайте характеристику следующим типам диаграмм: *ARIS Organizational chart*, *ARIS Product/Service tree*, *ARIS Function tree* и *Диаграмме процесса*.

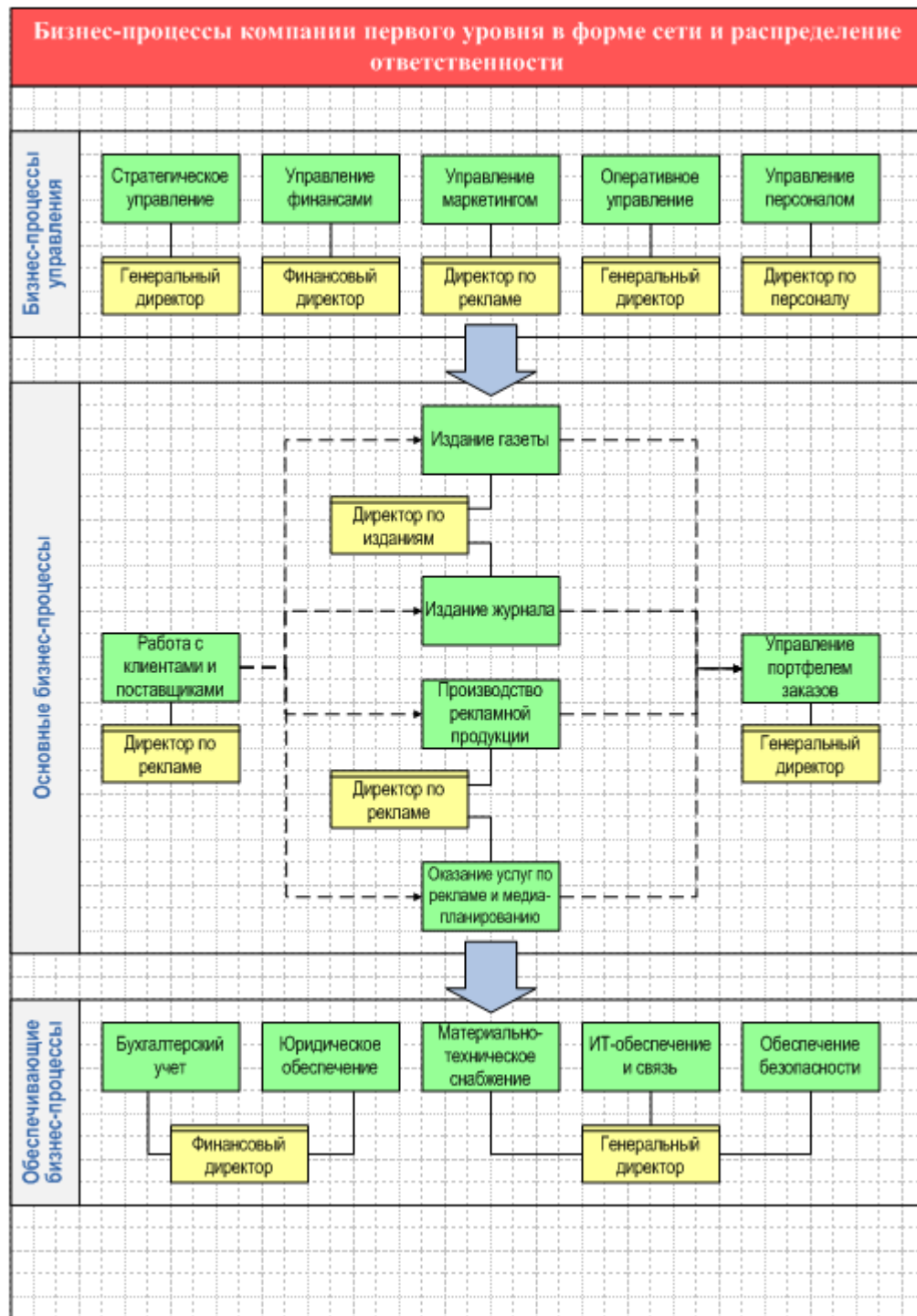


Рис. 70. Диаграмма Бизнес-процессы компании первого уровня в форме сети и распределение ответственности

ПРАКТИЧЕСКАЯ РАБОТА № 7

Тема работы: Создание функциональных диаграмм описания бизнес-процесса «Издание журнала» в *График-студии Лайт*.

В рамках функциональной модели *ARIS* базовыми являются следующие диаграммы:

- *Value added chain diagram (VAD)* – диаграмма цепочки добавленной стоимости;
- *Extended event driven process chain (eEPC)* – событийная цепочка процесса;
- *Function allocation diagram (FAD)* – диаграмма окружения функции.

Модель *ARIS Value added chain diagram (VAD)* (*Диаграмма цепочки добавленной стоимости*) описывает функции организации, которые непосредственно влияют на реальный выход ее продукции. Эти функции создают последовательность действий, формируя добавленные значения: стоимость, количество, качество и т.д. Модель цепочки добавленной стоимости показывает, из чего складывается конечная стоимость готового продукта, т.е. определяются процессы, добавляющие стоимость продукта.

Построение модели цепочки добавленной стоимости целесообразно начинать с обзорного представления взаимосвязанных частей процесса путем расположения элементов процесса согласно временной последовательности их выполнения. Затем необходимо отразить взаимосвязь различных элементов процесса путем нанесения соответствующих связей. После отображения в модели структуры процесса каждый из элементов процесса рассматривается с точки зрения необходимости его детализации. Если это требуется, то элемент детализируется на соответствующие блоки.







Аналогично дереву функций описываемые функции могут размещаться в диаграмме согласно иерархическому принципу, т.е. наиболее важные функции располагаются левее и выше. Эта иерархия всегда иллюстрирует подчинение функций. Кроме того, рассматриваемая диаграмма может представлять связи между функциями, организационными единицами и преследуемыми целями.

Графические фигуры, которые наиболее часто используются в диаграммах *ARIS Value added chain diagram (VAD)*, приведены в табл. 28.

Кластер используется для представления сложных объектов и является совокупностью некоторого количества связанных типов сущностей. Кластеры применяются для указания структуры информации, изменение которой фиксируется в событии.

Таблица 28

Фигуры диаграммы *ARIS Value added chain diagram*

Изображение фигуры	Название
	Цепочка добавленной стоимости (вид 1)
	Цепочка добавленной стоимости (вид 2), используемый как начальный элемент цепочки
	Кластер
	Организационная единица (вид 1)
	Организационная единица (вид 2)
	Ключевой показатель

Упражнение 1.

1. Запустите *График-студิโอ Лайт* и создайте новую диаграмму типа *ARIS Value added chain diagram (VAD)*.

2. Установите альбомную ориентацию страницы.

3. В соответствии с рис. 71 создайте диаграмму *Схема бизнес-процесса «Издание журнала»*, между элементами которой установите следующие типы связей:

– *Связь предшествования* – между фигурами типа *Цепочка добавленной стоимости*;

– *Входная связь* и *Выходная связь* – между фигурами типов *Кластер* и *Цепочка добавленной стоимости* в зависимости от направления стрелки;

– *Роль в процессе* – между фигурами типов *Должность* и *Цепочка добавленной стоимости*.

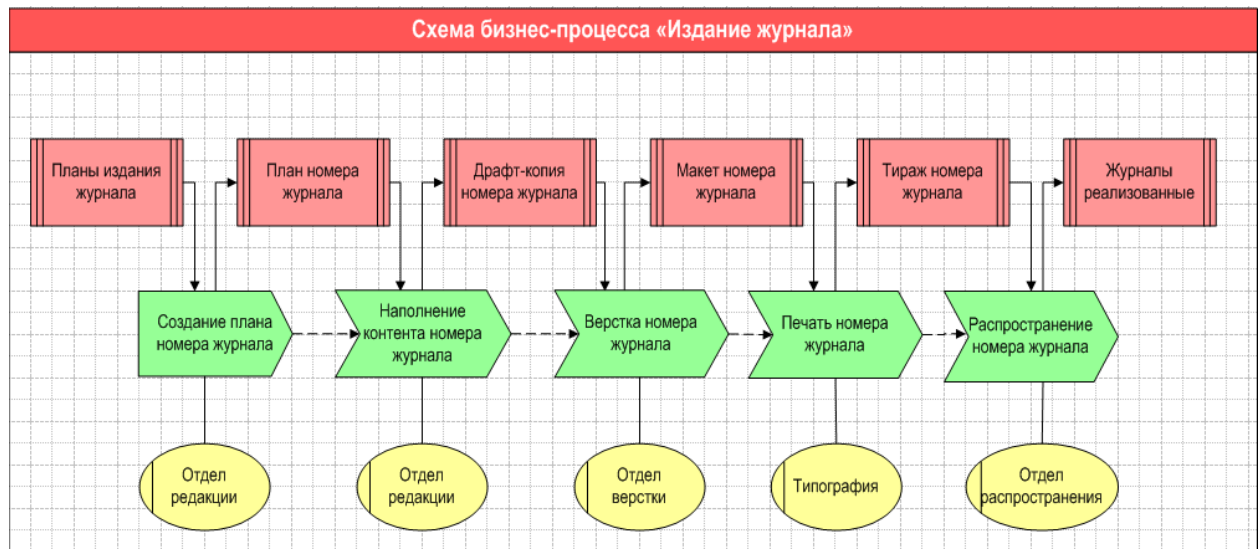


Рис. 71. Диаграмма типа *ARIS Value added chain diagram (VAD)*

Модель *ARIS Extended event driven process chain (eEPC)* (*Расширенная событийная цепочка процесса*) предназначена для детального описания процессов, выполняемых в рамках одного подразделения или конкретным сотрудником, и отражает последовательность функциональных шагов (действий) в рамках одного бизнес-процесса.

Модель *eEPC* строится с использованием четырех групп элементов: функциональные элементы, логические элементы, элементы данных и организационные элементы (табл. 29). При использовании всех групп элементов получается всесторонняя модель бизнес-процесса, показывающая основные действия, выполняемые конкретными сотрудниками с помощью прикладных и технических устройств.












Процедурная последовательность функций в рамках бизнес-процессов отображается в виде цепочки процесса, где для каждой функции могут быть определены начальное и конечное событие. *Событие* – это состояние, которое является существенным для целей управления бизнесом и которое оказывает влияние или контролирует дальнейшее развитие одного или более бизнес-процессов.

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. 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.

Таблица 29

Фигуры диаграммы *ARIS Extended event driven process chain (eEPC)*

Изображение фигуры	Название
	Событие
	Функция
	Логический оператор «И»
	Логический оператор «Исключающее ИЛИ»
	Логический оператор «ИЛИ»
	Должность
	Организационная единица (вид 1)
	Организационная единица (вид 2)
	Кластер
	Внешняя персона
	Ключевой показатель

Моделирование нелинейной последовательности выполнения функций (разветвление, объединение, цикл) осуществляется с помощью логических вы-

ражений, которые состоят из параметров состояния объектов, характеризующих событие (операнды), и логических операций.

Основные правила построения *eEPC*-диаграммы:

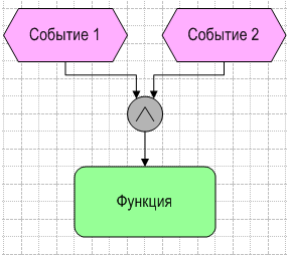
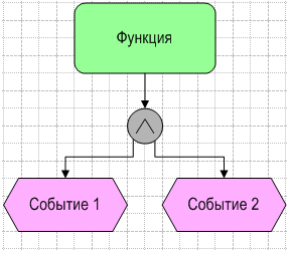
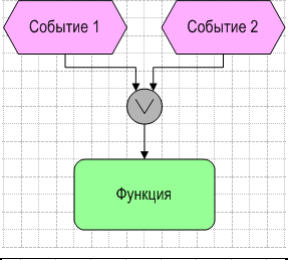
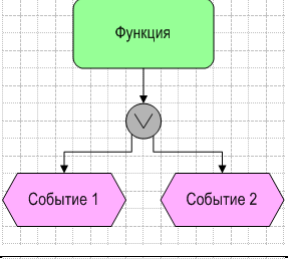

- каждая функция инициируется событием и завершается так же событием;
- в каждую функцию может входить только одна стрелка, инициирующая выполнение функции, и из каждой функции может выходить только одна стрелка, описывающая завершение выполнения функций;
- оператор разветвления может иметь один вход и неограниченное число выходов;
- оператор соединения фрагментов бизнес-процесса может иметь неограниченное число входов и только один выход.

Одно событие может инициировать выполнение одновременно нескольких функций, и наоборот, функция может быть результатом наступления нескольких событий. Эти ветвления и циклы обработки отображаются на диаграмме *eEPC* с помощью соединителей (в виде небольшого кружка), которые не только отображают графические связи между элементами модели, но и определяют логические связи между объектами.

Различают два типа операторов: операторы событий и операторы функций. В табл. 30 и 31 описаны все возможные операторы связи событий и функций.

Особое внимание необходимо уделить ограничениям, которые существуют для операторов функций. Поскольку события не могут принимать решения (в то время как функции могут), переключающееся событие не должно быть связано операторами «ИЛИ» или «Исключающее ИЛИ».

Описание операторов связей событий в *eEPC* модели

Логический оператор		Изображение связи	Описание связи
1	2	3	4
«И»	Связь событий, инициирующих выполнение функции		Выполнение функции начинается после того, как произойдут все события
	Связь событий после выполнения функции		В результате выполнения функции происходят все события
«ИЛИ»	Связь событий, инициирующих выполнение функции		Для исполнения функции необходимо выполнение хотя бы одного события
	Связь событий после выполнения функции		В результате выполнения функции происходит по крайней мере одно событие
«Исключающее ИЛИ»	Связь событий, инициирующих выполнение функции		Функция начинает выполняться после того, как произойдет одно (и только одно) событие

Окончание таблицы 30

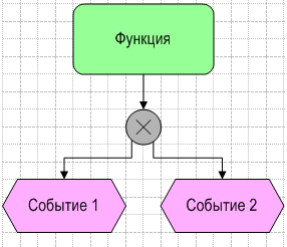
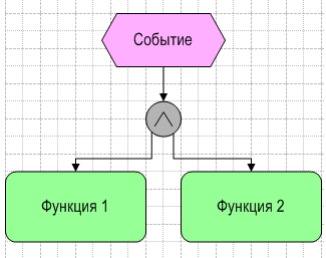
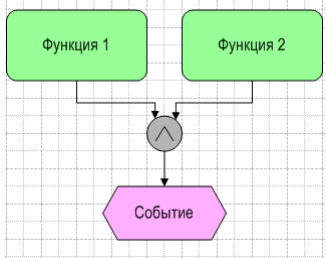
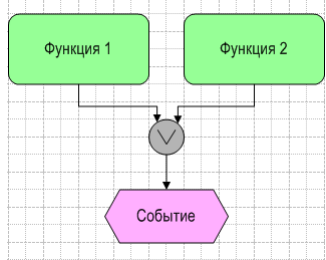
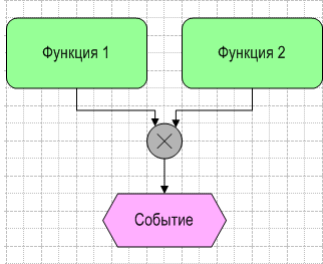
1	2	3	4
«Исключающее ИЛИ»	Связь событий после выполнения функции		В результате выполнения функции происходит только одно из событий

Таблица 31

Описание операторов связей функций в *eEPC* модели

Логический оператор		Изображение связи	Описание связи
1	2	3	4
«И»	Связь функций, инициируемых событием		Событие определяет необходимость обязательного выполнения всех функций
	Связь функций, завершающих событием		Событие происходит после того, как все функции выполнены
«ИЛИ»	Связь функций, инициируемых событием	Не используется	
	Связь функций, завершающих событием		Событие произойдет после того, как будет выполнена хотя бы одна функция

Окончание таблицы 31

1	2	3	4
«Исключающее ИЛИ»	Связь функций, инициируемых событием	Не используется	
	Связь функций, завершающих событием		Событие произойдет после того, как будет выполнена одна (и только одна) функция

Упражнение 2

1. Создайте новую диаграмму типа *ARIS Extended event driven process chain (eEPC-расширенная)*.
2. Установите книжную ориентацию страницы.
3. В соответствии с рис. 72 создайте диаграмму бизнес-процесса «Издание журнала/Наполнение контента номера журнала».

В диаграммах *eEPC* все символы могут располагаться в столбцах (или строках). Преимущество такого представления состоит в том, что *eEPC*-диаграмму проще интерпретировать. Элементы организационной структуры и информационной системы выносятся в заголовок диаграммы. Все другие символы размещаются во второй строке каждого столбца.

Отличительная особенность всех моделей, которые представляются с помощью строк и столбцов, заключается в том, что в них автоматически создаются невидимые отношения. Связь *Роль в процессе* (означает *Выполняет*) будет создана автоматически между элементами организационной структуры и функциями.

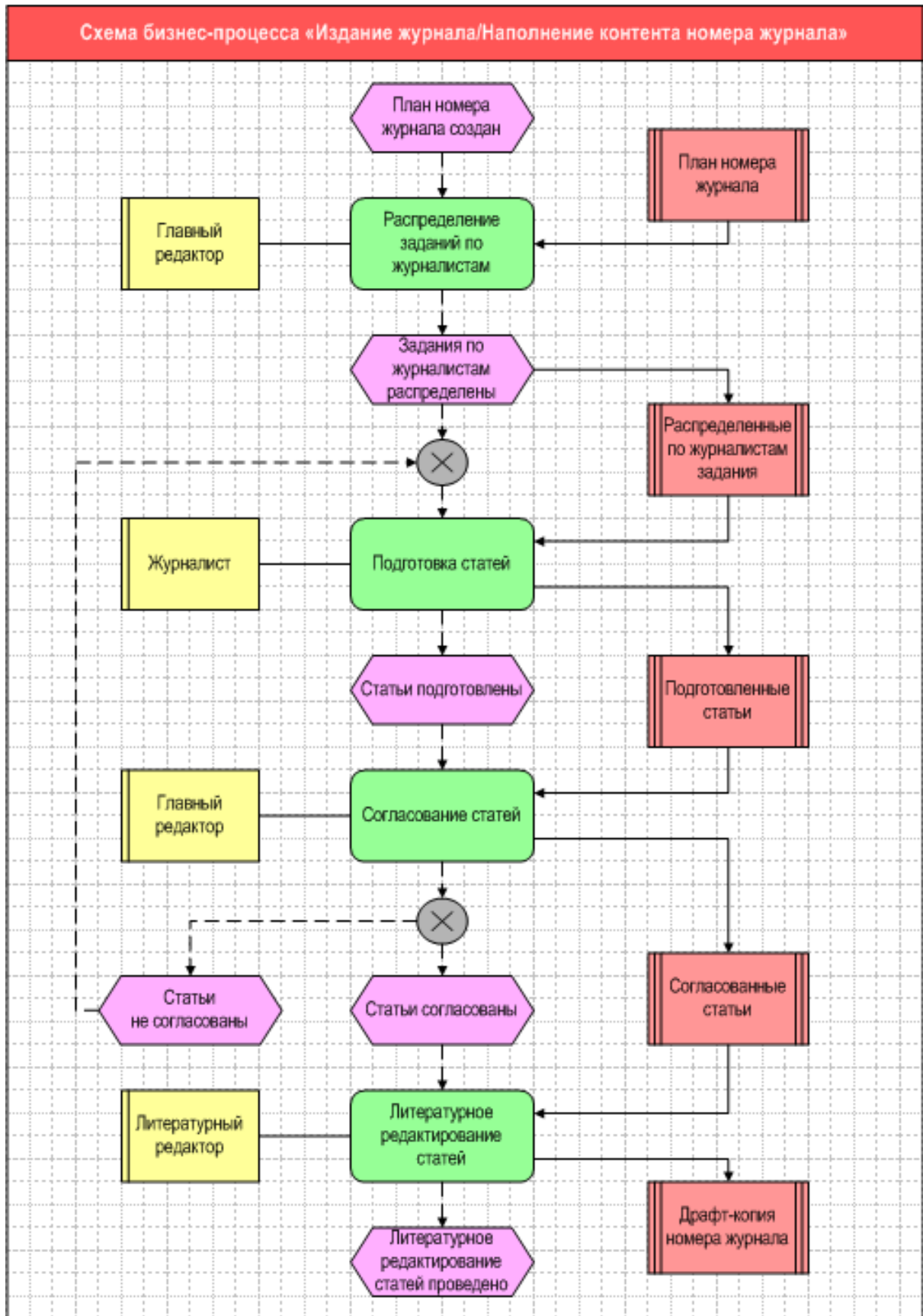


Рис. 72. Диаграмма ARIS *Extended event driven process chain* (eEPC-расширенная)

Упражнение 3

1. Создайте новую диаграмму типа *Диаграмма процесса (расширенная)*.

2. Установите книжную ориентацию страницы.



3. В соответствии с рис. 73 создайте диаграмму бизнес-процесса «Издание журнала/Наполнение контента номера журнала» в виде столбцов. Для того, чтобы подписать стрелку, следует ее выделить и на панели инструментов *Редактирование* нажать на кнопку *Вставить и редактировать текстовый блок* . В появившееся поле следует вводить необходимый текст. Для изменения местоположения вставленного текста следует выделить его, появится желтый маркер *Изменить положение текста* , переместите который в то место, куда необходимо поместить текст.

Диаграмма окружения функции *ARIS Function allocation diagram (FAD)* предназначена для того, чтобы описать все объекты, которые окружают функцию: исполнителей, входные и выходные потоки информации, документы, материалы, продукты/услуги, а так же используемое оборудование. Этот тип моделей целесообразно применять для детализации функций в модели *eEPC*, в результате чего отражаются дополнительные связи и отношения, детализирующие эту функцию на уровне данных. Основные элементы модели *FAD*, такие же, как в *eEPC*-модели, за исключением событий и правил (логических операторов). *FAD*-модель показывает:

- потоки входной информации, необходимой для реализации данной функции;
- потоки выходной информации, образующейся в результате переработки входной;
- физические лица или группы людей, принимающие участие в выполнении описываемого процесса;
- средства связи, посредством которых информация поступает в систему.

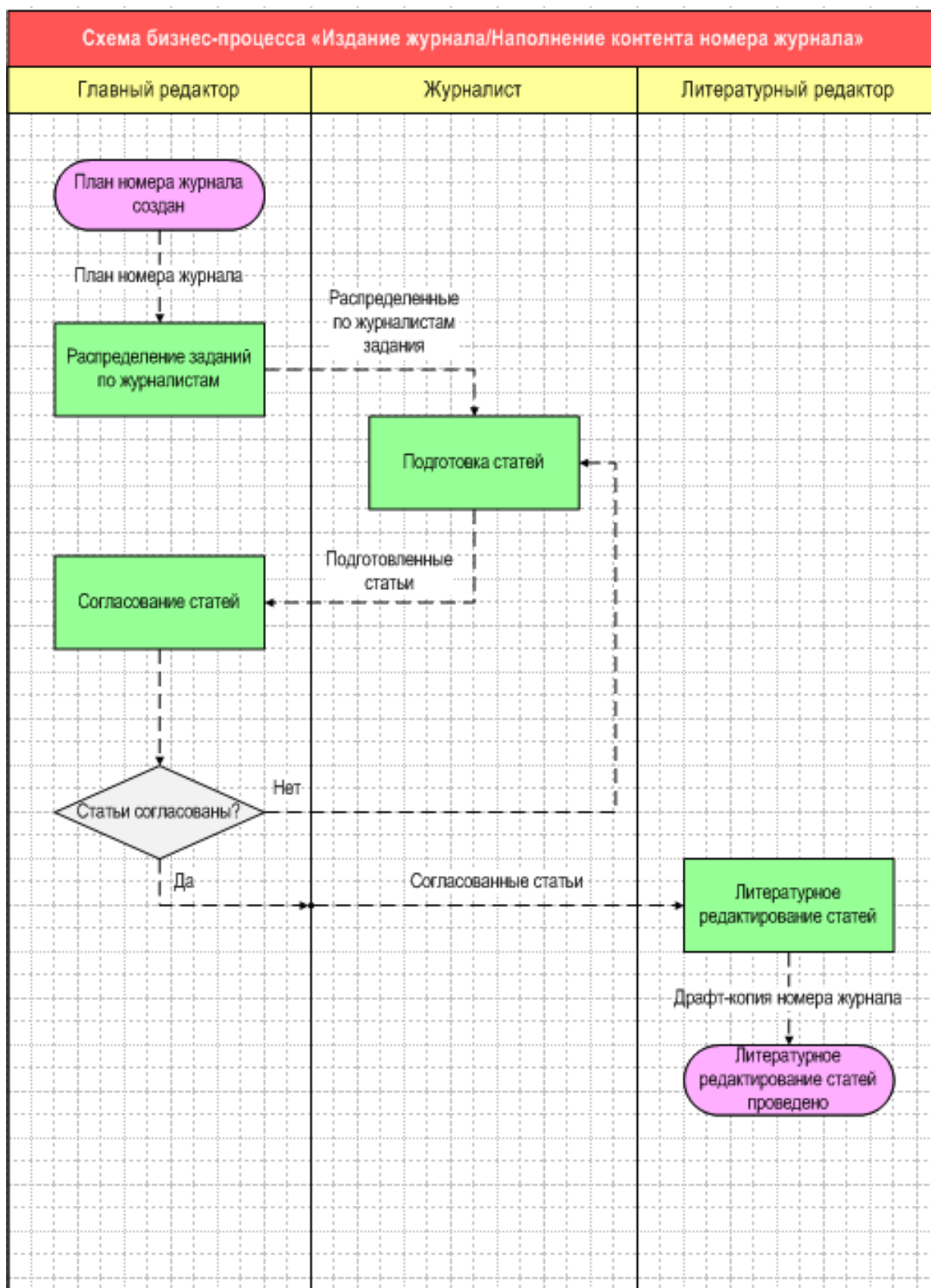


Рис. 73. Диаграмма бизнес-процесса «Издание журнала/Наполнение контента номера журнала» в виде столбцов

Упражнение 4

1. Создайте новую диаграмму типа *ARIS Extended event driven process chain (eEPC-расширенная)*.
2. Установите альбомную ориентацию страницы.
3. В соответствии с рис. 74 создайте диаграмму окружения бизнес-процесса «Издание журнала».

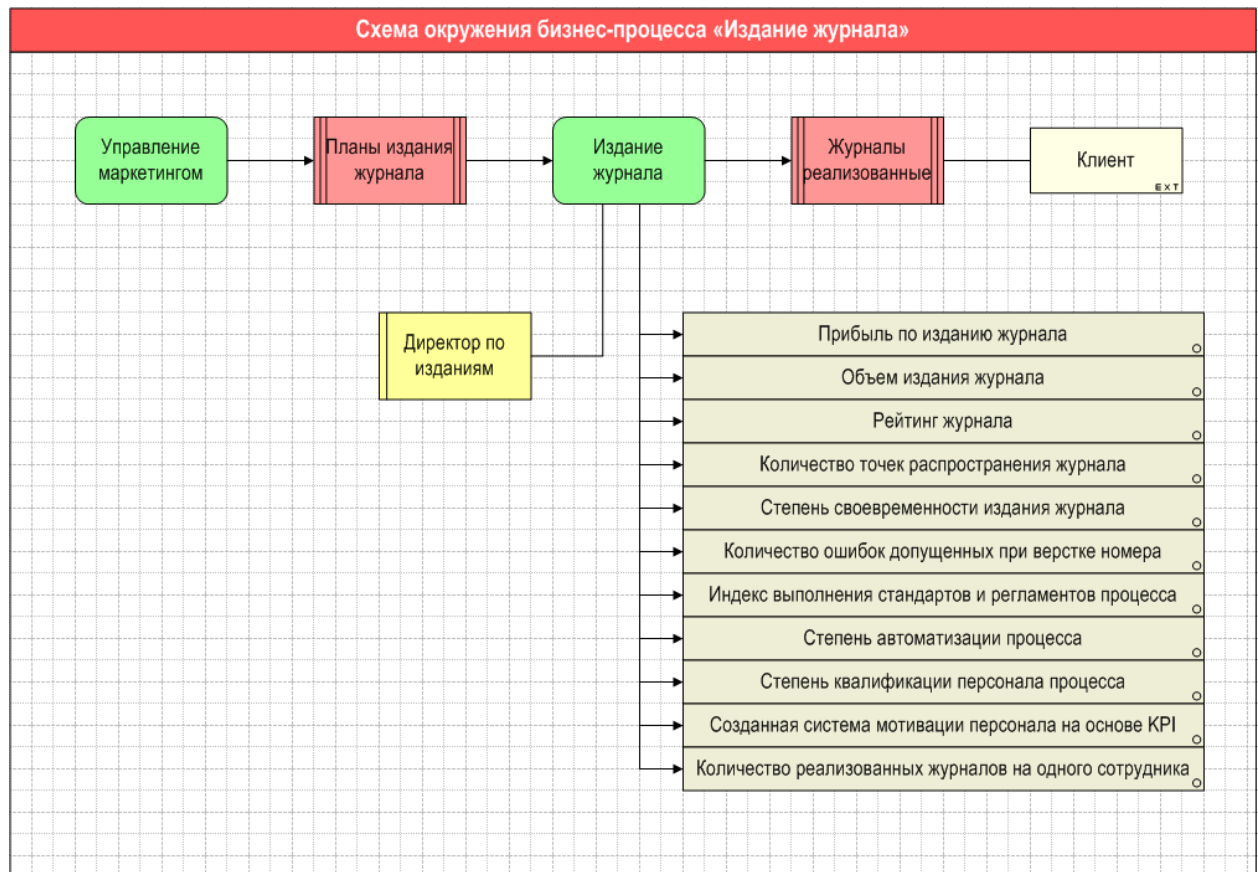


Рис. 74. Диаграмма *Схема окружения бизнес-процесса «Издание журнала»*

Вопросы для самопроверки

1. Дайте характеристику диаграмме цепочки добавленной стоимости: назначение, принципы создания, основные фигуры.
2. Охарактеризуйте диаграмму *Extended event driven process chain (eEPC)*. В чем заключается различие событий от функций, изображаемых на диаграмме.
3. Расскажите правила использования логических операторов на диаграмме *eEPC*.
4. Какие особенности представления диаграммы *eEPC* в виде столбцов?
5. Дайте характеристику диаграммы окружения бизнес-процесса.

ПРАКТИЧЕСКАЯ РАБОТА № 8

Тема работы: Моделирование бизнес-процессов в нотации *BPMN*.

Нотация BPMN 2.0 разработана компанией OMG в 2013 г. легла в основу международного стандарта моделирования бизнес-процессов ISO/IEC 19510:2013.

Основными элементами диаграммы процесса в нотации BPMN являются:

- пул;
- дорожка;
- событие;
- задача;
- шлюз или условие;
- элементы деловых коммуникаций;
- объект и хранилища данных.

Для представления этих элементов нотация BPMN предлагает более 100 символов, однако на практике используется гораздо меньше.

Элементы *Пул* и *Дорожка* используются для отражения взаимодействия участников процесса в ходе его реализации. Под *пулом* следует понимать совокупность всех операций процесса и ответственных лиц за их исполнение. Таким образом, он предназначен для обозначения границ процесса. Для отражения ответственных исполнителей (ролей в процессе) используется элемент *дорожка*, поэтому в рамках одного пула могут находиться несколько дорожек.

В нотации BPMN пул с дорожками, который является диаграммой бизнес-процесса, принято называть *оркестровкой*.

Пулы с дорожками могут иметь как горизонтальное, так и вертикальное расположение. Какое из них выбрать при описании процесса, зависит от количества исполнителей и операций, которые содержит бизнес-процесс, поскольку в методологии моделирования одним из основных требований, предъявляемых к схеме процесса, является размещение диаграммы только на одном листе. Так, если в процесс занято много исполнителей, то дорожки лучше размещать горизонтально, если немного (2-4), то вертикально.

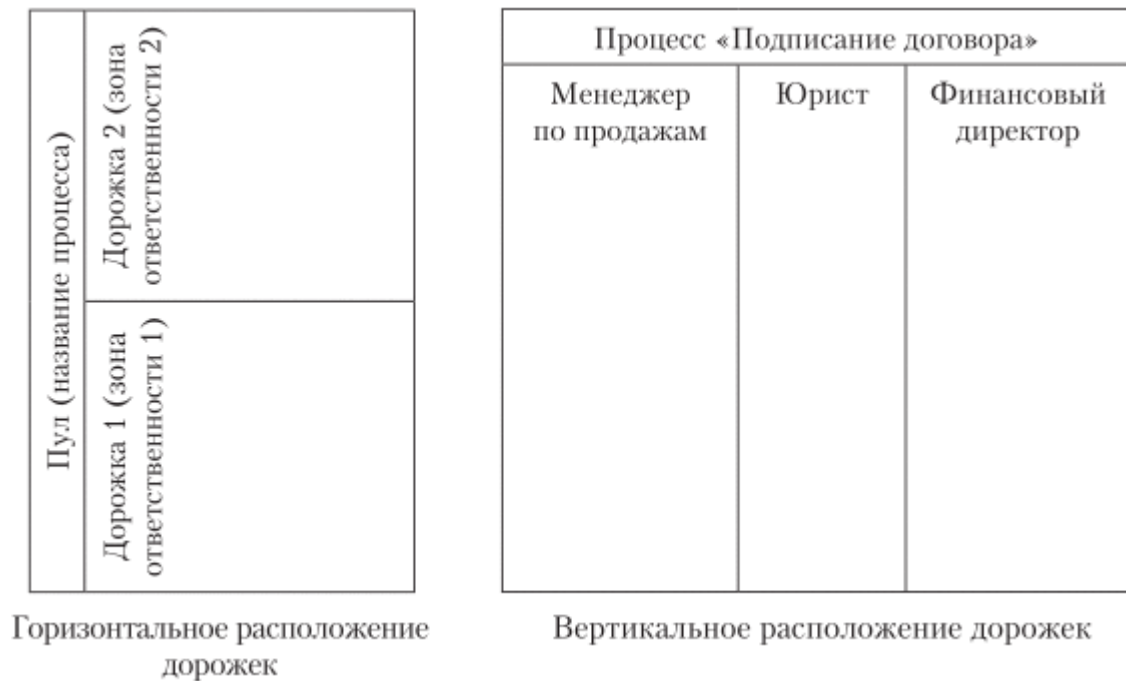


Рис. 75. Расположение пула с дорожками

Элемент *Событие*. События бывают стартовые, конечные и промежуточные. Каждое из них имеет несколько разновидностей. Чаще используют простое событие, событие-таймер и событие-сообщение. Для того чтобы указать тип события на диаграмме процесса, в центре кружка события указывают специальный символ.

Стартовое событие показывает, с какого момента начинается выполнение процесса, т.е. начальную точку процесса, поэтому никакой входящий поток не может поступать в стартовое событие. Оно отражается пустым кружком (предпочтительно зеленого цвета). Наиболее универсальным типом является *простое стартовое событие*. Его можно использовать в любых случаях. Если же нужно акцентировать внимание на том, что процесс начинается только при получении определенного сообщения, то можно применить стартовое событие-сообщение. Таким типом события также можно показать, что запуск процесса происходит при условии получения сведений об оплате товара или услуги. В случае, если процесс запускается в строго определенное время или через определенный промежуток времени, то целесообразно использовать символ стартового события-таймера.

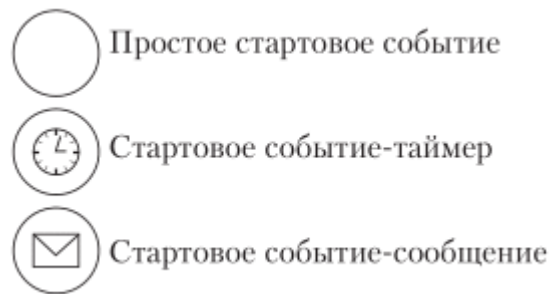


Рис. 76. Типы стартовых событий

Например, стартовым событием процесса «Оказание консультации по телефону» является звонок клиента.

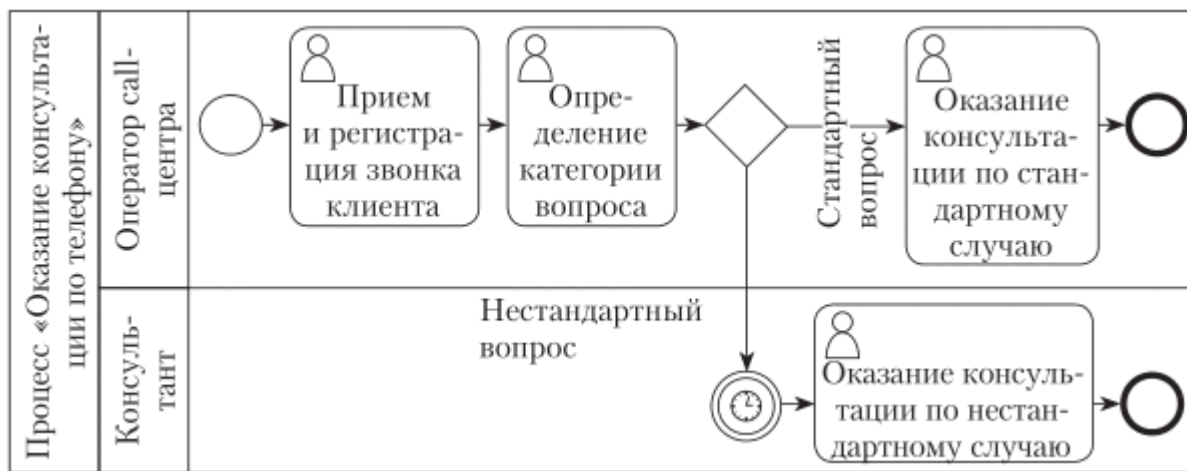


Рис. 77. Модель процесса «Оказание консультации по телефону»

Заканчивается процесс одним или несколькими *конечными событиями*. Таким образом, обозначаются моменты, где завершается процесс, поэтому никакие выходящие потоки и операции не могут стоять после конечного события. Простое конечное событие отражается в виде пустого кружка с жирными линиями предпочтительно красного цвета (рис. 78). Окружность, обозначенная именно жирной линией, помогает отличить конечное событие от стартового. Кроме простого часто используют конечное событие-сообщение, показывающее, что процесс завершается отправлением информационного сообщения участником.

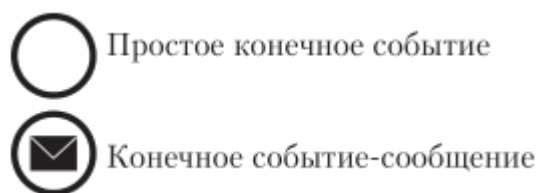


Рис. 78. Типы конечных событий

На рис. 78 (модели процесса «Оказание консультации по телефону») простые конечные события показывают, что процесс «Оказание консультации по телефону» завершается после выполнения операции «Оказание консультации по стандартному случаю» или после операции «Оказание консультации по нестандартному случаю».

Таким образом, все, что находится между стартовым и конечным событиями, показывает, каким образом осуществляется или должен осуществляться бизнес-процесс.

Для описания непосредственно бизнес-процесса используются *промежуточные события*, которые влияют на ход выполнения процесса, но не являются какими-либо действиями или событиями начала и конца процесса. Промежуточные события отражаются в виде кружка, где окружность изображается двойной линией (рис. 79). Двойная линия позволяет отличить промежуточное событие от стартового или конечного.

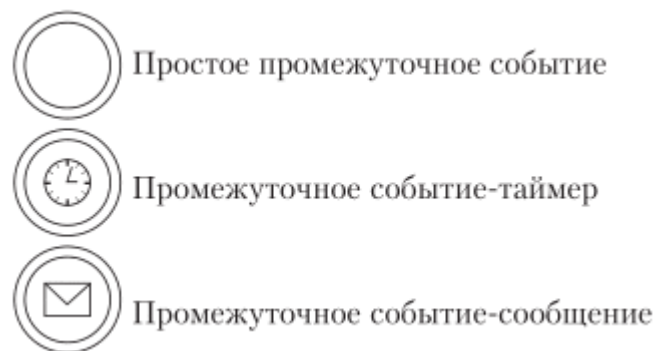


Рис. 79. Типы промежуточных событий

В качестве примера на рис. (модели процесса «Оказание консультации по телефону») приведено *промежуточное событие-таймер*, показывающее, что клиенту для получения консультации по нестандартной ситуации нужно подождать, когда консультант освободится и сможет ответить на его вопрос. Данный тип промежуточного события в общем виде предназначен для указания ситуации, при которой выполнение процесса приостанавливается на определенное или неопределенное время.

Другим типом промежуточного события является *событие-сообщение*, показывающее, что на этом этапе реализации процесса происходит отправление информационного сообщения другому участнику процесса.

Кроме описанных выше типов событий в BPMN 2.0 предлагается к использованию и такие события, как:

- эскалация – для переноса рассмотрения вопроса на более высокий уровень иерархии;
- условное событие – для отражения реакции на изменение деловых условий или интеграции бизнес-правил;
- ошибка – для отражения генерации и обработки конкретного типа ошибок;
- отмена – для иллюстрации события, связанного с отменой транзакции или инициированием отмены;
- остановка – для указания немедленного прекращения выполнения процесса.

Элемент «Задача» используется для представления на диаграмме процесса действий, выполняемых в рамках описываемого бизнес-процесса (например, подготовки документа, согласования, подписания, оказания консультационной услуги, изготовления образца продукта и т.д.). Таким образом, *Задача* – основной элемент модели процесса в BPMN-нотации. На схеме она изображается в виде прямоугольника с закругленными углами.

Нотация BPMN предполагает использование различных типов задач, из которых наиболее часто используются следующие (рис. 80):

- пользовательская задача;
- ручное выполнение;
- сценарий;
- подпроцесс;
- получение сообщения;
- отправка сообщения.

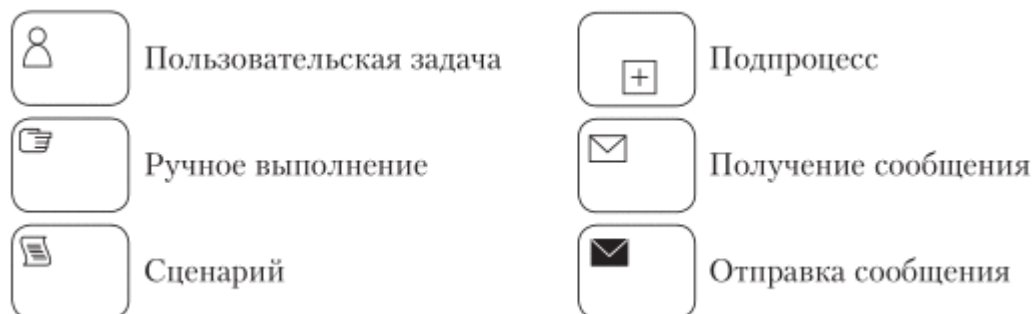


Рис. 80. Типы задач

Графически тип задач можно идентифицировать по символу, изображенному в блоке задачи в верхнем левом углу. На схеме все задачи (кроме задач «Получение сообщения» и «Отправка сообщения») связываются друг с другом

линией со стрелкой на конце, которая показывает последовательность выполнения действий.

Пользовательская задача применяется тогда, когда исполнителем действия является человек. Как правило, таких задач в бизнес-процессах большинство. Например, «Прием и регистрация звонка клиента». На схеме такая задача изображена в виде прямоугольника, в верхнем левом углу которого изображен человек.

Задача типа *ручное выполнение* используется для представления на диаграмме действий, которые выполняются людьми без использования каких-либо механизмов исполнения или информационных систем, т.е. для информационной системы данное действие является неуправляемым, поскольку находится вне зоны контроля, и она никоим образом не может отследить выполнение данной операции. Примером таких действий могут служить проведение собеседования и устранение неисправности оборудования заказчика на месте. Время начала и окончания выполнения документа в системе не может быть зафиксировано, поэтому, как правило, переход к реализации следующей операции осуществляется в ручном режиме ответственным исполнителем данного процесса. Такой тип задач на диаграмме процесса представляется в виде прямоугольника с символом руки в верхнем левом углу.

Задача типа *Сценарий* позволяет описывать действия, которые осуществляются информационной системой в автоматическом режиме и без участия человека. На схеме процесса она представлена в виде прямоугольника, в верхнем левом углу которого находится символ в виде листа бумаги (свитка). Например, определение типа договора или количества дней до истечения срока обработки информации.

Задача типа *Подпроцесс* является особой категорией задач, поскольку используется не для отражения одного действия, а некоторой последовательности действий. На диаграмме подпроцесс изображается в виде прямоугольника с закругленными углами, внутри которого снизу по центру указывается символ «+».

На практике элемент типа *Подпроцесс* используется для выполнения декомпозиции или для описания повторяющихся действий.

В целях упрощения схемы процесса задача данного типа позволяет показать в виде одного прямоугольника подпроцесс, для которого строится собст-

венная, уточняющая его реализацию диаграмма. Таким образом, осуществляется принцип декомпозиции в BPMN-нотации. Поэтому для построения процессов верхнего уровня преимущественно используется именно этот тип задач.

Кроме того, BPMN-нотация позволяет применять подпроцессы для отражения типовых часто повторяющихся операций внутри как одного процесса, так и разных, например:

- процесс «Контроль исполнения» (операция осуществляется параллельно с процессом «Исполнение входящих документов» и «Выполнение приказов и распоряжений генерального директора компании»);
- процесс «Согласование проекта документа» (состоит из написания замечаний или проставления визы согласования), который осуществляется, как правило, несколькими должностными лицами в рамках реализации одного процесса «Разработка документа ...».

Такой подход позволяет сократить время на создание одинаковых схем процессов, а также сократить риски возникновения ошибок при моделировании. При необходимости внесения изменений в часто повторяющийся процесс (например, в целях оптимизации), корректировка одной диаграммы предпочтительнее корректировки нескольких.

В качестве примера использования различных типов задач на рис. 81. представлена BPMN-диаграмма процесса «Обработка заказа на изготовление кондитерского изделия».

Поскольку оформление заказа осуществляется конкретным сотрудником отдела продаж, то данное действие, как и операцию согласования даты и времени доставки, изображаем в виде пользовательской задачи. Определение категории изделия осуществляется специализированной информационной системой на основании тех данных, которые ввел сотрудник отдела продаж в ходе выполнения предыдущего действия. Поэтому данную операцию изображаем в виде задачи типа «сценарий». Поскольку изготовление заказов «торт» и «пирог» представляет собой цепочку операций, то в целях упрощения восприятия схемы процесса на данной диаграмме они показаны в виде подпроцессов. Так как доставка заказа осуществляется сотрудником отдела доставки (курьером) и никакая информационная система не управляет ее деятельностью, то данная операция относится к типу «ручное выполнение».

Задачи «Получение сообщения» и «Отправка сообщения» будут рассмотрены далее (вместе с рассмотрением элементов деловых коммуникаций).

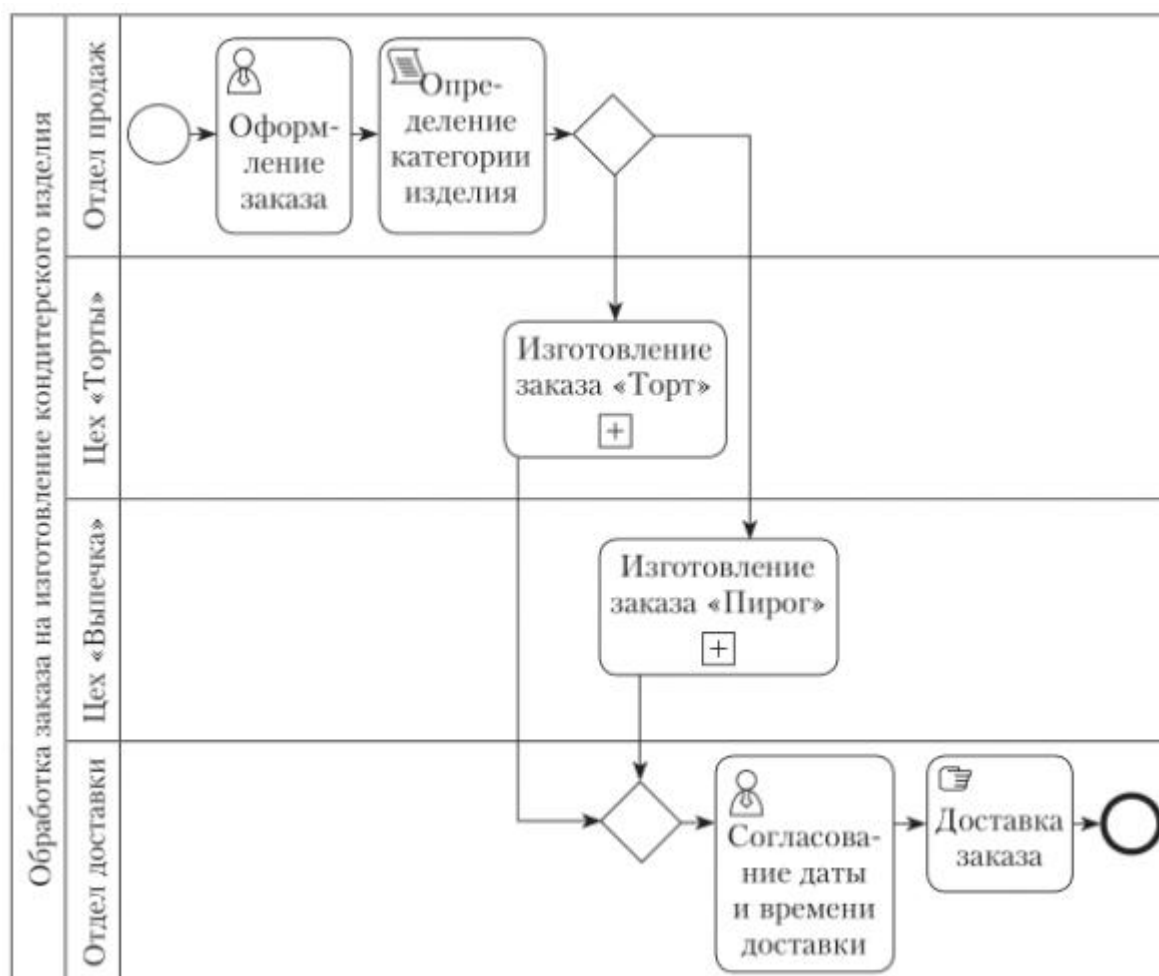


Рис. 81. Обработка заказа на изготовление кондитерских изделий

Вопросы для самопроверки

1. Дайте характеристику диаграмме цепочки добавленной стоимости: назначение, принципы создания, основные фигуры.
2. Охарактеризуйте диаграмму *ВРМН*. В чем заключается различие событий от функций, изображаемых на диаграмме.
3. Расскажите правила использования логических операторов на диаграмме *eEPC*.
4. Какие особенности представления диаграммы *eEPC* в виде столбцов?
5. Дайте характеристику диаграммы окружения бизнес-процесса.

ПРАКТИЧЕСКАЯ РАБОТА № 9

Тема работы: Система имитационного моделирования *Arena*. Построение модели процесса обработки документов.

Имитационное моделирование – это метод исследования, при котором изучаемая система заменяется моделью, с достаточной точностью описывающей реальную систему, с которой проводятся эксперименты с целью получения информации об этой системе. Экспериментирование с моделью называется *имитацией* (имитация – это постижение сути явления, не прибегая к экспериментам на реальном объекте).

Существует класс объектов, для которых по различным причинам не разработаны аналитические модели или методы их решения. В этом случае аналитическая модель заменяется имитационной моделью.

Имитационная модель – это логико-математическое описание объекта, которое может быть использовано для экспериментирования на компьютере в целях проектирования, анализа и оценки функционирования объекта.

Цель имитационного моделирования состоит в воспроизведении поведения исследуемой системы на основе результатов анализа наиболее существенных взаимосвязей между ее элементами, т.е. в разработке симулятора исследуемой предметной области для проведения различных экспериментов.

Имитационное моделирование позволяет имитировать поведение системы во времени. При этом временем можно управлять: замедлять в случае с быстропротекающими процессами и ускорять для моделирования систем с медленной изменчивостью. Можно имитировать поведение тех объектов, реальные эксперименты с которыми дороги, невозможны или опасны.

Система имитационного моделирования Arena

Система имитационного моделирования *Arena* разработана корпорацией *Rockwell Automation Incorporation* и позволяет создавать динамические модели разнородных процессов и систем, оптимизировать построенную модель. Система *Arena* имеет удобный объектно-ориентированный интерфейс и обладает широкими функциональными возможностями по адаптации к различным предметным областям:

- исследованию и анализу бизнес-процессов промышленных предприятий, страховых компаний, банковской сферы;
- прогнозированию рентабельности внедрения новых технологий и капитальных вложений в оборудование;
- анализу сложных производственных процессов, включающих интенсивные операции по транспортировке материалов;
- определению оптимальных режимов производственных линий и программ обслуживания клиентов в системах массового обслуживания;
- совершенствованию сложных экономических процессов и систем;
- эффективному управлению складскими запасами, снабжения и перевозок;
- созданию моделей сложных динамических систем и т.д.

Основой технологии моделирования *Arena* являются язык моделирования *SIMAN* и анимационная система *Cinema Animation*.


Процесс моделирования в системе *Arena* организован следующим образом: сначала в визуальном редакторе пользователь строит модель, затем система генерирует по ней соответствующий код на *SIMAN*, запускается имитационный процесс, по окончании которого генерируются разнообразные отчеты.

Упражнение 1

Запустите систему *Arena*. На экране появится окно (рис. 81), которое содержит 3 области: область построения модели; область свойств объектов модели и панель инструментов проекта (*Project Bar*).

Область построения модели предназначена для отображения графических диаграмм, технологических схем, анимации и других элементов чертежа. *Область свойств объектов* в виде таблицы отображает параметры объектов модели.

Панель *Project Bar* состоит из трех вкладок:

- *Basic Process (Основной процесс)* – содержит графические объекты, называемые модулями, которые используются для построения модели (табл.). Модули помещаются в область построения модели путем перетаскивания, а соединяются – с помощью инструмента *Connect (Соединение)* , расположенного на стандартной панели инструментов;

- *Reports (Отчеты)* – содержит шаблоны отчетов, которые доступны для отображения результатов моделирования;
- *Navigate (Навигация)* – позволяет передвигаться по различным уровням модели, быстро менять виды и масштабы.

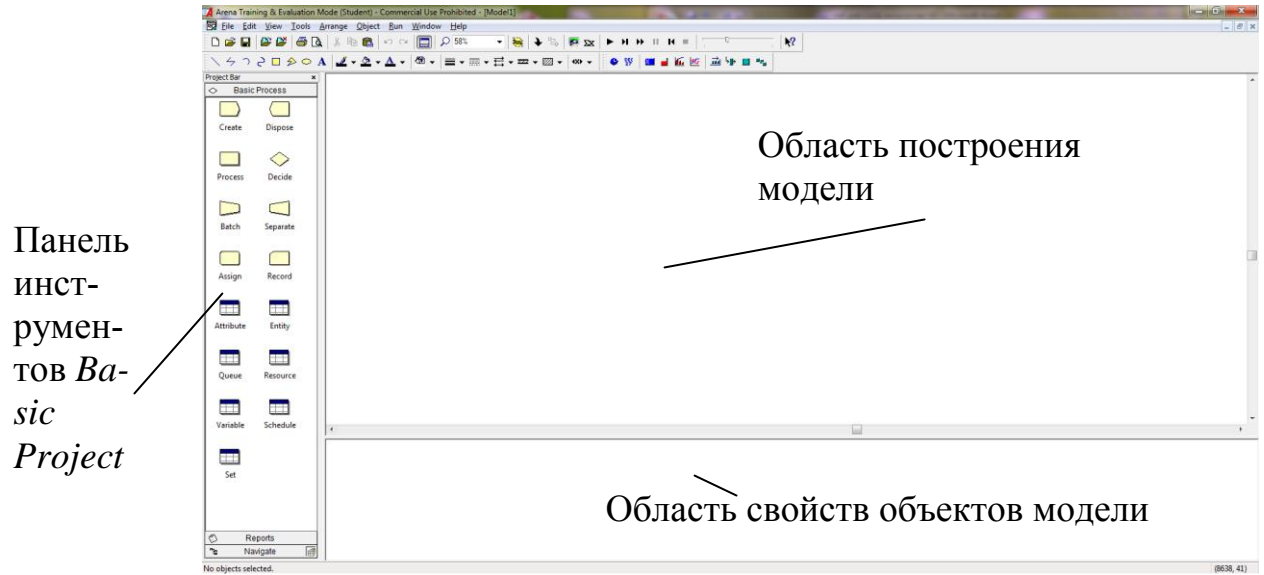







Рис. 81. Окно системы *Arena*




Основные объекты панели *Basic Process*

Название и изображение модуля	Описание	Применение
1	2	3
Графический модуль <i>Create</i>  Create	Модуль является отправной точкой для сущностей (заявки, люди, сообщения) в имитационной модели. Сущности – это индивидуальные элементы, обрабатываемые в системе. Создание сущностей модулем происходит по расписанию или же, основываясь на значении времени между прибытиями сущности в модель. Покидая модуль, сущности начинают обрабатываться в системе. В этом модуле определяется тип создаваемых сущностей и время их создания. Параметры модуля приведены в табл.	<ul style="list-style-type: none"> – Прибытие документов в сфере бизнеса (заказы, чеки, документация); – прибытие клиентов в сфере обслуживания (в магазин); – начало изготовления продукции на производственной линии и т.д.
Графический модуль <i>Dispose</i>  Dispose	Модуль является выходной точкой из имитационной модели. Статистика о сущности может собираться до того момента, пока она не выйдет из системы. Параметры модуля приведены в табл.	<ul style="list-style-type: none"> – Окончание бизнес-процесса; – клиенты покидают отдел и т.д.

Продолжение таблицы 32

1	2	3
<p>Графический модуль <i>Process</i></p>  <p>Process</p>	<p>Модуль является основным модулем процесса обработки сущностей в имитационной модели. В модуле имеются опции использования ресурсов, можно задать стоимостные и временные характеристики процесса обработки сущности. Параметры модуля приведены в табл.</p>	<ul style="list-style-type: none"> – Проверка документов; – выполнение заказов; – обслуживание клиентов; – обработка деталей и т.д.
<p>Графический модуль <i>Decide</i></p>  <p>Decide</p>	<p>Модуль позволяет описать и задать логику модели, учитывая принятие решений в модели. Опции принятия решений основаны на условии или на вероятности. Если условие выполняется, то сущности покидают систему через ветвление <i>Истина (True)</i>, иначе – <i>Ложь (False)</i>. Параметры модуля приведены в табл.</p>	<ul style="list-style-type: none"> – Разделение дел на срочные и несрочные; – перенаправление недоделанных или сделанных неправильно работ на доработку и т.д.
<p>Модуль данных <i>Entity</i></p>  <p>Entity</p>	<p>Модуль <i>Сущность</i> определяет тип сущности и ее анимационную картинку в имитационном процессе, а также – стоимостную информацию. Для каждого источника должен быть определен тип сущности, который он генерирует. Параметры модуля приведены в табл.</p>	<ul style="list-style-type: none"> – Документы: факсы, письма, отчеты; – люди в моделях магазина, ресторана и т.д.

Продолжение таблицы 32

1	2	3
<p>Модуль данных <i>Queue</i></p> 	<p>Модуль данных <i>Очередь</i> предназначен для изменения правила расстановки сущностей в очереди. По умолчанию – <i>First in First out (Первый пришел, первый ушел)</i>. Параметры модуля приведены в табл.</p>	<ul style="list-style-type: none"> – Стопка документов, ожидающих освобождения ресурса; – очередь покупателей у кассы в магазине; – место для сбора частей, ожидающих упаковки и т.д.
<p>Модуль данных <i>Resource</i></p> 	<p>Модуль предназначен для определения ресурсов и их свойств в имитационном процессе. Может включать в себя стоимостную информацию о ресурсах и их вместимость ресурсов. Ресурсы могут иметь фиксированную вместимость или ж основанную на расписании. Ресурс должен быть связан с каким-либо процессом. Параметры модуля приведены в табл.</p>	<ul style="list-style-type: none"> – Люди (сотрудники); – оборудование (станки, компьютеры).
<p>Модуль данных <i>Schedule</i></p> 	<p>Модуль Расписание может использоваться вместе с модулем <i>Resource</i> для определения вместимости ресурса и с модулем <i>Create</i> – для задания расписания прибытия сущностей.</p>	<ul style="list-style-type: none"> – Расписание работы персонала с перерывами на обед и др.

Окончание таблицы 32


1	2	3
<p>Модуль данных <i>Set</i></p> 	<p>Модуль данный описывает группу ресурсов, использующихся в модуле <i>Process</i>. В группе могут находиться несколько ресурсов. Модуль <i>Set</i> автоматически создает ресурсы, вместимость которых по умолчанию равна 1, без стоимостной информации.</p>	<p>– Организация работы группы работников по очереди.</p>

Таблица 33

Параметры модуля *Create*

Параметры	Описание
1	2
Name	Уникальное имя модуля, которое будет отражено в блок-схеме. Задается английскими буквами.
Entity Type	Название типа сущности, который будет создаваться модулем
Type	Способ формирования потока прибыли. Type может иметь значения: <i>Random</i> (используется экспоненциальное распределение со средним значением, определенным пользователем), <i>Schedule</i> (определяется модулем <i>Schedule</i>), <i>Constant</i> (будет использоваться постоянное значение, определенное пользователем) или <i>Expression</i> (поток прибытия будет формироваться по определенному выражению).
Value	Определяет среднее значение времени между прибытиями сущностей
Schedule name	Имя расписания, которое определяет характер прибытия сущности в систему

Окончание таблицы 33

1	2
Expression	Этот параметр задает тип распределения или любое выражение, определяющее время между прибытиями сущностей в модель
Units	Единицы измерения времени между прибытиями (день, час, минута, секунда)
Entities per arrival	Количество сущностей, входящих в систему за одно прибытие
Max arrivals	Максимальное число сущностей, которое может создать этот модуль (ресурс генератора)
First creation	Время, через которое прибудет первая сущность в модель, от начала моделирования

Таблица 34

Параметры модуля *Decide*

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок-схеме
Record Entity Statistics	Определяет, будет ли вестись статистика о выходе сущности из системы

Таблица 35

Параметры модуля *Process*

Параметры	Описание
1	2
Name	Уникальное имя модуля, которое будет отражено в блок-схеме
Type	Определяет логическую схему модуля. <i>Standard</i> означает, что логическая схема находится внутри модуля и зависит от параметра <i>Action</i> . <i>Submodel</i> показывает, что логическая схема будет находиться ниже в иерархической модели. Подмодель может содержать любое количество логических модулей.

1	2
Action	<p>Тип обработки, происходящей внутри модуля, может быть четырех типов:</p> <ul style="list-style-type: none"> – <i>Delay</i> просто показывает, что процесс занимает какое-то время и не отражает использование ресурсов; – <i>Seize Delay</i> указывает на то, что в этом модуле были размещены ресурсы, и будет происходить их захват и задержка, ресурсы будут захватываться (т.е. будут заняты обработкой сущности), а их освобождение будет происходить позднее с помощью какого-то другого модуля; – <i>Seize Delay Release</i> указывает на то, что ресурсы были захвачены, а затем (через время) освободились; – <i>Delay Release</i> означает, что ресурсы до этого были захвачены сущностью, а в таком модуле сущность задержится и освободит ресурс. <p>Все эти параметры доступны только тогда, когда <i>Type = Standard</i></p>
Priority	Значение приоритета модулей, использующих один и тот же ресурс где угодно в модели. Это свойство недоступно, если <i>Action = Delay</i> (или <i>Delay Release</i>) или когда <i>Type = Submodel</i>
Resources	Определяет ресурсы или группы ресурсов, которые будут обрабатывать сущности в этом модуле
Delay Type	Тип распределения или процедура, определяющая параметры задержки
Units	Единицы измерения времени задержки (день, час, минута, секунда)
Allocation	Определяет стоимостные характеристики обработки. <i>Value Added</i> означает учитывать стоимостные характеристики, а <i>Non-Value Added</i> – не учитывать
Minimum	Поле, определяющее минимальное значение для равномерного и треугольного распределения

Окончание таблицы 35

1	2
Maximum	Поле, определяющее максимальное значение для равномерного и треугольного распределения
Value	Поле, определяющее среднее значение для нормального и треугольного распределения или значения для постоянной временной задержки
Std Dev	Параметр, определяющий стандартное отклонение для распределения
Expression	Поле, в котором задается выражение, определяющее значение временной задержки, если <i>Delay Type = Expression</i>

Таблица 36

Параметры модуля *Decide*

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок-схеме
Type	Тип принятия решения: <i>By Chance</i> – выбор направления основывается на вероятности и <i>By Condition</i> – проверка на выполнение конкретно заданного условия
Percent True	Значение, определяющее процент сущностей, который пойдет по направлению <i>True</i>
If	Тип условия, которое будет проверяться на выполнение
Named	Имя переменной, атрибута или типа сущности, который будет проверяться при входе сущности в модуль
Is	Математический знак условия, например: больше, меньше, равно и т. д.
Value	Значение, с которым будет сравниваться атрибут или переменная пришедшей сущности. Если тип условия – <i>Expression</i> , то в выражении должен стоять знак условия, например <i>Color<> Red</i>

Таблица 37

Параметры модуля *Entity*

Параметры	Описание
Entity Type	Название типа сущности
Initial Picture	Графическое представление сущности в начале имитационного процесса. Это значение может быть впоследствии изменено с помощью модуля <i>Assign</i> . Просмотреть анимационные картинки можно так: <i>Edit/Entity picture</i>

Таблица 38

Параметры модуля *Queue*

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок-схеме
Attribute Name	Имя атрибута, значение которого будет учитываться, если тип = <i>Lowest Attribute Value</i> или <i>Highest Attribute Value</i>
Type	Правило расстановки сущностей в очереди: <ul style="list-style-type: none"> – <i>First in First out</i> – первый вошел, первый вышел; – <i>Last in first out</i> – последний пришел, первый вышел; – <i>Lowest Attribute Value</i> – первый выйдет из очереди тот, значение атрибута у которого низшее; – <i>Highest Attribute Value</i> – первый выйдет из очереди тот, значение атрибута у которого наивысшее

Таблица 39

Параметры модуля *Resource*

Параметры	Описание
1	2
Name	Имя ресурса
Type	Метод, определяющий вместимость ресурса. <i>Fixed Capacity</i> – фиксированная вместимость ресурса. <i>Based on Schedule</i> – вместимость ресурса определяется модулем <i>Schedule</i>

Окончание таблицы 39

1	2
Capacity	Число ресурсов, находящихся в системе
Schedule Name	Имя <i>Schedule</i> модуля, который определяет вместимость ресурса, если <i>Type = Based on Schedule</i>
Busy / Hour	Почасовая стоимость обработки сущности ресурсом. Время учитывается только тогда, когда ресурс занят обработкой, и прекращает учитываться, когда ресурс освобождается
Idle / Hour	Стоимость ресурса, когда он не занят
Per Use	Стоимость обработки ресурсом одной сущности (не зависит от времени)

Таблица 40

Параметры модуля *Schedule*

Параметры	Описание
Name	Название расписания
Type	Тип расписания, который может быть <i>Capacity</i> (расписание для ресурсов), <i>Arrival</i> (для модуля <i>Create</i>) или <i>Other</i> (разнообразные временные задержки или факторы)
Time Units	Масштаб оси времени в графике расписания

Таблица 41

Параметры модуля *Set*

Параметры	Описание
Name	Название группы
Members	Перечисляет ресурсы, входящие в группу. Порядок перечисления ресурсов важен, когда в модуле <i>Process</i> используется правило выбора <i>Cyclical</i> или <i>Preferred Order</i>
Resource Name	Названия ресурсов, входящих в группу

Упражнение 2

Построить имитационную модель на примере обработки однотипных документов. Обработка документов выполняется менеджером компании. Документы поступают каждый час $\pm 0,1$ ч. Время, которое требуется менеджеру на обработку документа, составляет 2 ± 1 ч.

Для этого:

1. Создайте путем перетаскивания с панели *Basic Process* в рабочую область блок-схемы модуль *Create*. Дважды щелкните по созданному объекту, появится окно (рис. 81), в котором задаются свойства модуля *Create*.

2. В окне *Create* установите:

Name – Doc arrive

Type – Expression

Expression – NORM (1, 0.1)

Units – Hours

Entities per Arrival – 1

Max Arrivals – 100

First Creation – 0.0

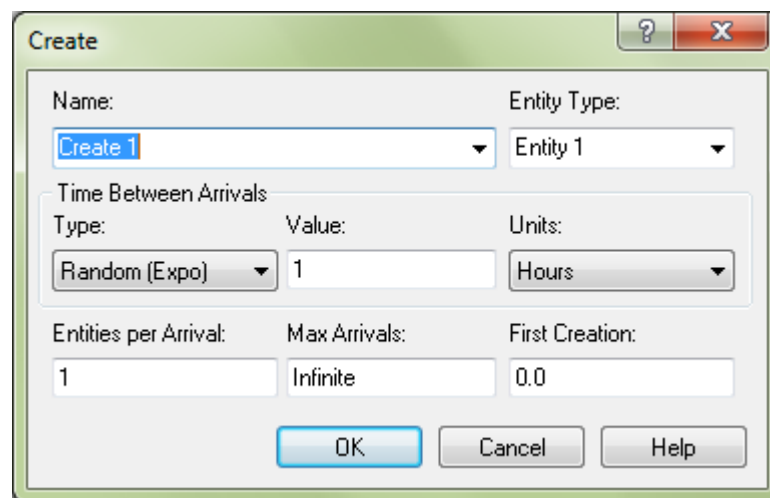


Рис. 81. Окно свойств модуля *Create*

3. Создайте модуль *Process* и дважды щелкните на графическом объекте. Появившееся окно свойств (рис. 82) заполните следующим образом:

Name – Doc Process

Type – Standard

Action – Seize delay Release

Priority – Medium(2)

Minimum – 1

Value (Most Likely) – 2

Maximum – 3

В секции *Resource* нажмите на кнопку Add (Добавить) и в появившееся окно (рис. 83) введите:

Type – Resource


Resource Name – Menager

Quantity – 1

и нажмите OK.

Рис. 82. Окно свойств модуля *Process*

Рис. 83. Окно добавления модуля данных *Resource*

4. Создайте модуль *Dispose 1* и соедините все добавленные модули с помощью инструмента *Connect* (Соединение) , расположенного на стандартной панели инструментов, как показано на рис. 84.

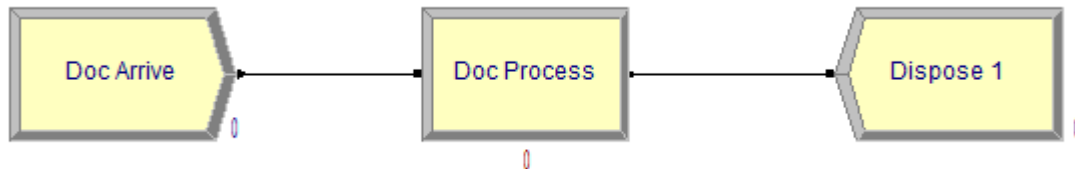



Рис. 84. Графическая схема процесса обработки документов

Упражнение 3

1. Для установки параметров модели выберите в меню *Run* команду *Setup*. На экране появится окно Run Setup (рис. 85), в котором установите следующие параметры:

- на вкладке Run Speed (Скорость выполнения):
Animation Speed Factor (Time Units Per Frame) – 0.01
- на вкладке Replication Parameters (Параметры репликации):
Number of Replications: –1
Warm-up Period – 0
Replication Length – 24
Hours Per Day – 24
Base Time Units – Hours

2. Для запуска модели выберите в меню команду *Run*  *Go*. Из изображения модели (рис.) видно, что очередь к менеджеру (блок Doc Process) растет. Параметры работы менеджера не отвечают интенсивности поступления документов.

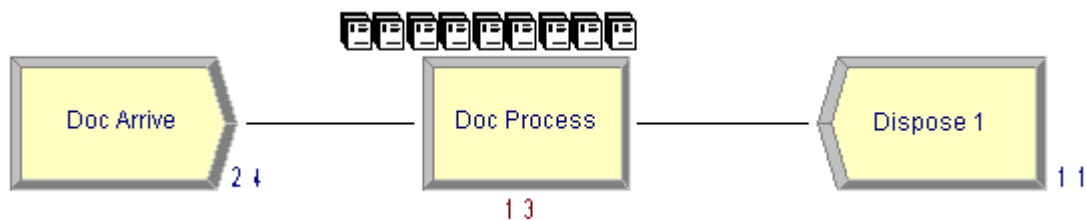


Рис. 85. Окно *Run Setup*

Вопросы для самопроверки

1. Что такое имитационное моделирование? В чем его особенность?
2. Что называется имитационной моделью?
3. Какая основная цель имитационного моделирования?
4. Какие возможности предоставляет пакет *Arena* для проектирования имитационных моделей?
5. Какие в *Arena* основные модули используются для построения имитационной модели?

ПРАКТИЧЕСКАЯ РАБОТА № 10

Тема работы: Моделирование ветвлений в системе *Arena*. Построение модели производственного цеха

1. В сборочный цех поступают детали трех видов. Детали первого типа поступают 20 ± 3 мин, второго типа – 16 ± 5 мин, третьего – 20 мин. Как только сборщику поступают три детали (любые), он производит монтаж готового изделия за 5 мин. Из собранных изделий 15% бракованные. Не бракованные изделия упаковываются по 5 штук за 3 мин упаковщиком.

Смоделировать 8 часовой рабочий день. Собрать статистику по бракованным изделиям.

2. Распределенный банк данных системы сбора информации организован на базе компьютерной сети, состоящей из двух компьютеров. Поступающий запрос обрабатывается на первом компьютере и с вероятностью 50 % необходимая информация обнаруживается на месте. В противном случае необходима посылка запроса во второй компьютер.

Запросы поступают через 10 ± 3 с., первичная обработка запроса занимает 2 с., выдача ответа требует 18 ± 2 с., передача по каналу связи занимает 3 с. Временные характеристики второго компьютера аналогичны первому.

Смоделировать прохождение 400 запросов.

3. В компанию поступают запросы 20 ± 4 мин. Поступающий запрос обрабатывается двумя сотрудниками, причем первый сотрудник обрабатывает 75 % запросов, второй обрабатывает остальные запросы. Первичная обработка запроса занимает 23 мин., выдача ответа требует 18 ± 5 мин., как у первого, так и у второго сотрудника.

Смоделировать прохождение 350 запросов.

4. Участок ремонта кузовов автомобилей состоит из двух рабочих мест: первое рабочее место – это кузовной ремонт автомобиля, второе рабочее место – окраска кузова. После восстановления кузова автомобили поступают в окрасочную камеру.

Время поступления на ремонт поврежденных автомобилей первой модели – случайная величина, равномерно распределенная на интервале от 1 до 6 часов, второй модели – от 1 до 2 часов.

На кузовной ремонт автомобилей первой модели тратится от 1 до 3 часов, второй модели – от 2 до 5 часов.

Время окраски любого автомобиля равномерно распределено на интервале (15 – 20) минут.

Модели первого типа при обслуживании имеют более высокий приоритет.

В случае, если ремонтная мастерская и покрасочная камера заняты, автомобили дожидаются обслуживания в очередях, длины которых не ограничены.

Смоделировать 12 часовый рабочий день.

5. В магазин за покупками приходят клиенты. Для работников магазина клиенты классифицируются на постоянных и обычных. Продавцы (менеджеры) затрачивают в среднем 2 минуты на человека для разъяснения информации по товарам и ответа на вопросы. Приоритетное право на обслуживание без очереди имеют постоянные клиенты. 25 % посетителей уходят без покупок, а остальные встают в очередь в кассу. Кассир один, он обслуживает из очереди постоянных клиентов, а потом обычных посетителей, причем как только кассир рассчитал одного клиента, он сразу же обслуживает следующего, время обслуживания клиента занимает 5 минут.

Постоянные клиенты в основном приходят с утра с 9 до 11 часов и в конце рабочего дня с 16 до 18 часов, а обычные посетители в основном в середине дня.

Построить график, отображающий уровень посещаемости магазина покупателями, и график загруженности кассира.

6. Люди приносят на почту письма, которые могут быть двух видов: заказные и обычные. Затем почтовые работники их обрабатывают.

Заказные письма поступают круглосуточно раз в 5-20 минут, а простые письма принимаются только с 8.00 до 20.00 (8.00-12.00 их количество увеличивается от 50 до 120, наибольшее их количество (260) поступает между 12.00 и 14.00, а затем их количество плавно убывает от 180 до 70).

Оба вида писем обрабатываются одним работником почтовой службы, причем заказные письма обрабатываются вне очереди, т.к. они важнее. Время обработки заказных писем 1-3 минуты, а время обработки простых писем – 3-5, чаще 4 минуты.

Затем все эти письма поступают в отдел подготовки к отправлению, где заказные письма обслуживаются также вне очереди. Время подготовки писем к отправке 10-15 минут.

Создать анимацию работы сотрудников почты и отразить процесс обработки простых писем на гистограмме.

7. Рассматривается работа столовой самообслуживания. Обеды выдают 3 повара. Количество мест за столами всегда достаточно для размещения лиц, уже получивших обед.

Длины временных промежутков между прибытиями посетителей в столовую распределены по равномерному закону на интервале (0 – 20) мин. Время обслуживания на одного посетителя описывается равномерно распределенной величиной на интервале (1 – 2) мин.

Условия работы столовой таковы, что в очереди могут одновременно стоять не более 40 человек. Посетитель стоит в очереди 30 мин., после чего он покидает столовую.

На обед посетитель затрачивает время, которое распределено равномерно на интервале (15 – 25) мин.

В течение 4 часов оценить:

- сколько в среднем требуется посетителю времени на посещение столовой,
- среднее время, которое посетитель проводит в очереди,
- среднее число одновременно обедающих, их максимальное и минимальное число,
- количество посетителей, покинувших столовую.

8. На автозаправочной станции (АЗС) осуществляется заправка автомобилей бензином марок А-80 и АИ-96. На каждый вид топлива работает по две

заправочных колонки, стоящих друг за другом, которые подают бензин со скоростью 1 литр/сек. Автомобили заезжают на АЗС каждые (45 ± 15) сек., причем 2 из 3-х автомобилей заправляется топливом марки АИ-96 и только 1/3 автомобилей топливом А-80. Автолюбители покупают от 5 до 75 литров бензина, причем заказываемый объем кратен 5.

Заказы от клиентов принимает один оператор АЗС, который осуществляет обслуживание одного клиента за (30 ± 15) сек.

Смоделировать 10 часов работы АЗС.

9. Вычислительная система включает три компьютера. В систему в среднем через 30с поступают задания, которые попадают в очередь на обработку к первой машине, где они обрабатываются около 30с. После этого задание поступает одновременно во второй и третий компьютеры.

Второй компьютер может обработать задание за (14 ± 5) с., а третий – за (16 ± 1) с. Окончание обработки задания на любой машине означает снятие ее с решения с той и другой машины. В свободное время второй и третий компьютеры заняты обработкой фоновых задач.

Смоделировать 1 час работы вычислительной системы.

10. Самолеты прибывают для посадки в район крупного аэропорта каждые 10 ± 5 мин. Если взлетно-посадочная полоса свободна, прибывший самолет получает разрешение на посадку. Если полоса занята, самолет выполняет полет по кругу и возвращается к аэропорту через каждые 4 мин. Если после пятого круга самолет не получает разрешения на посадку, он отправляется на запасной аэродром.

В аэропорту через каждые 10 ± 2 мин к взлетно-посадочной полосе вырываются готовые к взлету машины и получают разрешение на взлет, если полоса свободна. Для взлета и посадки самолеты занимают полосу ровно на 2 мин. Если при свободной полосе одновременно один самолет прибывает для посадки, а другой – для взлета, полоса предоставляется взлетающей машине.

Смоделировать работу аэропорта в течение суток. Подсчитать количество самолетов, которые взлетели, сели и были направлены на запасной аэродром. Определить коэффициент загрузки взлетно-посадочной полосы.

Вопросы для самопроверки

1. Что такое имитационное моделирование? В чем его особенность?
2. Что называется имитационной моделью?
3. Какая основная цель имитационного моделирования?
4. Какие возможности предоставляет пакет *Arena* для проектирования имитационных моделей?
5. Какие в *Arena* основные модули используются для построения имитационной модели?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ARIS Express – Free Modelling Software [Электронный ресурс]. – Режим доступа: <http://www.ariscommunity.com/aris-express>. – Загл. с экрана.
2. Грекул, В. И. Проектирование информационных систем [Текст] : учеб. пособие / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. – 2-е изд., испр. – М. : Интернет-университет информ. технол. : БИНОМ. Лаб. знаний , 2012. – 300 с.
3. Евдокимова, С. А. CASE-технологии [Электронный ресурс] : методические указания для самостоятельной работы студентов по направлениям подготовки 38.03.01 – Экономика, 38.03.02 - Менеджмент / С. А. Евдокимова. – Воронеж, 2016. – 16 с. – ЭБС ВГЛТУ.
4. Заботина, Н. Н. Проектирование информационных систем [Электронный ресурс] : рек. УМО по образованию в обл. прикладной информатики в качестве учеб. пособия для студентов высш. учеб. заведений / Н. Н. Заботина. – М. : ИНФРА-М, 2014. – 331 с. – ЭБС "Знаниум".
5. Коваленко, В. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / В. В. Коваленко. – М. : ФОРУМ : ИНФРА-М, 2014. – 320 с. – ЭБС "Знаниум".
6. Официальный сайт CA technologies [Электронный ресурс]. – Режим доступа: <http://www.ca.com/ru/default.aspx>. – Загл. с экрана.

Учебное издание

Евдокимова Светлана Анатольевна

CASE-ТЕХНОЛОГИИ

Лабораторный практикум

Редактор

Подписано в печать _____ Формат 60×90 1/16. Объем

Усл. п.л. Уч.-изд. л. Тираж 100 экз. Заказ _____

ФГБОУ ВО «Воронежский государственный лесотехнический университет
имени Г.Ф. Морозова»

РИО ФГБОУ ВО «ВГЛТУ им. Г.Ф. Морозова». 394087, г. Воронеж, ул. Тимирязева, 8

Отпечатано в УОП ФГБОУ ВО «ВГЛТУ им. Г.Ф. Морозова»

394087, г. Воронеж, ул. Докучаева, 10