

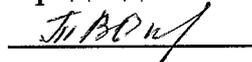
**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ИРКУТСКОЙ ОБЛАСТИ  
«ЧЕРЕМХОВСКИЙ ГОРНОТЕХНИЧЕСКИЙ КОЛЛЕДЖ  
ИМ. М.И. ЩАДОВА»**

Рассмотрено на заседании  
ЦК Информатики и ВТ

« 04 » 06 2020 г.

Протокол № 10

Председатель



Т. В. Окладникова

УТВЕРЖДАЮ

Зам. директора по УР

 Н.А. Шаманова

« 23 » 06 2020г.

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

для выполнения

практических (лабораторных) работ студентов 4курса  
по профессиональному модулю

**ПМ.01 ЭКСПЛУАТАЦИЯ И МОДИФИКАЦИЯ  
ИНФОРМАЦИОННЫХ СИСТЕМ**

**программы подготовки специалистов среднего звена**

09.02.04 Информационные системы (по отраслям)

Разработали

преподаватели:

Папанова О.В.

Чипиштанова Д.В.



2020 г.

## СОДЕРЖАНИЕ

	<b>СТР.</b>
1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
2. ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ РАБОТ	6
3. СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ РАБОТ	8
4. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРАКТИЧЕСКИХ РАБОТ	102
5. ЛИСТ ИЗМЕНЕНИЙ И ДОПОЛНЕНИЙ, ВНЕСЁННЫХ В МЕТОДИЧЕСКИЕ УКАЗАНИЯ	103

## 1.ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические указания по выполнению практических (лабораторных) работ по профессиональному модулю **ПМ.01 Эксплуатация и модификация информационных систем** предназначены для студентов специальности **09.02.04 Информационные системы (по отраслям)**, составлены в соответствии с рабочей программой профессионального модуля **ПМ.01 Эксплуатация и модификация информационных систем** и направлены на достижение следующих целей:

- овладение видом профессиональной деятельности Эксплуатация и модификация информационных систем и соответствующими профессиональными компетенциями;
- формирование у обучающихся умений осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития;
- развитие у обучающихся познавательных интересов, интеллектуальных и творческих способностей при изучении междисциплинарного курса;

Методические указания являются частью учебно-методического комплекса по профессиональному модулю **ПМ.01 Эксплуатация и модификация информационных систем** и содержат задания, указания теоретического минимума, таблицы, схемы. Перед выполнением практической работы каждый студент обязан показать свою готовность к выполнению работы: пройти инструктаж по ТБ при работе за ПК, ответить на вопросы. По окончании работы студент оформляет файл-отчет, защищает работу.

В результате выполнения полного объема практических работ студент должен **уметь:**

- осуществлять сопровождение информационной системы, настройку под конкретного пользователя, согласно технической документации;
- поддерживать документацию в актуальном состоянии;
- идентифицировать технические проблемы, возникающие в процессе эксплуатации системы;
- производить документирование на этапе сопровождения;
- осуществлять сохранение и восстановление базы данных информационной системы;
- составлять планы резервного копирования, определять интервал резервного копирования;
- организовывать разноуровневый доступ пользователей информационной системы в рамках своей компетенции;
- манипулировать данными с использованием языка запросов баз данных, определять ограничения целостности данных;
- выделять жизненные циклы проектирования компьютерных систем;
- использовать методы и критерии оценивания предметной области и методы определения стратегии развития бизнес-процессов организации;
- строить архитектурную схему организации;
- проводить анализ предметной области;
- осуществлять выбор модели построения информационной системы и программных средств;

- оформлять программную и техническую документацию, с использованием стандартов оформления программной документации;
- применять требования нормативных документов к основным видам продукции (услуг) и процессов;
- применять документацию систем качества;
- применять основные правила и документы системы сертификации Российской Федерации.

При проведении практических работ применяются следующие технологии и методы обучения:

1. проблемно-поисковых технологий
2. тестовые технологии
3. информационно-коммуникационные технологии

#### **Правила выполнения практических работ:**

1. Внимательно прослушайте инструктаж по технике безопасности, правила поведения в кабинете информатики.
2. Запомните порядок проведения практических работ, правила их оформления.
3. Изучите теоретические аспекты практической работы
4. Выполните задания практической работы.
5. Оформите отчет в виде файла.

Файл - поименованная совокупность однотипных данных, хранящихся на внешнем носителе под одним именем.

#### **Структура и оформление**

1. Титульный лист;
2. Листинг программы, скриншоты экрана, описание действий (для файла);
3. Перечень основных настроек.
4. Заключение (подводятся итоги, и дается обобщенный вывод ходу реализации программы, даются рекомендации).

#### **Требования к рабочему месту:**

1. Количество ПЭВМ, необходимых для оснащения лаборатории информационных систем, рассчитана на каждого обучающегося с учетом деления класса на две группы.
2. В состав лаборатории информационных систем включена одна машина для преподавателя с соответствующим периферийным оборудованием, диапроектором и экраном.

#### **Критерии оценки:**

**Оценки «5» (отлично)** заслуживает студент, обнаруживший при выполнении заданий всестороннее, систематическое и глубокое знание учебно-программного материала, умения свободно выполнять профессиональные задачи с всесторонним творческим подходом, обнаруживший познания с использованием основной и дополнительной литературы, рекомендованной программой, усвоивший взаимосвязь изучаемых и изученных дисциплин в их значении для приобретаемой специальности, проявивший творческие способности в понимании, изложении и использовании учебно-программного материала, проявивший высокий профессионализм, индивидуальность в решении поставленной перед собой

задачи, проявивший неординарность при выполнении практических заданий.

**Оценки «4» (хорошо)** заслуживает студент, обнаруживший при выполнении заданий полное знание учебно-программного материала, успешно выполняющий профессиональную задачу или проблемную ситуацию, усвоивший основную литературу, рекомендованную в программе, показавший систематический характер знаний, умений и навыков при выполнении теоретических и практических заданий по профессиональному модулю.

**Оценки «3» (удовлетворительно)** заслуживает студент, обнаруживший при выполнении практических и теоретических заданий знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебной и профессиональной деятельности, справляющийся с выполнением заданий, предусмотренных программой, допустивший погрешности в ответе при защите и выполнении теоретических и практических заданий, но обладающий необходимыми знаниями для их устранения под руководством преподавателя, проявивший какую-то долю творчества и индивидуальность в решении поставленных задач.

**Оценки «2» (неудовлетворительно)** заслуживает студент, обнаруживший при выполнении практических и теоретических заданий проблемы в знаниях основного учебного материала, допустивший основные принципиальные ошибки в выполнении задания или ситуативной задачи, которую он желал бы решить или предложить варианты решения, который не проявил творческого подхода, индивидуальности.

В соответствии с учебным планом программы подготовки специалистов среднего звена по специальности **09.02.04 Информационные системы (по отраслям)** и рабочей программой на практические (лабораторные) работы по **ПМ.01 Эксплуатация и модификация ИС** отводится **139 часов**.

## 2. ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ РАБОТ

№п/п	Название практической работы	Количество часов
<b>МДК.01.01 Эксплуатация информационных систем</b>		
Раздел ПМ 1. Инсталляция, настройка и сопровождение информационных систем		
1	Установка управляющего сервера	2
2	Установка информационного сервера	2
3	Установка сервера приложений	2
4	Администрирование управляющего сервера	2
5	Администрирование информационного сервера	2
6	Администрирование сервера приложений	2
7	Настройка параметров операционной системы с помощью консоли управления	2
8	Настройка файлового сервера	2
9	Настройка и эксплуатация веб-сервера	2
10	Настройка и эксплуатация веб-сервера	2
11	Установка, настройка и сопровождение серверов безопасности	2
12	Установка клиентского программного обеспечения	2
13	Установка клиентского программного обеспечения	2
14	Адаптация клиентской части программного обеспечения	2
15	Адаптация клиентской части программного обеспечения	2
16	Создание архивов: обычный, многотомный, самораспаковывающийся, многотомный-самораспаковывающийся	2
17	Использование утилит общего назначения. Стандартная проверка и дефрагментация дисков. Norton Utilites: проверка дисков устройств	2
18	Восстановление системы, работа с реестром	2
Раздел ПМ 2. Управление доступом в информационных системах		
19	Организация доступа к информационной системе	2
20	Обновление и копирование информационной базы	2
21	Планирование и создание учетных записей пользователей информационной системы	2
22	Администрирование групп пользователей информационной системы	2
23	Планирование и установка разрешений для файлов, папок отдельным пользователям и группам	1
24	Наследование разрешений	2
25	Изменение параметров учетных записей пользователей	2
26	Настройка политики учетных записей	2
27	Настройка параметров безопасности информационной системы	2
28	Нахождение ошибок кодирования в модулях	2

	информационной системы	
29	Обновление и восстановление информационной системы	2
30	Анализ систем отслеживания ошибок	2
<b>МДК.01.02 Методы и средства проектирования информационных систем</b>		
Раздел ПМ 3.Проектирование, разработка и модификация информационных систем		
1	Выделение жизненных циклов проектирования информационных систем	2
2-3	Проведение анализа предметной области	4
4	Выбор модели построения информационной системы и программных средств.	2
5-6	Формирование требований к автоматизированной системе	4
7	Моделирование бизнес-процессов средствами BPwin. Анализ среды	2
8-9	Моделирование бизнес-процессов средствами BPwin. Построение модели IDEF0	4
10-11	Моделирование бизнес-процессов средствами BPwin. Диаграммы дерева узлов и FEO. Слияние моделей.	4
12	Моделирование бизнес-процессов средствами BPwin. Создание отчетов	2
13	Моделирование бизнес-процессов средствами BPwin. Диаграммы потоков данных	2
14	Моделирование бизнес-процессов средствами BPwin. Описание процессов (IDEF3)	2
15	Составление технического задания на разработку информационной системы.	2
16-17	Разработка модели фрагмента ИС с применением UML.	4
18	Разработка фрагмента информационной системы.	2
19	Использование языка запросов баз данных для манипулирования данными	2
20	Определение ограничения целостности данных.	2
21	Построение архитектурной схемы организации (модель «как есть»)	2
22	Оформление программной и технической документации с использованием стандартов оформления программной документации	2
23	Использование нормативных документов при разработке информационной системы.	2
24	Тестирование и контрольный расчет программы.	2
25	Составление руководства пользователя к программе.	2

### **3.СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ РАБОТ**

#### **МДК.01.01 Эксплуатация информационных систем**

##### **Практическая работа №1,№2,№3**

##### **Установка управляющего сервера, Установка информационного сервера, Установка сервера приложений**

**Цель работы:** приобретение студентами практических навыков установки операционной системы MicrosoftWindowsServer 2003. Ознакомиться и изучить роли серверов и их назначение.

**Задание:** Ознакомиться и изучить роли серверов и их назначение (файл «ЭИС. Теоретический материал для практических работ»). Представить отчет по проделанной работе.

Контрольные вопросы:

1. Какие роли доступны серверу WindowsServer 2003?
2. Какие службы обычно содержат серверы?
3. Что такое WindowsServer 2003?
4. Укажите основные этапы установки WindowsServer 2003

##### **Практическая работа №4, №5, №6**

##### **Администрирование управляющего сервера**

##### **Администрирование информационного сервера, администрирование сервера приложений**

**Цель работы:** приобретение студентами практических навыков настройки управляющего, информационного сервера и сервера приложений.

**Задание:** Ознакомиться и изучить основы администрирования серверов (файл «ЭИС. Теоретический материал для практических работ»). на практике осуществить настройку ActiveDirectory. Представить отчет по проделанной работе.

Контрольные вопросы:

1. Что такое каталог?
2. Что такое служба каталогов?
3. Какие элементы входят в состав службы ActiveDirectory?
4. Охарактеризуйте понятия «дерево» и «лес». Чем они отличаются?

## Практическая работа 7

### Настройка параметров операционной системы с помощью консоли управления

**Цель работы:** Изучить основные принципы организации и построения консоли администрирования, а также базовые возможности некоторых инструментов системного администратора ОС Windows XP.

#### Теоретические сведения

Консоль управления MicrosoftManagementConsole, сокращенно MMC является инструментом создания, сохранения и открытия средств администрирования (называемых консолями MMC), которые управляют оборудованием, программными и сетевыми компонентами операционной системы (ОС), иными словами, это основа администрирования любой ОС, в частности ОС Windows XP.

В практической работе предполагается ознакомление с основными принципами организации и построения консоли администрирования MMC.

#### Порядок выполнения практической работы

Перед началом выполнения практической работы в среде ОС Windows XP необходимо выполнить следующее:

- 1) загрузить ОС Windows XP и активировать справочное меню (**Пуск | Справка и поддержка**);
- 2) ознакомиться с описанием и возможностями запуска и применения консоли администрирования MMC;
- 3) ознакомиться с возможностью получения сведений пункта 2 из альтернативного источника информации, доступного непосредственно в справке консоли администрирования MMC (**Справка | Вызов справки**);
- 4) ознакомиться с описанием и возможностями оснасток "Локальные пользователи и группы" и "Редактор объекта групповой политики" ("Групповая политика").

**Задание 1.** Создание консоли администрирования MMC в авторском режиме. Для выполнения задания сделайте следующие действия:

- нажмите **Пуск | Выполнить**,
- наберите в появившемся окне **MMC.exe** (или просто **mmc**),
- нажмите **Enter** для ввода.

Возможны следующие альтернативные варианты авторского запуска созданной ранее консоли администрирования:

**A.** запуск из командной строки, используя синтаксис:

**Mmc** *путь\имя\_файла.msc* /a,

где параметр:

*путь\имя\_файла.msc* – запускает консоль MMC с одновременным открытием файла сохраненной консоли с именем *имя\_файла.msc* (**Приложение 1**). Если файл консоли не указан, будет открыта новая консоль MMC.

/a — открывает консоль MMC в авторском режиме.

Дополнительными параметрами команды могут быть:

/64 — открывает 64-разрядную версию консоли MMC (MMC64). Этот параметр используется только при работе в ОС Windows XP 64-Bit Edition.

/32 — открывает 32-разрядную версию консоли MMC (MMC32). При работе в ОС Windows XP 64-Bit Edition в окне консоли MMC, запущенной с этим параметром, открываются 32-разрядные оснастки.

Дополнительная информация по данной команде доступна одноименном разделе справки ОС Windows XP (**Пуск | Справка и поддержка**). Справку также можно получить, набрав в окне командной оболочки строку **Mmc /?** и нажав **Enter** для ввода.

**В.** Запуск из файлового менеджера Проводник ОС Windows XP:

- наведите манипулятор мышь на файл с расширением .msc, находящийся в системной папке ОС (%systemroot%\system32\),
- кликните правой кнопкой мыши на файле и из контекстного меню выберите **Автор**.

**Задание 2.** Изучение возможности изменения параметров, настройки и сохранения консоли администрирования MMC.

Для придания уникального вида сохраненной (новой) консоли администрирования MMC в авторском режиме выполните следующие действия:

1. В меню **Консоль** выберите команду **Параметры**.
2. На вкладке **Консоль** в поле названия введите новый заголовок.
3. На вкладке **Консоль** выполните следующие действия:
  - нажмитекнопку**Сменитьзначок**,
  - в поле **Имя файла** введите путь к файлу, содержащему значки (например, %systemroot%\system32\shell32.dll),
  - в поле **Текущий значок** выберите необходимый значок,
  - кликните**Применить**дляподтверждения.
4. Сохраните окончательно сконфигурированную консоль администрирования MMC, выбрав самостоятельно ее имя и путь к месту расположения в меню **Консоль | Сохранить как...** При сохранении обратите внимание на то, что файлы консоли по умолчанию размещаются в папке «Администрирование», имеющей полный путь C:\DocumentsandSettings\Имя\_пользователя\Главное меню\Программы\Администрирование\.

**Задание № 3.** Добавление различных элементов и компонентов к дереву консоли администрирования MMC.

Дополнительные сведения о добавлении различных элементов в дерево консоли администрирования MMC можно получить, воспользовавшись справкой ОС Windows XP (**Пуск | Справка и поддержка**) в разделе **Общее представление о MMC \ Консоль MMC в авторском режиме \ Оснастки \ Создание консолей**.

Первым необходимым компонентом, добавляемым к дереву консоли администрирования MMC при ее организации и построении, является оснастка. Для добавления оснастки в авторском режиме выполните следующие действия:

1. Откройте **Консоль** управления MMC, созданную ранее.
2. В меню **Консоль** выберите команду **Добавить или удалить оснастку**.
3. В диалоговом окне **Добавить/удалить оснастку** нажмите кнопку **Добавить** вкладки **Изолированная оснастка**. Список **Оснастки** в диалоговом окне **Добавить/удалить оснастку** определяет элемент дерева консоли, к которому

выполняется добавление элементов. В этом списке можно найти любой элемент дерева консоли. Обратите внимание на то, что по умолчанию это **Корень консоли**.

4. В диалоговом окне **Добавить изолированную оснастку**, выберите оснастки **Службы** из списка доступных в системе, кликнув на ней манипулятором мышь и нажав кнопку **Добавить**. Для добавления другой оснастки из списка, повторите указанные действия настоящего пункта повторно.

5. Для некоторых оснасток в процессе их инсталляции выводится диалоговое окно **Выбор целевого компьютера**, определяющее, чем, устанавливаемая оснастка будет управлять в дальнейшем – локальным или сетевым компьютером. Выберите **Локальный компьютер**, установив переключатель в соответствующее положение.

6. Нажмите **Готово**, **Заккрыть** и затем кликните **ОК** для подтверждения ввода.

7. Скройте меню и панель инструментов оснастки **Службы**, выполнив действия указанные ниже:

- В меню **Вид** выберите команду **Настроить**,
- В группе **Оснастка** снимите флажок **Меню**,
- В группе **Оснастка** снимите флажок **Панели инструментов**,
- Нажмите **ОК**.

При устанавливании или снятии флажков, соответствующие им меню и панели инструментов отображаются или скрываются, причем, для всех оснасток консоли, включая текущую. Если переключение флажков не приводит к изменению вида консоли, тогда текущая оснастка не имеет специальных меню или панелей инструментов.

8. Не закрывая консоль администрирования ММС, сохраните ее, выбрав команду **Сохранить** в меню **Консоль**.

Для добавления расширений к уже установленной в предыдущем задании оснастке **Службы** выполните следующее:

9. В меню **Консоль** выберите команду **Добавить или удалить оснастку**.

10. В диалоговом окне **Добавить/удалить оснастку** выберите вкладку **Расширение**. На этой вкладке можно выбрать любой элемент дерева консоли из списка **Оснастки**, которые могут быть расширены, и просмотреть **Доступные расширения**, которые могут быть включены или отключены. После подключения расширение автоматически размещается в дереве консоли под оснасткой, к которой оно относится. Если дерево консоли содержит больше одного экземпляра оснастки, к которой подключено расширение, все остальные экземпляры автоматически получают это расширение.

11. Среди **Доступных расширений** оснастки **Службы** удалите флажок с расширения **Расширенный вид** (предварительно сняв флажок **Добавить все разрешения**) и отметьте, к чему привело это действие. Повторите аналогичные действия с другими расширениями данной оснастки и изучите получаемый результат.

12. Не закрывая консоль администрирования ММС, сохраните ее.

Следующим элементом, необходимым в ряде случаев администрирования и предназначенным, в том числе, для удобства отображения информации, является **новый вид панели задач**. Для добавления видов панелей задач и собственно задач в авторском режиме выполните следующее:

1. Откройте **Консоль** управления ММС, созданную ранее.
2. Добавьте оснастку **Службы** в корень консоли ММС.

3. В дереве консоли кликните манипулятором мышь на этой оснастке.

4. В меню **Действие** или кликнув правой кнопкой манипулятора на оснастке, выберите команду **Новый вид панели задач**.

5. Следуйте инструкциям "**Мастера создания вида панели задач**", чтобы добавить на консоль новую панель вида.

6. Если сразу после создания вида панели задач необходимо создать задачи, установите флажок "**Запустить мастер создания новой задачи**" на последнем экране "**Мастера создания вида панели задач**".

7. Следуйте инструкциям "**Мастера создания новой задачи**", чтобы добавить на консоль новую задачу к существующей панели вида.

8. В дереве консоли кликните элемент или компонент (в нашем случае это оснастка), связанный с видом панели задач, затем в меню **Действие** выберите команду **Правка вида панели задач**.

9. На вкладке **Задачи** нажмите кнопку **Создать**.

10. Повторите инструкции пункта 7 настоящего задания.

11. Не закрывая консоль администрирования ММС, сохраните ее.

Измените вид панели задач сохраненной консоли администрирования ММС, выполнив следующие действия:

- введите новое имя,
- введите новое описание,
- установите переключатель **Стиль для области сведений** в положение, соответствующее новому формату списка,
- удалите соответствующий флажок, чтобы отобразить стандартную вкладку,
- установите переключатель **Стиль для описания задачи** в положение, соответствующее новому стилю задачи,
- выберите новое значение ширины для вертикального списка или высоты для горизонтального списка,
- нажмите кнопку **Параметры** и установите переключатель в одно из необходимых положений,
- нажмите **ОК** для подтверждения ввода,
- изучите полученный результат.

Важной особенностью при построении и организации консоли администрирования ММС является возможность добавления элементов и компонентов дерева консоли в виде списка ярлыков в меню "Избранное". Для добавления элемента или компонента в авторском режиме выполните следующее:

1. Откройте **Консоль** управления ММС, созданную ранее.

2. В дереве консоли кликните элемент или компонент (в нашем случае это оснастка), который нужно добавить в список "Избранное".

3. В области сведений выберите вкладку вида панели задач, которую нужно добавить, в случае, если для элемента или компонента, указанного в дереве консоли, настроен вид панели задач. В противном случае в области сведений вкладки не видны.

4. Выберите в меню **Избранное** команду **Добавить в избранное**.

5. В поле **Создать в:** диалогового окна **Добавление в папку "Избранное"** выполните указанные ниже действия:

– создайте новую папку с названием, выбранным самостоятельно, кликнув папку, которая будет выступать в качестве родительской для создаваемой папки и нажав кнопку **Создать папку**,

– нажмите кнопку **ОК** для ввода,

– в поле **Имя папки** введите имя, под которым будет добавлен элемент,

– кликните **ОК** для подтверждения ввода.

6. Не закрывая консоль администрирования ММС, сохраните ее.

Упорядочите "Избранное" сохраненной консоли администрирования ММС, выполнив следующие действия:

– добавьте новую папку, введя ее имя в соответствующее поле и кликнув **ОК** для подтверждения ввода,

– переместите элемент, созданный в **пункте 5** настоящего задания, в новую, только что созданную, папку и кликните **ОК** для ввода,

– переименуйте выбранный элемент и нажмите клавишу **Enter** для подтверждения ввода,

– удалите все элементы, расположенные ниже папки "Избранное",

– нажмите **Заккрыть** для завершения задания,

– изучите полученный результат,

## Практическая работа № 8, №9, 10

### Настройка файлового сервера. Настройка и эксплуатация веб-сервера

**Цель работы:** получить практические навыки установки и настройки файлового и веб-серверов.

#### Задание 1. Подготовьте файловый сервер.

1. Запустите ОС Windows.

2. Добавьте новую **роль** серверу – *Файл-сервер*:

• откройте диалоговое окно **Управление данным сервером (Пуск/администрирование/Управление Данным Сервером)**;

• активизируйте добавление ролей кнопкой *Добавить или удалить роль*;

• выберите **Файловый сервер** и щелкните *Далее*

3. установите параметры файлового сервера:

• Предоставить доступ UNIX-системам к файлам;

• Предоставить доступ Apple--системам к файлам;

• подтвердите введенные параметры кнопкой *Далее*;

• запустите установку роли сервера кнопкой *Далее*.

4. Перезагрузите виртуальный компьютер кнопкой *Перезагрузить*.

5. Откройте диалоговое окно **Настройки файлового сервера (Пуск/администрирование/Управление Данным Сервером/Управление этим файловым сервером)**.

6. Установите стандартные квоты использования места на диске:

▪ установите флажок *Установить дисковые квоты по умолчанию для новых пользователей данного сервера*;

▪ укажите **размер квот** - 50Мб;

▪ установите **предупреждение о квоте** - 40Мб;

- установите флажок *Не выделять место на диске при превышении дискового пространства*;
  - завершите ввод стандартных квот кнопкой *Далее*.
8. Откажитесь от включения службы индексирования.
  9. Укажите папку на сервере, для хранения файлов, например *C:\Documents and settings\Администратор\Рабочий стол\РUB*.
  10. Далее мастер установки завершит свою работу. Попробуйте теперь зайти на созданную вами сетевую папку с другого компьютера сети. Обратите внимание на способ подключения. Попробуйте заполнить папку для превышения квоты.

## Задание 2. Настройте Web-сервер.

1. Установите **Internet Information Service (IIS) (Пуск/администрирование/Управление приложениями)** **Данным Сервером/Сервер** **IIS):**

о Подготовьте тестовую страницу: создайте временную страницу, вызываемую по умолчанию: наберите в **Блокноте** и сохраните в файле с именем *Default.html* в каталоге *|Inetpub\wwwroot*.

### 2. Настройте Web-сервер:

откройте консоль управления сервером **IIS (Пуск/администрирование/Управление Данным Сервером/Управление этим сервером приложений)**;

перейдите к web-узлу, заданному по умолчанию (*Диспетчер служб IIS/Веб-узлы/Веб-узел по умолчанию*);

▪ откройте диалоговое окно **Свойства узла по умолчанию (контекстное меню/Свойства)**;

▪ добавьте страницу по умолчанию:

- перейдите на вкладку **Документы**;
- установите флажок *Задать страницу содержания по умолчанию*;
- откройте окно добавления кнопкой *Добавить*;
- введите в поле *Default.html*;

▪ подтвердите добавление кнопкой **ОК**. закройте окно свойств кнопкой **ОК**.

### 3. Проверьте настройку Web-сервера:

- на вашем компьютере откройте **Internet Explorer (Пуск/Программы/Internet Explorer)**;
- наберите в адресной строке **http://127.0.0.1/**;
- сделайте скриншот происходящего на экране и сохраните его в своей папке.

## Задание 3. Установите и настройте сервер FTP

### 1. Установите сервер FTP - FileZilla.

### 2. Запустите FileZilla Server Interface.

### 3. Ограничьте количество одновременных подключений к серверу:

- откройте окно настройки сервера (*Edit/Settings*);
- перейдите в раздел **General Settings** (общие настройки);
- введите в поле **Max. number of users** – 2;

### 4. Установите текст приветствия:

перейдите в раздел **Welcome message**;

введите в поле **Custom welcome message** – *Добро пожаловать на мой сервер*;

Установите ограничения по скорости:

- перейдите в раздел **SpeedLimits** (ограничения скорости);
- включите использование правил ограничения скорости радио-
- кнопкой *Use Speed Limit rules*;
- добавьте ограничение по скорости не более 3 Кб/с в понедельник;
- откройте окно задания параметров ограничений кнопкой **Add** (Добавить);
- сбросьте все флажки кроме *Monday* (Понедельник);
- введите в поле **Speed** – 3;
- подтвердите ввод данных кнопкой **OK**; примените параметры кнопкой **OK**.

Создайте группы пользователей **FTP-сервера**:

- откройте диалоговое окно **добавления групп** кнопкой на панели инструментов;
- активируйте добавление групп кнопкой **Add** (Добавить);
- введите **имя группы**, например *Students* (**OK**);
- задайте общую папку для созданной группы:
- перейдите в раздел **SharedFolders** (Общие папки);
- активируйте добавление папок кнопкой **Add** (Добавить);
- укажите общую папку, например *C:\Documents and settings\Администратор\Рабочий стол* и подтвердите выбор кнопкой **OK**;
- разрешите чтение и удаление содержимого общей папки – установите флажок *WriteuDelete*;
- завершите добавление групп пользователей кнопкой **OK**.
- Добавьте нового пользователя:
- откройте диалоговое окно **добавления пользователей** кнопкой на панели инструментов;
- активируйте добавление пользователей кнопкой **Add** (Добавить);
- введите **имя группы**, например *justuser*;
- выберите в списке **Usershouldbememberofthefollowinggroup** созданную ранее группу и подтвердите создание пользователя кнопкой **OK**;
- установите пароль для созданного пользователя:
- перейдите на вкладку **General** (Общие);
- введите в поле **Password** новый пароль, например *123*;
- завершите добавление групп пользователей кнопкой **OK**.
- Проверьте работу сервер:
- запустите командную строку (*Пуск/Программы/Стандартные/Командная строка*);
- введите команду для подключения к FTP-серверу на текущем компьютере: **FTP 127.0.0.1**
- введите имя пользователя - *justuser* (**ENTER**);
- введите пароль - *123* (**ENTER**);
- просмотрите содержимое домашней папки: **DIR**
- отключитесь от сервера: **QUIT**
- закройте командную строку.
- Закройте интерфейс управления **FTP-сервером**.

## Контрольные вопросы

1. Где располагается файл журнала FTP-сервера?

2. Какие варианты ограничения доступа по IP-адресу существуют?
3. В чем отличие виртуального каталога от обычного?
4. Для чего необходимо создавать виртуальный каталог upload, если можно было сделать обычный каталог upload?
5. Можно ли средствами IIS и FTP-сервера ограничить доступ к данным определенному пользователю (по его имени)?
6. Шифруются ли имя пользователя и пароль при аутентификации пользователя на FTP-сервере?
7. В чем преимущество использования FTP-сервера перед использованием обычным папок, к которым предоставлен общий доступ?
8. Если разрешены анонимные подключения к FTP-серверу, то от имени какой учетной записи будет происходить доступ к различным папкам на сервере? Можно управлять данным параметром?

**Итог работы:**

## Практическая работа №11

### Установка, настройка и сопровождение сервера безопасности

**Цель работы:** узнать параметры настройки стандартного брандмауэра, научиться создавать различные типы исключений

#### Теоретические сведения

*Брандмауэр позволяет обеспечить безопасность системы путем ограничения трафика, который поступает от других компьютеров, и предоставления пользователю возможности контролировать локальные данные.* (запишите в отчет)

Чтобы настроить параметры брандмауэра Windows, выполните следующие действия.

1. Выберите в меню **Пуск** пункт **Выполнить**, введите команду **wscui.cpl** и нажмите

кнопку **ОК**.

2. В диалоговом окне **Центр обеспечения безопасности Windows** щелкните ссылку

**Брандмауэр Windows.**

*Запишите в отчет способы запуска брандмауэра.*

Диалоговое окно **Брандмауэр Windows** содержит следующие вкладки.

**Общие Исключения** **Дополнительно**

На вкладке **Общие** расположены следующие элементы управления.

**Включить** (рекомендуется)

**Не разрешать** исключения

**Выключить** (не рекомендуется)

*Запишите в отчет, какие вкладки и элементы управления находятся в настройках брандмауэра.*

Если установлен флажок **Не разрешать исключения**, брандмауэр Windows блокирует все запросы на подключение к компьютеру, включая запросы от программ и служб, перечисленных на вкладке **Исключения**. Кроме того, брандмауэр блокирует совместный

доступ к файлам и принтерам, а также обнаружение сетевых устройств.

Использовать брандмауэр Windows в этом режиме целесообразно при подключении к сетям общего пользования, например сетям в гостиницах и аэропортах. Защита компьютера обеспечивается путем блокирования всех попыток установить к нему подключение.

Брандмауэр Windows в режиме «Не разрешать исключения» позволяет просматривать веб-страницы, обмениваться сообщениями электронной почты или немедленными сообщениями.

На вкладке **Исключения** можно создать исключение *для порта или программы*, чтобы разрешить получение определенных видов входящего трафика. Каждому исключению можно назначить область.

Для домашних и небольших офисных сетей рекомендуется ограничивать такую область пределами локальной сети. В этом случае установленные на компьютере программы доступны другим компьютерам в составе той же подсети, а трафик из удаленных сетей блокируется.

Вкладка **Дополнительно** служит для выполнения следующих действий:

настройка правил для подключений, которые применяются к каждому сетевому интерфейсу;

настройка параметров ведения журнала безопасности;

настройка глобальных правил для трафика по протоколу ICMP (Internet Control Message Protocol), который служит для передачи сведений об ошибках и состоянии;

установка параметров по умолчанию.

### **Настройка исключений**

Брандмауэр Windows содержит несколько предопределенных исключений, которые разрешают сетевой доступ для некоторых общих системных задач, *таких как удаленное управление и совместное использование файлов и принтеров*. В таблице перечислены включенные по умолчанию исключения брандмауэра Windows, а также открытые порты и использующие их программы.

***Запишите в отчет, что такое исключение и какие исключения заданы в работе брандмауэра по умолчанию. Проверьте настройки на соответствующей вкладке. Запишите таблицу в отчет.***

Администратор может также задать собственный набор исключений локально через приложение "Брандмауэр Windows" в панели управления или удаленно с использованием механизма групповых политик. Для этого достаточно просто указать имя программы, которая генерирует трафик или сетевые параметры (номер порта TCP или UDP) и адрес источника, а затем разрешить сформированное исключение в брандмауэре Windows.

Предопределенные исключения предоставляют более гибкие возможности настройки, чем пользовательские, поскольку в них допускается задание нескольких портов в одном правиле. Например, исключение для совместного доступа к файлам и принтерам разрешает использование портов TCP 139, TCP 445, UDP 137 и UDP 138.

**При создании пользовательского исключения система позволяет указать только один порт, так что для открытия диапазона портов необходимо создать набор исключений для каждого порта. При этом допускается создание пользовательских областей, т.е. адресов IP или диапазонов адресов IP, для которых разрешен обмен. Это позволяет задавать наборы портов для предопределенных и для пользовательских исключений.**

Администраторы корпоративных сетей для настройки брандмауэра

Windows могут использовать групповые политики. При этом необходимо указать порт (например, 80), транспорт (TCP или UDP), область, статус (разрешен или запрещен), и имя соединения. Конструкция имеет вид Порт:Транспорт:Область:Статус:Имя (Port:Transport:Scope:Status:Name).

### **Ведение журнала**

Можно настроить брандмауэр Windows для ведения журнала в виде текстового файла на локальном компьютере или сетевом диске. Брандмауэр Windows может записывать события блокирования пакетов и успешных подключений. Журнал содержит необходимые сведения для устранения неисправностей и ошибок при невозможности подключиться к необходимым ресурсам или просмотре разрешенных соединений.

По умолчанию брандмауэр Windows выдает пользователю предупреждение о том, что некоторая программа пытается использовать указанный порт. При централизованной настройке брандмауэра Windows через групповые политики администратор может по своему желанию отключить выдачу предупреждений пользователям.

***Запишите в отчет, как создать файл журнала и указать путь для его хранения. После его создания, просмотрите его содержимое, предварительно осуществив какое-либо сетевое подключение.***

### **Управление брандмауэром Windows из командной строки**

**Программа Netsh** позволяет выполнять настройку клиентов брандмауэра Windows из командной строки напрямую или через пакетный файл. Например, команда ***netsh firewall show config*** показывает текущее состояние брандмауэра Windows на компьютере клиента.

***Выполните команду, запишите команду и ее назначение в отчет.***

Netsh позволяет управлять почти всеми параметрами брандмауэра Windows, так что администраторы, активно использующие в работе сценарии, могут использовать сценарии подключения к домену для настройки и проверки работы брандмауэра Windows.

Например, приведенная ниже команда Netsh создает локальное правило брандмауэра, разрешающее доступ по Telnet к компьютеру, защищенному брандмауэром Windows, с указанных адресов.

```
netsh firewall add portopening protocol = TCP port = 23  
name = Telnet mode = ENABLE scope = CUSTOM addresses =  
192.168.0.0/255.255.255.0, 10.0.0.0/255.255.240.0
```

Эту команду можно было сократить, исключив имена атрибутов:

```
netsh firewall add portopening  
TCP 23 Telnet ENABLE CUSTOM 192.168.0.0/255.255.255.0, 10.0.0.0/255.255.240.0
```

Результат исполнения приведенных выше команд можно проверить с помощью команды netsh firewall show portopening

### **Как работает брандмауэр Windows в режиме повышенной безопасности**

Появился, только начиная с Windows Vista.

Брандмауэр Windows в режиме повышенной безопасности – это брандмауэр, регистрирующий состояние сети, для рабочих станций. В отличие от брандмауэров для маршрутизаторов, которые развертывают на шлюзе между локальной сетью и Интернетом, брандмауэр Windows создан для работы на отдельных компьютерах. Он отслеживает только трафик рабочей станции: трафик, приходящий на IP-адрес данного компьютера, и исходящий трафик самого компьютера. Брандмауэр

Windows в режиме повышенной безопасности выполняет следующие основные операции:

Входящий пакет проверяется и сравнивается со списком разрешенного трафика. Если пакет соответствует одному из значений списка, брандмауэр Windows передает пакет протоколу TCP/IP для дальнейшей обработки. Если пакет не соответствует ни одному из значений списка, брандмауэр Windows блокирует пакет, и в том случае, если включено протоколирование, создает запись в файле журнала.

Список разрешенного трафика формируется двумя путями:

Когда подключение, контролируемое брандмауэром Windows в режиме повышенной безопасности, отправляет пакет, брандмауэр создает значение в списке разрешающее прием ответного трафика. Для соответствующего входящего трафика потребуется дополнительное разрешение.

Когда Вы создаете разрешающее правило брандмауэра Windows в режиме повышенной безопасности, трафик, для которого создано соответствующее правило, будет разрешен на компьютере с работающим брандмауэром Windows. Этот компьютер будет принимать явно разрешенный входящий трафик в режимах работы в качестве сервера, клиентского компьютера или узла одноранговой сети.

Первым шагом по решению проблем, связанных с Брандмауэром Windows, является проверка того, какой профиль является активным. Брандмауэр Windows в режиме повышенной безопасности является приложением, отслеживающим сетевое окружение. Профиль брандмауэра Windows меняется при изменении сетевого окружения. Профиль представляет собой набор настроек и правил, который применяется в зависимости от сетевого окружения и действующих сетевых подключений.

**Брандмауэр различает три типа сетевого окружения: домен, публичная и частная сети.** Доменом является сетевое окружение, в котором подключения проходят аутентификацию на контроллере домена. По умолчанию все другие типы сетевых подключений рассматриваются как публичные сети. При обнаружении нового подключения Windows Vista предлагает пользователю указать, является ли данная сеть частной или публичной. Общий профиль предназначен для использования в общественных местах, например, в аэропортах или кафе. Частный профиль предназначен для использования дома или в офисе, а также в защищенной сети. Чтобы определить сеть как частную, пользователь должен обладать соответствующими административными полномочиями.

Хотя компьютер может быть подключен одновременно к сетям разного типа, активным может быть только один профиль.

### **Итог работы:**

1. Запишите в отчет все, что указано по тексту работы.
2. Ответьте на контрольные вопросы

### **Контрольные вопросы:**

1. Зачем на вкладке «Дополнительно» используется настройка протокола ICMP?
2. В каком случае исключения создаются автоматически (то есть мы не заносим программу вручную в список запрещенных)?
3. Какая информация сохраняется в файле журнала брандмауэра. Чем она может помочь?
4. Какие типы сетевого окружения поддерживает брандмауэр в режиме повышенной безопасности.

## Практическая работа №12, 13

### Установка клиентского программного обеспечения

**1. Цель работы:** изучить этапы установки прикладного программного обеспечения, научиться настраивать и работать с прикладным программным обеспечением.

**Задание 1.** Ознакомьтесь с теоретическими данными по теме (файл «ЭИС. Теоретический материал для практических работ»).

**Напишите этапы установки программного обеспечения.** Для выполнения задания можете воспользоваться ссылкой [http://www.oszone.net/4186\\_2](http://www.oszone.net/4186_2)

**Задание 2.** Напишите клиентские компоненты установки ПО для Windows Professional. Для выполнения задания можете воспользоваться ссылкой [http://www.oszone.net/4186\\_2](http://www.oszone.net/4186_2)

**Задание 3.** Напишите этапы настройки программного обеспечения. Для выполнения задания можете воспользоваться ссылкой [http://www.oszone.net/4186\\_2](http://www.oszone.net/4186_2)

**Задание 4. Сравни, чего больше:**

А) Графических редакторов или прикладных программ?

Б) Антивирусных программ или системных программ?

В) Отладчиков или языков программирования?

**Задание 5. Укажите, какое ПО необходимо людям в следующих ситуациях:**

Ситуация	Системное ПО	Прикладное ПО	Инструментальное ПО
Ландшафтные дизайнеры создают проект нового городского ландшафта			
Профессиональный программист пишет компьютерную программу по заказу крупной фирмы			
Ученые научно-исследовательского института расшифровывают записи, переданные марсоходом			
Выпускной 11 класс готовит фотоальбом и собирает воспоминания о своей школьной жизни			
Web-дизайнер создает сайт известной фирмы			
Школьник играет в компьютерную игру			
Создатели нового мобильного телефона пробуют различные варианты дизайна			
Учитель пишет компьютерный тест по своему предмету			
Конструкторы исследуют модель новой подводной лодки			

**Задание 6.** Напишите в чем принципиальное отличие прикладного программного обеспечения общего назначения от иных видов прикладного программного обеспечения?

**Задание 7.** Напишите в чем заключается принцип организации диалога «компьютер—пользователь» с помощью меню?

**Задание 8.** Ваша работа будет состоять в следующем. У вас в задании приведен алгоритм выполнения практической. Ваша задача, выполняя его, справиться с максимальным количеством заданий.









**ЗАДАНИЕ 1. (перед выполнением ознакомьтесь с материалом по теме - файл «ЭИС. Теоретический материал для практических работ»).**

Удалите с вашего ПК файловый менеджер и установите его, пользуясь имеющимся дистрибутивом.

**ЗАДАНИЕ 2.**Оцените занимаемый объем следующих служб:

1. Службы доступа к файлам и принтерам в сети.
2. Обновление корневых сертификатов.
3. Очереди сообщений.
4. Службы индексирования.
5. Службы факсов.
6. Средства управления и наблюдения.

**ЗАДАНИЕ 3.**Выполните следующие действия:

1.Произведите проверку диска 2-я способами:

- используя интерфейс Windows;
- используя командную строку.

2. Запустите программу дефрагментации дисков. Проанализируйте 2 тома. После проведения анализа отображается диалоговое окно с сообщением о проценте фрагментированных файлов и папок в томе и рекомендацией к действию, сохраните информацию анализа в файл.

3. Убедитесь, что на наименьшем по объему диске не менее 15% свободного пространства и выполните его дефрагментацию.

4. Выполните п.2,3 с помощью командной строки

Введите**defrag**ом[/a] [/v]

Аргумент	Описание
<i>Том</i>	Буква диска или точка подключения тома, который требуется проанализировать.
/a	Анализ тома и вывод краткого отчета об анализе.
/v	Отображение полных отчетов об анализе и дефрагментации

1. Выполните очистку диска от всех временных файлов.
2. Выполните быстрое форматирование диска A:, с заданием метки.
3. Выполните форматирование флэш-карты в файловой системе NTFS.

**Содержание отчета:**

- 1.Номер, тему, цель работы.
2. Протокол работы и скриншоты результатов.
3. Ответы на контрольные вопросы.
4. Вывод о проделанной работе.

**Контрольные вопросы**

1. Каково назначение служебных программ Windows?
2. Как осуществляется доступ к служебным программам Windows?
3. Для чего используется программа проверки диска?
4. Каково назначение программы дефрагментации диска?
5. Что такое дефрагментация диска?
6. Что подразумевается под «очисткой диска»?
7. Что такое форматирование дисков? В каких случаях прибегают к форматированию?

**Практическая работа №18  
Восстановление системы, работа с реестром**



\_\_\_\_\_. Может показаться, что программа и процесс - понятие похожие, они фундаментально отличаются. Программа - \_\_\_\_\_, а процесс - \_\_\_\_\_

Известная утилита для анализа активности процессов-системный TaskManager (\_\_\_\_\_). В ядре Windows нет такого понятия, как задачи, поэтому TaskManager

Диспетчер задач Windows отображает список активных процессов. Его можно запустить различными способами:

В этом окне процессы идентифицируются по имени образа, экземплярами которого они являются. В отличие от других объектов в Windows процессам нельзя присваивать глобальные имена.

Если вкладка Processes окна диспетчера задач показывает список процессов, то содержание вкладки Applications (приложения) отображает список видимых окон верхнего уровня всех объектов "рабочий стол" интерактивного объекта WindowStation. Столбик Status (состояние)

\_\_\_\_\_. Состояние "Running" ("выполняется")

\_\_\_\_\_, aNot Responding ("не отвечает")

Вкладка

Applications

позволяет

Для предотвращения доступа приложений к критически важным данным операционной системы и устранения риска их модификации Windows использует два режима доступа к

процессору (даже если он поддерживает более двух режимов): \_\_\_\_\_ . Код программ работает в \_\_\_\_\_ , тогда как код операционной системы (например, системные сервисы и драйверы устройств) - в \_\_\_\_\_ . В режиме ядра предоставляется доступ ко всей системной памяти и разрешается выполнять любые машинные команды процессора.

Прикладные \_\_\_\_\_ программы

\_\_\_\_\_ . Переключение с пользовательского режима в режим ядра осуществляется \_\_\_\_\_

\_\_\_\_\_ . Операционная система \_\_\_\_\_

\_\_\_\_\_ . Перед возвращением управления пользовательскому потоку процессор переключается обратно в пользовательский режим. Благодаря этому операционная система \_\_\_\_\_

Поэтому ситуация, когда пользовательский поток часть времени работает в пользовательском режиме, а часть - в режиме ядра, обычная. Поскольку подсистема, отвечающая за поддержку графики и окон, функционирует в режиме ядра, то программы, интенсивно работающие с графикой, большую часть времени действуют в режиме ядра, а не в пользовательском режиме. Самый простой способ проверить это - \_\_\_\_\_

## **Задание 2. Диспетчере задач.**

### **Задание 2.1. Просмотр состояния процессов через диспетчер задач.**

#### ***Просмотр и анализ взаимосвязей процессов и потоков.***

Запустите диспетчер задач. Переключитесь на закладку Процессы. В меню Вид выберите пункт Выбрать столбцы... и укажите столбцы согласно рисунку 1. ***Сделайте скриншот вашей работы и сохраните его под именем «Выбор столбцов диспетчера задач»***

Потом определите сколько процессов запущено системой. Для этого я передите на закладку Быстродействие и в меню Вид выберите пункт Загрузка ЦП. (см. рис.2) Например: из рисунка мы видим что системой запущено 53 процесса, загрузка ЦП колеблется от 3-5%, файл подкачки загружен на 931 МБ. ***Сделайте скриншот вашей работы и сохраните его под именем Диспетчер задач. Вкладка "Быстродействие"***

Теперь запускаем приложение Paint и смотрим на изменения ЦП. Например в примере: загрузка ЦП колеблется в пределах 1-4%, файл подкачки загрузился на 942 МБ. Если сравнить объемы используемого файла подкачки и доступной физической памяти, то можно сделать вывод что в сумме они дают общий объем физической памяти.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 3. Результат после запуска приложения Paint». Напишите в каких пределах колеблется загрузка ЦП, как изменился файл подкачки.**

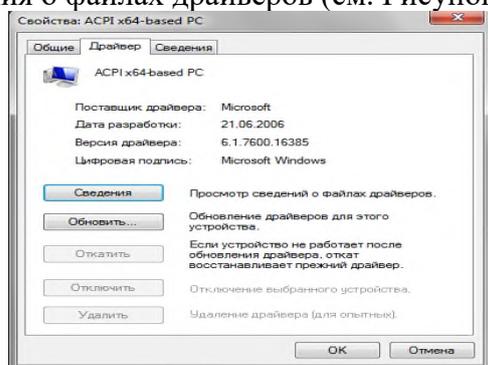
**Просмотр времени работы системы в пользовательском режиме и режиме ядра.**

Переключитесь на закладку Быстродействие и в меню Вид выберите пункт Вывод времени ядра. Быстро прокручиваем страницы любого текстового документа. На индикаторе Загрузка ЦП процент работы процессора в пользовательском режиме отражается зеленым цветом, в режиме ядра красным. Из рисунка четко видны изменения работы процессора при прокрутке страниц.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 4. Состояние системы при прокрутке страниц»**

**Задание 2.2. Просмотр сведений о поддержке многопроцессорных систем.**

Согласно следующему заданию откройте диспетчер устройств (Пуск – Панель управления – Диспетчер устройств). В списке устройств раскройте узел Компьютер. В окне свойств (Правой кнопкой свойства) перейдите на закладку Драйвер и с помощью кнопки Сведения просмотрите сведения о файлах драйверов (см. Рисунок 5).



**Рисунок 5. Сведения о файлах поддержки многопроцессорных систем**

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 5. Сведения о файлах поддержки многопроцессорных систем»**

**Задание 2.3. Изучение отношения процессов родитель-потомок**

1. Запустите командную строку (Пуск – Все программы – Стандартные – Командная строка)

2. Потом наберите startcmd для запуска второго окна командной строки.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 6. Запуск окон командной строки»**

3. Далее я откройте диспетчер задач.

4. Переключился на второе окно командной строки.

5. Ввел mspaint для запуска Microsoft Paint.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 7. Запуск приложения Paint»**

6. Потом выберите второе окно командной строки.

7. Ввел команду exit, тем самым завершив работу второго окна.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 8. Завершение работы второго окна»**

8. Потом переключитесь в диспетчер задач. Откройте его вкладку Приложения. Щелкнув правой кнопкой мыши задачу Командная строка (cmd.exe) и выбрал Перейти к процессу.

**Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 9. Диспетчер задач. Переход к процессу cmd.exe»**

11. Щелкните процесс Cmd.exe, выделенный цветом.

12. Щелкнув правой кнопкой мыши, выберите команду Завершить дерево процессов.

13. В окне Предупреждение диспетчера задач щелкните кнопку Завершить дерево процессов.

Первое окно командной строки исчезнет, но окно Paintпо прежнему работает.

*Сделайте скриншот вашей работы и сохраните его под именем «Рисунок 10. Диспетчер задач. Переход к процессу cmd.exe»*

#### контрольные вопросы

1. Что такое WindowsAPI
2. Что такое процессы
3. Что такое потоки
4. Что такое задачи
5. Что такое диспетчер задач

### Практическая работа №19

#### Организация доступа к информационной системе

##### Цель работы.

- 1) формирование умений реализации доступа к базе данных MySQL;
  - 2) формирование знаний о программных интерфейсах доступа к данным.
- Подключение в СУБД можно организовать различными способами.

**Первый способ. Подключение через ODBC.** Для работы с СУБД MySQL 5 через ODBC необходимо скачать и установить MySQLConnectors (<http://dev.mysql.com/downloads/connector/>). После установки, необходимо настроить драйвер ODBC. Рассмотрим по шагам. Нажать «Пуск \ Панель управления \ Администрирование \ Источники данных (ODBC)».

Далее нужно нажать «Добавить»:

Выберите «MySQL ODBC 5.1 Driver»:

В открывшемся окне нужно прописать настройки БД. DataSourcename – имя подключения к БД. Можно использовать любую последовательность из латинских символов без пробела. В нашем примере это mysql\_bd. Description – описание, носит информативный характер. Server – localhost, порт – 3306. User и Password – пользователь БД, с помощью которого осуществится подключение.

По умолчанию, в примере используется пользователь root, для которого не установлен пароль. Можно создать отдельного пользователя. После ввода имени пользователя и пароля, можно выбрать БД. В выпадающем списке будут указаны все доступные БД. Естественно, и пользователя, и БД нужно было создать заранее.

##### **Второй способ. Подключение через php.**

##### **Создадим3 файла.mysql\_pass.php**

*//Файл будет хранить имя пользователя, пароль и имя базы данных.*

*//Каждое значение нужно написать с новой строки без лишних*

*//пробелов. Файл должен содержать только 4 строки*

```
<?php die; ?>
login
password

data_base_name
```

##### **connect\_mysql.php**

*//Файл, который устанавливает соединение с БД. Скрипт открывает*

*//файл mysql\_pass.php, получает имя, пароль и имя БД и*

```

//устанавливает соединение
<?php $access=array();

$access=file("mysql_pass.php");
$login_db=trim($access[1]);
$passwd_db=trim($access[2]);
$db_db=trim($access[3]);

$db = mysql_connect("localhost", "$login_db", "$passwd_db");
mysql_query("SET CHARACTER SET cp1251");

mysql_query("SET NAMES cp1251");
mysql_query("SET character_set_client='cp1251'"); mysql_query("SET
character_set_results='cp1251'"); mysql_query("SET
collation_connection='cp1251_general_ci'");
mysql_select_db("$db_db", $db);

if(!$db)
{
echo "<p>К сожалению, не доступен сервер MySQL</p>";
exit();
}
if (!mysql_select_db($db_db,$db) )
{
echo "<p>К сожалению, не доступна база данных</p>";exit();
}
$ver = mysql_query("SELECT VERSION()");
if(!$ver)
{
echo "<p>Ошибка в запросе</p>";
exit();
}
}
?>

```

## **index.php**

```

//В любом файле, где нужен доступ к БД нужно подключить файл
//connect_mysql.php, после чего можно исполнять запросы
<?php

Error_reporting(E_ALL & ~E_NOTICE & ~E_WARNING);
include("connect_mysql.php");

?>

```

Все файлы должны находиться в одной директории. В начале и в конце файла не должно быть пустых строк.

## **Пример**

По сценарию работа программы начинается с появления окна авторизации, в котором пользователь вводит свои данные. В таблице ниже перечислены

визуальные объекты, их назначение и первоначальная настройка.

Таблица 13

Объект	Тип/Панель	Назначение	Настройка
<i>Login.php</i>	-	Окно авторизации	Система учета оплаты КЦ
<i>\$login0</i>	-	Переменная логина	Получаем из базы данных
<i>\$pass0</i> <input name="login0">	Поле ввода	Переменная пароля Строка для ввода логина	Получаем из базы Данных Размер 25 символов, размер шрифта 12 (size="25" style="font-size: 12px;)
<inputname="pass0">	TYPE="password"	Строка для ввода пароля	style="font-size:12px;" size="25">
<td><p align="center">Пароль:</td>	-	Поясняющая надпись	Расположить рядом с строкой ввода пароля
<td><p align="center">Логин:</td>	-	Поясняющая надпись	Расположить рядом с строкой ввода логина
<i><b>Пожалуйста, авторизируйтесь в системе</b></i>	-	Поясняющая надпись	Расположить сверху таблицы
<imgsrc="key.jpg"	Image	Элемент дизайна	Расположить в левой части формы.
<INPUT VALUE="Войти">	TYPE="submit"	Кнопка войти для перехода на Главную страницу	Расположить под таблицей

В результате должно получиться визуальными компонентами

Кроме визуальных объектов потребуются специальные компоненты для доступа к базам данных, настройка которых зависит от механизма. Далее в таблице приводятся этапы реализации функционала созданного окна.

Следующим этапом служит создание таблицы users, где хранятся пароли и логины пользователей (два вида – пользователя-консультанта и администратора)

Для защиты пароля используется механизм md5, с помощью которого пароли хранятся в базе данных в зашифрованном виде.

Для преобразования символов пароля в шифр необходимо его пропустить через файл кода:

```
<?php
echo md5("ПАРОЛЬ, КОТОРЫЙ НАДО ШИФРОВАТЬ");
?>
```

Например, пароль «ПАРОЛЬ, КОТОРЫЙ НАДО ШИФРОВАТЬ» в зашифрованном виде выглядит так - 1e954982f6ce8d149ede1483f9858767.

### **Задания**

- Изучить правила доступа к базе данных средствами php.
- Связать пользовательский интерфейс с СУБД MySQL.

## **Практическая работа 20**

### **Обновление и копирование информационной базы**

Цель работы: научиться производить обновление и копирование информационной базы

**Задание (перед выполнением ознакомьтесь с материалом по теме - файл «ЭИС. Теоретический материал для практических работ»).**

Создать свою информационную базу (Для сохранения файлов конфигурации создать папку с именем Группа\_Фамилия, путь сохранения диск D). Ввести информацию о создаваемой конфигурации. Описать перечень подсистем. Создать необходимые справочники и константы. Выполнить проверку работоспособности системы.

### **Ход работы**

#### **1 Создайте свою информационную базу**

Для создания новой конфигурации с «нуля» требуется выполнить следующие действия:

- запустить систему 1С:Предприятие (Пуск-Все программы-1С Предприятие 8.2 - 1С: Предприятие);
- в окне «Запуск 1С: Предприятия» щелкаем по кнопке «Добавить»
- в окне «Добавление информационной базы» устанавливаем переключатель положение «Создание новой информационной базы» и нажимаем кнопку «Далее»;
- в появившемся окне устанавливаем переключатель в строку «Создание информационной базы без конфигурации» и нажимаем кнопку «Далее»;
- в следующем окне указываем название информационной базы и тип расположения и щелкаем «Далее»;
- в следующем окне выбираем каталог (место хранения) информационной базы и язык и щелкаем по кнопке «Далее»;
- в следующем окне указываем параметры запуска:
  - ✓ Вариант аутентификации – «Выбрать автоматический».
  - ✓ Скорость соединения – «Обычная».
  - ✓ Основной режим запуска – «Выбрать автоматически».
  - ✓ Версия 1С: Предприятия – «8.2»
  - ✓ Нажимаем кнопку «Готово»
- после завершения процедуры добавления информационной базы в списке окна запуска появится новая строка с названием нашей информационной базы.

#### **2 Введите информацию о создаваемой конфигурации**

Запустим созданную базу в режиме «Конфигуратор». Откроется окно конфигуратора, в котором надо открыть конфигурацию с помощью меню «Конфигурация - Открыть конфигурацию».

В открывшемся окне в корне дерева конфигурации указано имя «Конфигурация», а в самой структуре данных нет ни одного конкретного справочника, документа и.т.д.

Необходимо дважды щелкнуть по имени «Конфигурация» в корне дерева и в развернувшейся палитре свойств внести следующие данные:

- Имя: НоваяКонфигурация;
- Синоним: Новая конфигурация;
- Основной режим запуска: Обычный;
- Вариант встроенного языка: Русский;
- Основной язык: Русский;
- Краткая информация: Пустая конфигурация;
- Поставщик: внесите свою фамилию.

Теперь закроем палитру свойств, тогда увидим изменения в окне конфигурации (изменится имя).

### **3 Описать перечень подсистем**

Для создания собственного перечня подсистем в окне «Конфигурация» раскроем ветвь «Общие», выделим ветвь «Подсистемы» и щелкнем по кнопке «Добавить». Откроется палитра свойств для подсистемы.

Создадим две подсистемы: Бухгалтерский Учет и Оперативный Учет.

### **4 Создать справочник «Ед.измерения»**

Для создания нового справочника в окне «Конфигурация» надо выделить ветвь «Справочники» и щелкнуть по кнопке «Добавить». Откроется окно конструктора справочника.

На закладке «Основные» заполняем поля имени, синонима и комментария.

Переходим на вкладку «Данные». Длину кода, длину наименования, тип кода и основное представление оставляем без изменений.

Добавим новый реквизит «ПолноеНаименование» и заполним свойства этого реквизита в палитре свойств:

- ✓ Наименование: ПолноеНаименование
- ✓ Длина: 20

На закладке «Нумерация» оставляем все без изменений.

На закладке «Формы» добавить форму списка. В конструкторе форм:

- ✓ выберем тип формы - Форма списка справочника
- ✓ включим опцию - Назначить форму - основной
- ✓ включим опцию - Основная форма списка и выбора
- ✓ имя и синоним оставим без изменения.

Во втором окне конструктора выбираем реквизиты:

- ✓ картинка;
- ✓ код;
- ✓ наименование;
- ✓ полное наименование.

На закладке «Подсистемы» указать, что этот справочник будет участвовать в подсистеме «Оперативный Учет».

Остальные параметры не изменять. Закроем окно конструктора. В дереве конфигуратора появится новая ветвь, соответствующая созданному справочнику.

### **5 Создать справочник «Сотрудники»**

Добавляем новый справочник.

Ввести имя и синоним.

Переходим на вкладку «Данные». Добавим реквизит «Должность» (длина значения 20).

На закладке «Формы» устанавливается режим «В диалоге». Добавить форму списка и выбора. Во втором окне конструктора выбираем реквизиты:

- ✓ Картинка;
- ✓ Код;
- ✓ Наименование;
- ✓ Должность.

Создать форму элемента, вставив реквизиты код, наименование и должность.

На закладке «Подсистемы» указать, что этот справочник будет участвовать в подсистеме «Оперативный Учет».

Остальные параметры не изменять. Закроем окно конструктора. В дереве конфигуратора появится новая ветвь, соответствующая созданному справочнику.

### **6 Создать справочник «Контрагенты»**

Добавляем новый справочник.

Внести имя и синоним.

На закладке «Иерархия» включить опцию «Иерархический справочник» и «Ограничение количества уровней иерархии» = 2.

Переходим на вкладку «Данные». Добавим реквизиты «Адрес» и «Телефон», изменяя длины значений, «ИНН» (длину оставить 10)

На закладке «Формы» устанавливается режим «В диалоге». Добавить форму списка и выбора. Во втором окне конструктора выбираем реквизиты:

- ✓ Картинка;
- ✓ Код;
- ✓ Наименование.

Создать форму элемента, вставив реквизиты:

- ✓ Код;
- ✓ Наименование;
- ✓ Адрес;
- ✓ Телефон;
- ✓ ИНН.

Создать форму группы, вставив реквизиты:

- ✓ Код;
- ✓ Наименование.

Остальные параметры не изменять. Закроем окно конструктора. В дереве конфигуратора появится новая ветвь, соответствующая созданному справочнику.

### **7 Создать справочник «Номенклатура»**

Добавляем новый справочник.

Внести имя и синоним.

На закладке «Иерархия» включить опцию «Иерархический справочник» и «Ограничение количества уровней иерархии» = 2;

Переходим на вкладку «Данные». Добавим реквизиты:

- ✓ «ЕдИзмерения»(в свойствах выбираем тип данных СправочникСсылка.ЕдИзмерения)
- ✓ «ОтпускнаяЦена» (в свойствах выбираем тип данных - число, включить опцию - неотрицательный).

На закладке «Формы» устанавливается режим «В диалоге». Добавить форму списка и выбора.

Во втором окне конструктора выбираем реквизиты:

- ✓ Картинка;
- ✓ Код;
- ✓ Наименование;
- ✓ Ед.измерения;
- ✓ Отпускная цена;

Создать форму выбора и выбрать реквизиты:

- ✓ Картинка;
- ✓ Код
- ✓ Наименование.

Создать форму элемента, вставив реквизиты:

- ✓ Код
- ✓ Наименование
- ✓ Ед.измерения
- ✓ Отпускная цена.

Создать форму группы, вставив реквизиты:

- ✓ Код;
- ✓ Наименование.

Остальные параметры не изменять. Закроем окно конструктора. В дереве конфигуратора появится новая ветвь, соответствующая созданному справочнику.

### **8Создать константы**

В окне «Конфигурация» выделить ветвь «Константы» и «Добавить». Откроется окно свойств константы.

Добавить новые константы.

Для констант «Наименование» и «Адрес» оставить тип данных «Строка», увеличив его длину. А для констант «Руководитель» и «Гл.бухгалтер» - «СправочникСсылка.Сотрудники».

### **Отладка и тестирование созданных объектов конфигурации**

Для выполнения отладки конфигурации на панели инструментов нажмите кнопку «Отладка». При правильно выполненными заданиями система должна выдать запрос на принятие изменений в конфигурации, подтвердите эти изменения. Далее в режиме 1С: Предприятие, проверьте правильность и корректность работы, созданных объектов конфигурации, заполнив их соответствующими данными.

1. Ознакомиться с теоретическим материалом и методическими указаниями
2. Загрузить программу 1С: Предприятие 8.2
3. Создать свою информационную базу
4. Ввести информацию о создаваемой конфигурации
5. Описать перечень подсистем
6. Создать справочник «Ед. измерения»
7. Создать справочник «Сотрудники»
8. Создать справочник «Контрагенты»

9. Создать справочник «Номенклатура»
10. Создать константы
11. Выполнить проверку работоспособности системы
12. Ответить на контрольные вопросы
13. Сделать вывод
14. Оформить отчет по практической работе

### Контрольные вопросы по теме

1. Дайте определение - Общие объекты. Приведите характеристику общих объектов.
2. Опишите режимы работы системы 1С предприятие 8.2.
3. Отличие административных функций «Выгрузить информационную базу..» от «Сохранить конфигурацию в файл» и «Загрузить информационную базу..» от «Загрузить конфигурацию из файла».
4. Дайте определение константы. Какие данные, сохраняются в константах. Порядок работы с константами.
5. Дайте определение справочника. Методы работы со справочниками. Основные справочники, используемые в 1С. Операции над справочниками.

### Практическая работа 21

#### Планирование и создание учетных записей пользователей информационной системы

**Цель работы:** получить практический навык планирования и создания учетных записей информационной системы

#### Теоретические сведения

*ActiveDirectory* является **LDAP** (*LightweightDirectoryAccessProtocol* — «облегченный протокол доступа к каталогам»)–совместимой реализацией **службы каталогов** (это средство иерархического представления ресурсов, принадлежащих некоторой отдельно взятой организации, и информации об этих ресурсах) корпорации **Microsoft** для операционных систем семейства **Windows NT**. *ActiveDirectory* позволяет администраторам использовать глобальные политики, развёртывать программы на множестве компьютеров (через глобальные политики или посредством **MicrosoftSystemsManagementServer 2003**) и устанавливать важные обновления на всех компьютерах в сети (с использованием **WindowsServerUpdateServices (WSUS); SoftwareUpdateServices (SUS)** ранее). *ActiveDirectory* хранит данные и настройки среды в централизованной базе данных. Сети *ActiveDirectory* могут быть различного размера: от нескольких сотен до нескольких миллионов объектов.

Помимо обычных для служб каталогов задач, *ActiveDirectory* способна удовлетворить широкий спектр потребностей по обработке имен, обслуживанию запросов, регистрации, администрированию и устранению конфликтов. В *ActiveDirectory* используется тесно увязанный набор **API** и протоколов, так что она может работать с несколькими пространствами имен, собирать и предоставлять информацию о каталогах и ресурсах, находящихся в удаленных филиалах и под управлением разных ОС. *ActiveDirectory* имеет следующие возможности и характеристики:

- Поддержка открытых стандартов для облегчения межплатформных операций с каталогами, в т. ч. доменной системы имен **DNS** и стандартных протоколов, таких как **LDAP** (*LightweightDirectoryAccessProtocol*— «облегченный протокол доступа к каталогам»).
- Поддержка стандартных форматов имен для простоты миграции и эксплуатации.

- Богатый набор **API**, которые могут использоваться как для командных сценариев, так и в программах на **C/C++**.
- Простой и интуитивно понятный процесс администрирования благодаря несложной иерархической доменной структуре и использованию технологии «перетащить и оставить».
- Возможность расширения набора объектов в каталогах, за счет гибкой логической схемы.
- Быстрый поиск по глобальному каталогу.
- Быстрое и удобное обновление информации посредством многоуровневой (*multimaster*) репликации данных.
- Совместимость с предыдущими версиями ОС **Windows NT**.
- Взаимодействие с сетями **NetWare**.

*ActiveDirectory* позволяет управлять с одного рабочего места всеми заявленными ресурсами (файлами, периферийными устройствами, базами данных, подключениями к серверам, доступом к Web, пользователями, другими объектами, сервисами и т. д.). В качестве идентификационной службы в ней используется доменная система имен (**DNS**), применяемая в Интернете, объекты в доменах строятся в иерархию организационных единиц (**OE**), а домены могут быть объединены в древовидную структуру. Администрирование становится еще проще, так как в *ActiveDirectory* отсутствует понятие главного контроллера доменов (**ГКД**) и резервного контроллера доменов (**РКД**). В *ActiveDirectory* существуют только контроллеры домена (**КД**), и все они равны между собой. Администратор может сделать изменения на любом **КД**, и эти изменения будут скопированы на всех остальных **КД**.

*ActiveDirectory* отделяет логическую структуру иерархии доменов **Windows 2003** от физической структуры сети. *Объекты*: ресурсы хранятся в виде объектов. *Схема ActiveDirectory*:

- Домены: базовая организационная структура.
- Деревья: несколько доменов объединяются в иерархическую структуру.
- Леса: группа из нескольких деревьев домена.

*Организационные единицы*: позволяют делить домен на зоны и делегировать на них права.

Логическая структура *ActiveDirectory* не базируется на физическом местонахождении серверов или сетевых соединениях в пределах домена. Это позволяет структурировать домены, отталкиваясь не от требований физической сети, а от административных и организационных требований.

Объекты хранятся в *ActiveDirectory* в виде иерархической структуры контейнеров и подконтейнеров, упрощающей поиск, доступ и управление, она во многом похожа на файловую систему **Windows** с файлами в папках.

*Классы объектов*. Объект на самом деле представляет собой просто набор атрибутов. Например, объект пользователя (*userobject*) состоит из таких атрибутов, как имя, пароль, телефонный номер, сведения о членстве в группах и т. д. Атрибуты, образующие объект, определяются классом объекта.

Классы и атрибуты, определяемые ими, собирательно называются *ActiveDirectorySchema* — в терминологии баз данных схема (*schema*) — это структура таблиц и полей, а также их взаимосвязи. *ActiveDirectorySchema* можно считать набором данных (классов объектов), определяющим то, как организована и хранится реальная информация (атрибуты объекта) в каталоге.

*ActiveDirectory* — не первая служба каталогов. В современных сетях используется несколько служб каталогов и стандартов. Вот лишь некоторые из них:

- **X.500** и **Directory Access Protocol (DAP)**. **X.500** — спецификация **InternetStandardsOrganization (ISO)**, определяющая, как должны быть структурированы глобальные каталоги. **X.500** также описывает применение **DAP** для обеспечения взаимодействия между клиентами и серверами каталогов;
- **Lightweight Directory Access Protocol (LDAP)**. **LDAP** была разработана в ответ на критические замечания по **DAP**, которая оказалась слишком сложной для применения в

большинстве случаев. **LDAP** быстро стала стандартным протоколом каталогов в Интернете.

- **NovellDirectoryServices (NDS)**. Служба каталогов для сетей **NovellNetWare**, совместимая со стандартом **X.500**.
- **ActiveDirectory**. Составная часть сетей под управлением **WindowsServer 2000** или **WindowsServer 2003**. Соответствует стандарту **LDAP**.

### Задание 1. Установите ActiveDirectory.

1. Подготовьте виртуальную машину **VM-2** к установке службы каталогов:
  - подключите к ВМ образ установочного диска **win2003-1.iso**;
2. Откройте диалоговое окно **Управление данным сервером (Пуск/Администрирование/Управление Данным Сервером)**.
3. Проверьте наличие установленного сервера доменных имен (**DNS**). *Установка службы каталогов невозможна без DNS.*
4. Активизируйте добавление новых ролей для сервера (*Добавить или удалить роль*).
5. Выберите пункт **Контроллер домена (ActiveDirectory)**. Перейдите к следующему шагу кнопкой *Далее*.
6. Ознакомьтесь с информацией об устанавливаемых компонентах и щелкните *Далее*.
7. Ознакомьтесь с информацией **Мастера установки ActiveDirectory** и щелкните *Далее*.
8. Ознакомьтесь с информацией о совместимости с операционной системой и щелкните *Далее*.
9. Укажите **вариант создания контроллера домена** - *Контроллер домена в новом домене* и щелкните *Далее*.
10. Укажите **тип создаваемого домена** - *Новый домен в новом лесу* и щелкните *Далее*.
11. Введите **полное DNS-имя создаваемого домена** - *example.edu.ru* и щелкните *Далее*.
12. Введите **NetBIOS-имя домена** - *EXAMPLE* и щелкните *Далее*.  
*Обычно оно уже указано по умолчанию, исходя из того, какое DNS-имя было дано создаваемому домену. Например, при создании домена с именем example.edu.ru, по умолчанию NETBIOS-имя будет EXAMPLE.*
13. Укажите место хранения баз данных и журналов *ActiveDirectory*:
  - введите в поле **Папка расположения Баз** - *C:\WINDOWS\NTDS\BASE*;
  - введите в поле **Папка расположения журнала** - *C:\WINDOWS\JOURNAL*;
  - подтвердите изменения кнопкой *Далее*.
14. Ознакомьтесь с информацией об общей папке и щелкните *Далее*.  
*Если ваш DNS-сервер настроен неправильно, вы получите сообщение об ошибке и возможных путях её устранения (выберите Проблема будет решена позже).*
15. Установите разрешения для объектов службы каталогов - *Разрешения, совместимые только с Windows 2000 или WindowsServer 2003* и щелкните *Далее*.
16. Укажите пароль, для учётной записи администратора режима восстановления:
  - введите в поле **Пароль режима восстановления** – *123456*;
  - введите в поле **Подтверждение** – *123456*;
  - подтвердите изменения кнопкой *Далее*.
17. Ознакомьтесь с сводной информацией об установке службы каталогов и запустите ее установку кнопкой *Далее*.
18. Завершите работу мастера кнопкой *Готово*.
19. Перезагрузите виртуальную машину кнопкой *Перезагрузить сейчас*.
20. Войдите в систему после перезагрузки и завершите установку службы каталогов кнопкой *Готово*.

### Задание 2. Работа с Active Directory Manager

1. Откройте диалоговое окно **Пользователи и компьютеры (Пуск/Администрирование/ActiveDirectory – пользователи и компьютеры)**.
2. Ознакомьтесь с структурой созданного ранее домена:
  - разверните узел **example.edu.ru**;
  - просмотрите стандартных пользователей домена (**Builtin**);
  - просмотрите контроллеры домена (**DomainControllers**).
3. Создайте новый каталог (подразделение/контейнер) в корне сервера:
  - активизируйте узел **example.edu.ru**;
  - щелкните по кнопке **Создание нового подразделения в текущем контейнере**  на панели инструментов;
  - в появившемся диалоговом окне введите имя создаваемого подразделения - **Students (ОК)**;

*Будет создан новый контейнер для подразделения и выделение автоматически переместиться на него.*

4. Создайте новую учетную запись пользователя в контейнере **Students**:
  - откройте диалоговое окно **Новый объект – Пользователь**, кнопкой **Создание нового пользователя в текущем контейнере**  ;
  - введите данные о пользователе:
    - **Полное имя пользователя** – *Просто Пользователь*;
    - **Имя входа пользователя (логин)** – *JustUser*;
    - Подтвердите введенные данные кнопкой *Далее*.
  - установите пароль для пользователя:
    - введите в поле **Пароль** – *User1234*;
    - введите в поле **Подтверждение** – *User1234*;
    - установите флажок *Срок действия пароля не ограничен*;
    - завершите ввод пароля кнопкой *Далее*.
5. Введите более полную информацию о пользователе:
  - откройте диалоговое окно **свойств пользователя** (двойной щелчок по надписи *Просто пользователь*);
  - введите в поле **Описание** – *это тестовый пользователь*;
  - введите в поле **Комната** - *316* (номер кабинета, в котором проходит занятие);
  - введите в поле **Телефон** – *<номер\_телефона>*;
  - укажите *адресные данные* на вкладке **Адрес**;
  - укажите несколько дополнительных телефонов пользователя на вкладке **Телефоны**;
  - завершите изменение данных пользователя кнопкой **ОК**.
6. Создайте группу **group1** в контейнере **Students**:
  - откройте диалоговое окно **создания групп (контекстное меню/Создать/Группа)**;
  - введите **имя группы** – *group1*;
  - завершите создание группы кнопкой **ОК**.
7. Задайте дополнительную информацию для группы **group1**:
  - откройте диалоговое окно **свойств группы** (двойной щелчок по надписи **group1**);
  - введите в поле **Описание** – *Это тестовая группа*;
  - завершите изменение данных группы кнопкой **ОК**.
8. Включите созданного ранее пользователя **Просто пользователь (JustUser)** в группу **group1**:
  - откройте диалоговое окно **свойств пользователя** (двойной щелчок по записи пользователя);
  - перейдите на вкладку **Член групп**;
  - откройте диалоговое окно выбора группы кнопкой *Добавить*;

- введите **название группы** – *group1*;
  - завершите добавления пользователя в группу кнопкой **ОК**.
  - закройте диалоговое окно свойств пользователя кнопкой **ОК**.
9. Выполните редактирование политики безопасности домена, созданную автоматически:
- откройте диалоговое окно свойств домена **example.edu.ru** (*контекстное меню/Свойства*);
  - перейдите на вкладку **Групповая политика**:  
*В списке будет расположена политика домена по умолчанию DefaultDomainPolicy.*
  - откройте диалоговое окно (**Редактор объектов групповой политики**) изменения политики **DefaultDomainPolicy** (двойной щелчок по политике);
  - внесите в изменения в политику паролей:
    - перейдите в раздел **Политика паролей** (*Конфигурация компьютера/Конфигурация Windows/Параметры безопасности/Политики учётных записей/ Политики паролей*);
    - установите минимальную длину пароля:
      - откройте окно **изменения параметров пароля** (двойной щелчок по надписи *Мин. длина пароля*);
      - введите в поле **Длина пароля не менее** – 5;
      - подтвердите изменения кнопкой **ОК**;
    - отключите соответствие пароля требованиям сложности:
      - откройте диалоговое окно **свойств требования сложности** (двойной щелчок по надписи *Пароль должен отвечать требованиям сложности*);
      - установите радиокнопку *Отключить*;
      - подтвердите изменения кнопкой **ОК**;
  - закройте **Редактор объектов групповой политики** (*Консоль/Выход*);
  - закройте диалоговое окно свойств домена кнопкой **ОК**.
10. Выполните выход из системы с повторным входом для активации изменений в политике безопасности. *Изменения в политике паролей вступят в силу только после выхода из системы и повторного входа в неё.*
11. Измените пароль созданного ранее пользователя:
- активизируйте раздел **Students**;
  - задайте новый, более простой пароль пользователю **Просто пользователь**:
    - откройте диалоговое окно **задания пароль** (*контекстное меню/Смена пароля*);
    - введите в поле **Пароль** – 123;
    - введите в пол **Подтверждение пароля** – 123;
12. Исключите созданного ранее пользователя из группы **group1**:
- откройте диалоговое окно **свойств группы**;
  - перейдите на вкладку **Члены группы**;
  - выделите в списке удаляемого пользователя и щелкните по кнопке **Удалить**;
  - подтвердите удаление кнопкой **Да**;
  - закройте диалоговое окно свойств группы кнопкой **ОК**.
13. Включите созданного ранее пользователя в администраторы домена:
- откройте диалоговое окно **свойств пользователя Просто пользователь**;
  - перейдите на вкладку **Член групп** и щелкните **Добавить**;
  - введите в поле **Администраторы домена**;
  - завершите добавление в группу кнопкой **ОК**;
  - закройте окно свойств пользователя кнопкой **ОК**.

### Задание 3. Присоединение компьютеров под управлением Windows XP к домену.

1. Запустите виртуальную машину **VM-3** и загрузите в ней ОС **Windows XP**.
2. Откройте диалоговое окно **Свойства системы** (*Пуск/Панель управления/Система*) и перейдите на вкладку **Имя компьютера** (*Далее*).
3. Откройте диалоговое окно **Изменение имени** кнопкой **Изменить**.
4. Укажите в разделе **Является членом** – *домена*.
5. Введите в поле левую часть имени созданного ранее домена, например **example**.
6. Подтвердите изменения кнопкой **ОК**. *Через некоторое время домен к которому подключается компьютер запросит имя и пароль администратора домена.*
7. Введите **имя/пароль** администратора в созданном ранее домене (*JustUser/123*).
8. Ознакомьтесь с сообщением от домена *«Добро пожаловать в домен»* и щелкните **ОК**.
9. Ознакомьтесь с информацией о необходимости перезагрузки компьютера и закройте окно кнопкой **ОК**.
10. Закройте диалоговое окно **Свойства системы** кнопкой **ОК**.
11. В появившемся диалоговом окне согласитесь с перезагрузкой кнопкой **Да**. *После этого компьютер начнет перезагружаться.*
12. Войдите в систему с использованием любой из созданных вами учетных записей.
13. Проверьте действие сделанных вами ограничений в домене.
14. Выключите ВМ **VM-1**.

### Задание 4. Присоединение компьютеров под управлением OpenSUSELinux к домену.

1. Запустите виртуальную машину **VM-1** и загрузите в ней ОС **OpenSUSE**.
2. Откройте диалоговое окно **Настройки системы** (*Главное Меню/Компьютер/YaST Администратор*).
3. В появившемся диалоговом окне введите **пароль администратора системы** – *1233456* (задавался при установке системы).
4. Перейдите в раздел **Сетевые службы**.
5. Запустите **Мастер присоединения к домену**
6. В появившемся диалоговом окне подтвердите установку дополнительного пакета **samba-client** кнопкой **Продолжить**.
7. Введите в поле **Домен или группа** имя созданного ранее домена (левую часть полного имени домена) – *example*.
8. Включите разрешение создания общих ресурсов пользователями.
9. Щелкните по кнопке **Завершить**.
10. Появится сообщение с предложением присоединить компьютер к домену, щелкните по кнопке **Да**.
11. Укажите данные администратора домена:
  - введите в поле **Имя пользователя** – *JustUser*;
  - введите в поле **Пароль** – *123*;
  - подтвердите ввод данный кнопкой **Да**.
12. Проверьте правильность подключения к домену:
  - выйдите из системы;
  - войдите в систему под одной из учетных записей пользователей домена.
13. Завершите работу ВМ.

### Контрольные вопросы

1. Что такое ActiveDirectory?
2. Возможности ActiveDirectory.

3. Схема ActiveDirectory.
4. Классы объектов.

## Практическая работа №22

### Администрирование групп пользователей информационной системы

**Цель работы:** изучить процессы управления пользователями: создание пользователей, модификация свойств

#### Теоретические сведения

*Учетные записи (accounts)* пользователей, компьютеров и групп — один из главных элементов управления доступом к сетевым ресурсам, а значит, и всей системы безопасности сети в целом. В среде Windows 2003 ActiveDirectory существует 3 главных типа пользовательских учетных записей:

- **Локальные учетные записи пользователей.** Эти учетные записи существуют в локальной базе данных SAM (*SecurityAccountsManager*) на каждой системе, работающей под управлением Windows 2003. Эти учетные записи создаются с использованием инструмента *LocalUsersandGroups* (*Локальные пользователи и группы*) консоли *ComputerManagement* (*Управление компьютером*). Заметим, что для входа в систему по локальной учетной записи, эта учетная запись обязательно должна присутствовать в базе данных SAM на системе, в которую вы пытаетесь войти. Это делает локальные учетные записи непрактичными для больших сетей, вследствие больших накладных расходов по их администрированию.
- **Учетные записи пользователей домена.** Эти учетные записи хранятся в ActiveDirectory и могут использоваться для входа в систему и доступа к ресурсам по всему лесу AD. Учетные записи этого типа создаются централизованно при помощи консоли "*ActiveDirectoryUsersandComputers*" ("*ActiveDirectory – пользователи и компьютеры*").
- **Встроенные учетные записи.** Эти учетные записи создаются самой системой и не могут быть удалены. По умолчанию любая система, будь то изолированная (отдельно стоящая) или входящая в домен, создает две учетные записи – *Administrator* (*Администратор*) и *Guest* (*Гость*). По умолчанию учетная запись Гость отключена.

Сосредоточим свое внимание на учетных записях пользователей домена. Эти учетные записи хранятся на контроллерах домена, хранящих копию базы данных ActiveDirectory.

Существуют различные форматы, в которых могут быть представлены имена для входа пользователей в систему, потому что они могут отличаться для целей совместимости с клиентами, работающими под управлением более ранних версий Windows (такими как 95, 98, NT). Два основных вида имен входа — это с использованием суффикса *UserPrincipalName* (*основного имени пользователя*) и имя входа пользователя в системах пред-Windows 2000.

Основное имя пользователя (*UPN, UserPrincipleName*) имеет такой же формат, как и электронный адрес. Он включает в себя имя входа пользователя, затем значок "@" и имя домена. По умолчанию доменное имя корневого домена выделено в выпадающем окне меню, независимо от того, в каком домене учетная запись была создана (выпадающий список будет также содержать имя домена, в котором вы создали эту учетную запись).

Также можно создавать дополнительные доменные суффиксы (та часть имени, которая стоит после знака @), которые будут появляться в выпадающем списке и могут быть использованы при образовании UPN, если вы их выберете (это делается при помощи консоли "*ActiveDirectory – домены и доверие*" ("*ActiveDirectoryDomainandTrusts*").

Существует только одно обязательное условие при этом — все UPN в лесу должны быть уникальными (т.е. не повторяться). Если учетная запись входа пользователя использует UPN для входа в систему Windows 2003, вам необходимо только указать UPN и пароль — более нет нужды помнить и указывать доменное имя. Другое преимущество данной системы именования состоит в том, что UPN часто соответствует электронному адресу пользователя, что опять уменьшает количество информации о пользователе, которую необходимо запоминать.

#### Локальные учетные записи

Каждый компьютер с операционными системами Windows NT/2000/XP/2003 (если это не сервер, являющийся контроллером домена) имеет локальную базу данных учетных записей, называемую базой данных SAM. Эти БД обсуждались при описании модели безопасности "Рабочая группа". Локальные пользователи и особенно группы используются при назначении прав доступа к ресурсам конкретного компьютера даже в доменной модели безопасности. Общие правила использования локальных и доменных групп для управления доступом будут описаны ниже.

### Управление доменными учетными записями пользователей

Доменные учетные записи пользователей (а также компьютеров и групп) хранятся в специальных контейнерах AD. Это могут быть либо стандартные контейнеры *Users* для пользователей и *Computers* для компьютеров, либо созданное администратором Организационное подразделение (ОП). Исключение составляют учетные записи контроллеров домена, они всегда хранятся в ОП с названием *DomainControllers*.

Рассмотрим на примерах процесс создания учетных записей пользователей в БД ActiveDirectory и разберем основные свойства доменных учетных записей. Учетные записи для компьютеров создаются в процессе включения компьютера в домен.

#### Создание доменной учетной записи

1. Откроем административную консоль "ActiveDirectory – пользователи и компьютеры".
2. Щелкнем правой кнопкой мыши на контейнере, в котором будем создавать учетную запись, выберем в меню команду "Создать" и далее — "Пользователь".
3. Заполним поля "Имя", "Фамилия", например, "Иван" и "Иванов" (в английской версии — *FirstName, LastName*), поле "Полное имя" (*FullName*) заполнится само.
4. Введем "Имя входа пользователя" (*Userlogonname*), например, *User1*. К этому имени автоматически приписывается часть вида "@<имя домена>", в нашем примере — "@world.ru" (полученное имя должно быть уникальным в масштабах леса).
5. В процессе формирования имени входа автоматически заполняется "Имя входа пользователя (пред-Windows 2000)" (*Userlogonname (pre- Windows 2000)*), создаваемое для совместимости с прежними версиями Windows (данное имя должно быть уникально в масштабе домена). В каждой организации должны быть разработаны схемы именования пользователей (по имени, фамилии, инициалам, должности, подразделению и т.д.) В нашем примере получится имя "WORLD\User1". Нажмем кнопку "Далее".
6. Вводим пароль пользователя (два раза, для подтверждения).
7. Укажем начальные требования к паролю:
  - Требуется смену пароля при следующем входе в систему (полезно в случае, когда администратор назначает пользователю начальный пароль, а затем пользователь сам выбирает пароль, известный только ему);
  - Запретить смену пароля пользователем (полезно и даже необходимо для учетных записей различных системных служб);
  - Срок действия пароля не ограничен (тоже используется для паролей учетных записей служб, чтобы политики домена не повлияли на функционирование этих служб, данный параметр имеет более высокий приоритет по сравнению с политиками безопасности);
  - Отключить учетную запись.

Нажмем кнопку "Далее";

8. Получаем итоговую сводку для создаваемого объекта и нажимаем кнопку "Готово".

Правила выбора символов для создания пароля:

- длина пароля — не менее 7 символов;
- пароль не должен совпадать с именем пользователя для входа в систему, а также с его обычным именем, фамилией, именами его родственников, друзей и т.д.;
- пароль не должен состоять из какого-либо слова (чтобы исключить возможность подбора пароля по словарю);
- пароль не должен совпадать с номером телефона пользователя (обычного или мобильного), номером его автомобиля, паспорта, водительского удостоверения или другого документа;

- пароль должен быть комбинацией букв в верхнем и нижнем регистрах, цифр и спецсимволов (типа @#\$\$%^\* &()\_+ и т.д.).

И еще одно правило безопасности — регулярная смена пароля (частота смены зависит от требований безопасности в каждой конкретной компании или организации). В доменах Windows существует политика, определяющая срок действия паролей пользователей.

### Обзор свойств учетных записей пользователей

Свойства учетной записи пользователя содержат большой набор различных параметров, размещенных на нескольких закладках при просмотре в консоли " *ActiveDirectory – пользователи и компьютеры* ", причем при установке различных программных продуктов набор свойств может расширяться.

Рассмотрим наиболее важные с точки зрения администрирования свойства.

Откроем консоль " *ActiveDirectory – пользователи и компьютеры* " и посмотрим свойства только что созданного нами пользователя.

Закладка " *Общие* ". На данной закладке содержатся в основном справочные данные, которые могут быть очень полезны при поиске пользователей в лесу AD. Наиболее интересные из них:

- " *Имя* "
- " *Фамилия* "
- " *Выводимое имя* "
- " *Описание* "
- " *Номер телефона* "
- " *Электронная почта* "

Закладка " *Адрес* " — справочная информация для поиска в AD.

Закладка " *Учетная запись* " — очень важный набор параметров (параметры " *Имя входа пользователя* " и " *Имя входа пользователя (пред-Windows 2000)* " обсуждались выше при создании пользователя):

- кнопка " *Время входа* " — дни и часы, когда пользователь может войти в домен;
- кнопка " *Вход на...* " — список компьютеров, с которых пользователь может входить в систему (регистрироваться в домене);
- Поле типа чек-бокс " *Заблокировать учетную запись* " — этот параметр недоступен, пока учетная запись не заблокируется после определенного политиками некоторого количества неудачных попыток входа в систему (попытки с неверным паролем), служит для защиты от взлома пароля чужой учетной записи методом перебора вариантов; если будет сделано определенное количество неудачных попыток, то учетная запись пользователя автоматически заблокируется, поле станет доступным и в нем будет установлена галочка, снять которую администратор может вручную, либо она снимется автоматически после интервала, заданного политиками паролей;
- " *Параметры учетной записи* " (первые три параметра обсуждались выше):
  - " *Требовать смену пароля при следующем входе в систему* "
  - " *Запретить смену пароля пользователем* "
  - " *Срок действия пароля не ограничен* "
  - " *Отключить учетную запись* " — принудительное отключение учетной записи (пользователь не сможет войти в домен);
  - " *Для интерактивного входа в сеть нужна смарт-карта* " — вход в домен будет осуществляться не при помощи пароля, а при помощи смарт-карты (для этого на компьютере пользователя должно быть устройство для считывания смарт-карт, смарт-карты должны содержать сертификаты, созданные Центром выдачи сертификатов);
- " *Срок действия учетной записи* " — устанавливает дату, с которой данная учетная запись не будет действовать при регистрации в домене (этот параметр целесообразно задавать для сотрудников, принятых на временную работу, людей, приехавших в компанию в командировку, студентов, проходящих практику в организации и т.д.)

Закладки " *Телефоны* ", " *Организация* " — справочная информация о пользователе для поиска в AD.

## Закладка " Профиль "

*Профиль ( profile )* — это настройки рабочей среды пользователя. Профиль содержит: настройки рабочего стола (цвет, разрешение экрана, фоновый рисунок), настройки просмотра папок компьютера, настройки обозревателя Интернета и других программ (например, размещение папок для программ семейства MicrosoftOffice). Профиль автоматически создается для каждого пользователя при первом входе на компьютер. Различают следующие виды профилей:

- *локальные* — хранятся в папке " DocumentsandSettings " на том разделе диска, где установлена операционная система;
- *перемещаемые* (сетевые, или *roaming* ) — хранятся на сервере в папке общего доступа, загружаются в сеанс пользователя на любом компьютере, с которого пользователь вошел (зарегистрировался) в домен, давая возможность пользователю иметь одинаковую рабочую среду на любом компьютере (путь к папке с профилем указывается на данной закладке в виде адреса \\server\share\%username%, где server — имя сервера, share — имя папки общего доступа, %username% — имя папки с профилем; использование переменной среды системы Windows с названием %username% позволяет задавать имя папки с профилем, совпадающее с именем пользователя);
- *обязательные ( mandatory )* — настройки данного типа профиля пользователь может изменить только в текущем сеансе работы в Windows, при выходе из системы изменения не сохраняются.

Параметр "Сценарий входа" определяет исполняемый файл, который при входе пользователя в систему загружается на компьютер и исполняется. Исполняемым файлом может быть пакетный файл ( .bat, .cmd ), исполняемая программа (.exe, .com), файл сценария (.vbs, js).

Закладка " Член групп " — позволяет управлять списком групп, в которые входит данный пользователь.

## Закладка " Входящие звонки ".

Управление доступом пользователя в корпоративную систему через средства удаленного доступа системы WindowsServer (например, через модем или VPN-соединение). В *смешанном режиме* домена Windows доступны только варианты " Разрешить доступ " и " Запретить доступ ", а также параметры обратного дозвона (" Ответный вызов сервера "). В режимах " Windows 2000 основной " и " Windows 2003 " доступом можно управлять с помощью политик сервера удаленного доступа (не надо путать с групповыми политиками). Подробнее данный вопрос обсуждается в разделе, посвященном средствам удаленного доступа.

Закладки " Профиль служб терминалов ", " Среда ", " Сеансы ", " Удаленное управление " — данные закладки управляют параметрами работы пользователя на сервере терминалов:

- управление разрешением пользователя работать на сервере терминалов;
- размещение профиля при работе в терминальной сессии,
- настройка среды пользователя в терминальной сессии (запуск определенной программы или режим рабочего стола, подключение локальных дисков и принтеров пользователя в терминальную сессию);
- управление сеансом пользователя на сервере терминалов (длительность сессии, тайм-аут бездействия сессии, параметры повторного подключения к отключенной сессии);
- разрешение администратору подключаться к терминальной сессии пользователя.

## 1Создание пользователей, модификация свойств пользователей

1. Откройте консоль "ActiveDirectory -пользователи и компьютеры"
2. Создайте учетные записи пользователей User1-N, User2-N
3. Изучите свойства учетных записей

## Повышение уровня функционирования домена

Выполните повышение уровня функционирования домена:

1. Откройте консоль "ActiveDirectory -пользователи и компьютеры"
2. Щелкните правой кнопкой мыши на имени домена
3. Выберите пункт "Изменение режима работы домена"
4. Проверьте текущий режим работы домена - должен быть "Смешанный режим"
5. " Выберите режим работы домена " - выберите режим работы "Windows 2000 (основной режим)"
6. Нажмите кнопку "Изменить"

## КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Учетные записи (accounts) пользователей, компьютеров и групп.
2. Локальные учетные записи пользователей.
3. Учетные записи пользователей домена.
4. Встроенные учетные записи.
5. Обзор свойств учетных записей пользователей

## Практическая работа №23, №24

### Планирование и установка разрешений для файлов, папок отдельным пользователям и группам Наследование разрешений

**Цель:** научиться использовать общие папки для предоставления доступа к сетевым ресурсам; ознакомиться, как разрешения влияют на вид доступа к общим папкам; получить навыки по планированию: какие разрешения доступа к общей папке назначать пользователям и группам для папок данных и папок программ. получить навыки практического применения по созданию и модифицированию общих папок, а также научиться подключаться к общей папке. научиться комбинировать разрешения доступа к общей папке и разрешения NTFS.

(перед выполнением заданий ознакомьтесь с материалом по теме - файл «ЭИС. Теоретический материал для практических работ»).

#### Назначение разрешений

*Пусть пользователю User101 назначены разрешения для получения доступа к ресурсам как отдельному пользователю и как члену группы. Определите, какие результирующие разрешения будут у User101 в следующих ситуациях. (см задание к лабораторной работе)*

**задание 1:**

#### 1. Назначение разрешений для групп пользователей

User101 — член групп Group1, Group2 и Group3. Для папки ПапкаА у Group1 есть разрешение Read (Чтение), у Group3 — FullControl (Полный доступ), а для Group2 разрешений не назначено. Какими результирующими разрешениями будет обладать User101 для ПапкиА?

User101 также является членом группы Sales, которой назначено разрешение Read для ПапкаВ. Для User101 как отдельного пользователя, отменено разрешение Full Control для ПапкаВ. Какие результирующие разрешения будет иметь User101 для ПапкаВ?

1. управление доступом к общим папкам

*Определите результирующие разрешения пользователей, спланируйте совместное использование папок и разрешений доступа к ним, назначьте разрешения доступа к папке, подключитесь к ней, закройте к ней доступ и проверьте эффекты от сочетания разрешений доступа к общей папке и разрешений NTFS.*

### **задание: сочетание разрешений**

Общие папки на томе NTFS содержат вложенные папки, которым назначены разрешения NTFS(см задания к упражнению). Определите результирующие разрешения пользователей в каждом случае.

### **Задание 1: определите результирующие разрешения пользователей**

1. Открыт доступ к папке **Data**. Группа **Sales** имеет для нее разрешение read (Чтение), а для вложенной в нее папки **Sales** — NTFS-разрешение FullControl (Полный доступ).

Каким будет результирующее разрешение группы **Sales** для доступа к папке **Sales** при подключении по сети к папке **Data**?

2. Папка **Users** (Пользователи) содержит личные папки пользователей. Каждая личная папка содержит данные, доступные только пользователю, именем которого она названа. Папка **Users** доступна группе **Users** с разрешением FullControl (Полный доступ). **User1** и **User2** имеют разрешения NTFS FullControl только для своих личных папок: никаких разрешений NTFS для остальных. Эти пользователи — члены группы **Users**.

Какими разрешениями доступа к папке **User1** будет обладать **User1** при подключении к общей папке **Users**? Какими будут его разрешения для папки **User2**?

### **Упражнение 2: планирование общих папок**

*Спланируйте доступ к ресурсам на серверах главного офиса предприятия •(см задание к упражнению). Выполнив упражнение, занесите ваши решения в таблицу. Сделайте ресурсы на этих серверах доступными пользователям ЛВС компании, определив, к каким папкам открыть доступ и какие разрешения назначить группам, в том числе встроенным.*

Ваши решения должны удовлетворять следующим критериям:

Членам группы **Managers** надо читать и вносить исправления в документы в папке **Management**. Больше никто не должен иметь доступ к этой папке.

- Администраторы должны иметь полный доступ ко всем общим папкам, кроме папки **Management**.
- Отделу по работе с клиентами требуется отдельное место в сети для хранения рабочих файлов. Все сотрудники этого отдела — члены группы **Customer Service**.
- Всем сотрудникам компании требуется место в сети для обмена информацией.
- Всем сотрудникам нужны такие приложения, как электронные таблицы, базы данных и текстовые процессоры.
- Только члены группы **Managers** должны иметь доступ к ПО управления проектами предприятия.
- Члены группы **CustomerDBFull** должны читать и вносить информацию в БД клиентов.
- Члены группы **CustomerDBRead** должны только читать информацию из БД клиентов.

Каждый пользователь сети должен иметь собственное место в сети для хранения файлов, доступное только ему.

Имена общих ресурсов должны быть доступны с компьютеров

Windows 2000/NT/98/95, а также с альтернативных платформ.

Запишите свои ответы в следующую таблицу.

Имя папки	Общее имя	Группы и разрешения
<i>Например:</i> Management	Mgmt	Managers: Полный доступ

### Упражнение 3: открытие доступа к папкам

#### Задание 1: откройте доступ к папке

1. Зарегистрируйтесь как (Администратор).
2. Запустите (Проводник), создайте папку **C:\Dostup** щелкните ее правой кнопкой и выберите в контекстном меню команду (Свойства).
3. В диалоговом окне свойств папки **Dostup** перейдите на вкладку (Доступ).
4. Щелкните переключатель ShareThisFolder (Открыть общий доступ к этой папке), щелкните ОК.

WindowsExplorer изменил значок папки **Dostup**, добавив снизу изображение руки, значит, папка стала общей.

#### Упражнение 4: назначение разрешений доступа к общей папке

*Определите текущие разрешения доступа к общей папке и назначьте разрешения группам в вашем домене.*

#### Задание 1: определите текущие разрешения для общей папки Dostup

1. В окне Проводник щелкните правой кнопкой мыши папку C:\Dostup и выберите в контекстном меню команду Свойства.
2. Перейдите на вкладку Доступ и щелкните кнопку Permissions (Разрешения).

По умолчанию группе Everyone (Все) для этой папки назначено разрешение (Полный доступ).

#### Задание 2: аннулируйте разрешения для группы

Убедитесь, что выбрана группа Everyone

1. Щелкните кнопку Remove (Удалить).

#### Задание 3: назначьте разрешение FullControl группе Administrators

Щелкните кнопку Add (Добавить). Откроется диалоговое окно (Выбор: Пользователи, Компьютеры или Группы).

1. Убедитесь, что имя вашего компьютера (Comp1) выведено в поле (Искать в). В поле Name (Имя) щелкните Administrators (Администраторы), а затем — кнопку Add.
2. Группа Administrators добавится в список групп, имеющих разрешения.
3. Какой вид доступа будет назначен этой группе по умолчанию?
4. В столбце Allow (Разрешить) окна Permissions (Разрешения) пометьте флажок FullControl.

Почему также включилось разрешение Change (Изменение)?

4. Щелкните ОК, чтобы закрыть окно Разрешение для папки **Dostup**.
5. Щелкните ОК, чтобы закрыть окно свойств папки **Dostup**.
6. Закройте (Проводник).

**Упражнение 5: прекращение совместного использования папки**  
*Закройте доступ к папке.*

**Задание 1: закройте доступ к папке**

1. Войдите в систему на Comp1 как администратор и запустите Проводник.
2. Щелкните правой кнопкой папку C:\Dostup и выберите в контекстном меню команду (Свойства).
3. В диалоговом окне свойств папки перейдите на вкладку (Доступ).

Щелкните переключатель (Не открывать общий доступ к этой папке), затем — ОК. Под **Dostup** больше нет «руки», означавшей, что папка была общей. Возможно, вам сначала понадобится обновить экран — в этом случае нажмите клавишу F5.

4. Закройте (Проводник).

**Упражнение 6: назначение разрешений NTFS и открытие доступа к папкам**

Назначьте разрешения NTFS папкам **Dostup**, **Public** и **Manuals** и откройте к ним доступ.

Чтобы назначить разрешения NTFS, создайте с помощью Проводника папки и назначьте им разрешения NTFS согласно таблице. Не допускайте наследования разрешений для вложенных объектов и снимите все ранее существовавшие разрешения NTFS.

*Папки переданы в совместное пользование, назначены разрешения доступа к ним. Откройте доступ к папкам и назначьте разрешения пользователям сети согласно таблице. Снимите все остальные разрешения сетевого доступа.*

Путь	Группа	Разрешения NTFS
C:\Dostup	Администраторы Users(Пользователи)	Полный доступ Чтение и выполнение
C:\Dostup\ Manuals	Администраторы Users	Полный доступ Чтение и выполнение
C:\Dostup\ Public	Администраторы Users	Полный доступ Полный доступ

**Контрольные вопросы**

1. Общая папка расположена на томе FAT, и пользователь имеет для нее разрешение FullControl (Полный доступ). К каким объектам в папке он получит доступ?
2. Перечислите разрешения доступа к общей папке.
3. Какие разрешения назначаются общей папке по умолчанию?
4. Общая папка расположена на томе NTFS, и пользователь имеет для нее разрешение FullControl. К каким объектам в этой папке получит доступ пользователь?
5. Почему рекомендуется централизованно хранить общие папки данных?
6. Как лучше обеспечить безопасность общих файлов и папок на NTFS?

**Практическая работа №25**

**Изменение параметров учетных записей пользователей**

**Цель работы:** Научиться создавать учетные записи пользователей, локальных групп, блокировать учетные записи пользователей и т.д.

**Задание**

**А. Создание учетной записи.**

1. Откройте оснастку **Настройка – Панель управления – Администрирование Управление компьютером**

2. В оснастке **Локальные пользователи и группы** установите указатель мыши на папку **Пользователи** и нажмите правую кнопку.
  3. В появившемся контекстном меню выберите команду **Новый пользователь** Появиться окно диалога **Новый пользователь**
  4. В поле **Пользователь** введите имя создаваемого пользователя, например, свою фамилию.
- Примечание:** Имя пользователя должно быть уникальным для компьютера. Оно может содержать до 20 символов верхнего и нижнего регистра. Ниже приведены символы, применение которых в имени пользователя недопустимо: \*[];=.,<>? Имя пользователя не может состоять целиком из точек и пробелов.
5. В поле **Полное имя** введите полное имя создаваемого пользователя.
  6. В поле **Описание** введите описание создаваемого пользователя или учетной записи, например, «студент .....».
  7. В поле **Пароль** введите пароль пользователя и в поле **Подтверждение** подтвердите его правильность вторичным вводом.

Примечание: Длина пароля не может превышать 14 символов.

8. Установите или снимите флажки:
  - Потребовать смену пароля при следующем входе в систему
  - Запретить смену пароля пользователем
  - Срок действия пароля не ограничен
  - Отключить учетную запись
9. Чтобы создать ещё одного пользователя, нажмите кнопку **Создать** и повторите шаги с 1 по 8. Для завершения работы нажмите кнопку **Создать** и затем **Заккрыть**.

#### **В. Создание локальной группы.**

1. В окне оснастки **Локальные пользователи и группы** установите указатель мыши на папке **Группы** и нажмите правую кнопку.
2. В появившемся контекстном меню выберите команду **Новая группа**.
3. В поле **Имя группы** введите имя новой группы, например, **Студенты**.

**Примечание:** Имя локальной группы должно быть уникальным. Оно сможет содержать до 256 символов в верхнем и нижнем регистрах.

4. В поле **Описание** введите описание новой группы.
5. В поле **Члены группы** можно сразу добавить пользователей и группы, которые войдут в новую группу: для этого нужно нажать кнопку **Добавить** и выбрать их в списке. Для завершения нажмите кнопку **Создать** и затем **Заккрыть**.

#### **С. Изменение членства в локальной группе.**

1. В окне оснастки **Локальные пользователи и группы** щелкните на папке **Группы**.
2. В правом подокне установите указатель мыши на модифицируемую группу и нажмите правую кнопку.
3. В появившемся контекстном меню выберите команду **Добавить** в группу или Свойства.
4. Для того чтобы добавить новые учетные записи в группу, нажмите кнопку **Добавить**
5. Далее следуйте указателям окна диалога **Выбор: Пользователи или группы**.
6. Для того, чтобы удалить из группы некоторых пользователей, в поле **Члены группы** окна свойств группы выберите одну или несколько учетных записей и нажмите кнопку **Удалить**.

**Примечание:** В локальную группу можно добавлять как локальных пользователей, созданных на компьютере, так и пользователей и локальные группы, созданные в домене, к которому принадлежит компьютер, или в доверяемых доменах. Встроенные группы не могут быть удалены. Удаленные группы не могут быть восстановлены. Удаление группы не отражается на входящих в неё пользователей.

#### **D. Временная блокировка учетной записи.**

1. Откройте оснастку **Управление компьютером**.
2. Для этого либо выберите на **Рабочем столе** ярлык **Мой компьютер** и нажмите правую кнопку мыши, после этого выберите пункт контекстного меню **Управление**, либо воспользуйтесь разделом **Администрирование в Панели управления**.
3. В открывшейся оснастке выберите пункты **Службные программы/Локальные пользователи и группы** .

4. Откройте папку пользователи и выберите учетную запись **Гость**.
5. Нажмите правую кнопку мыши и выберите пункт свойства.
6. В открывшемся окне снимите отметку пункта **Отключить** учетную запись.
7. Нажмите кнопку **ОК** и сделайте вывод о состоянии учетной записи.
8. Выполните пункт 5 и отметьте пункт **Отключить учетную запись**.

**Задание для самостоятельной работы:**

1. Создайте учетную запись с именем ПЗ-6, используя команду PrintScreen клавиатуры, сохраните копию экрана со списком пользователей Вашего компьютера(для чего после нажатия клавиши PrintScreen вставьте скопированное изображение в новый документ Word) для представления в качестве отчета.
2. Создайте группу **Информационная безопасность** и, как в первом задании, сохраните окно со списком групп Вашего компьютера для отчета.
3. Заблокируйте учетную запись ПЗ-6 и после этого удалитею

**Контрольные вопросы:**

1. Какие методы управления доступом Вам известны?
2. Чем отличается мандатное управление доступом от дискретного?
3. Допустимо ли имя пользователя ПЗ8/44? Почему?

### **Практическая работа №26** **Настройка политики учетных записей**

**Цель работы:** Ознакомиться с механизмами аутентификации и идентификации, локальными политиками безопасности, встроенными в ОС WindowsXP.

**Задание:** Настроить параметры локальной политики безопасности операционной системы WindowsXP.

**Алгоритм выполнения работы:**

Для просмотра и изменения параметров аутентификации пользователей выполните следующие действия:

1. Выберите кнопку Пуск на панели задач.
2. Откройте меню Настроить – Панель управления.
3. В открывшемся окне выберите ярлык Администрирование – Локальная политика безопасности.
4. Выберите пункт **Политика учетных записей** (этот пункт включает два подпункта: **Политика паролей** и **Политика блокировки учетной записи**).
5. Откройте подпункт Политика паролей. В правом окне появится список настраиваемых параметров
6. В показанном примере политика паролей соответствует исходному состоянию системы безопасности после установки операционной системы, при этом ни один из параметров не настроен. Возможные значения параметров приведены в таблице №1.

Таблица №1

Значения параметров Политики паролей

Параметр	Значение
Требовать повторяемости паролей	Определяет число новых паролей, которые должны быть сопоставлены учетной записи пользователя, прежде чем можно будет снова использовать старый пароль. Это значение должно принадлежать диапазону от 0 до 24.
Максимальный срок действия пароля	Определяет период времени (в Днях), в течение которого можно использовать пароль, чем система потребует от пользователя заменить его. Можно задать значение в диапазоне от 1 до 999 дней или снять всякие ограничения срока действия, установив число дней равным 0.
Минимальный срок действия пароля.	Определяет период времени (в Днях), в течение которого можно использовать пароль, чем система потребует от пользователя заменить его. Можно задать значение в диапазоне от 1 до 999 дней или снять всякие ограничения срока действия, установив число дней равным 0.

Минимальная длина пароля.	Определяет наименьшее число символов, которые может содержать пароль учетной записи пользователя. Можно задать значение в диапазоне от 1 до 14 символов или отменить использование пароля, установив число символов равным 0
Пароль должен отвечать требованиям сложности	Определяет, должны ли отвечать пароли требованиям сложности. Если эта политика включена, пароли должны удовлетворять следующим минимальным требованиям. <ul style="list-style-type: none"> <li>➤ Пароль не может содержать имя учетной записи пользователя или какую-либо его часть;</li> <li>➤ Пароль должен состоять не менее чем из 6 символов;</li> <li>➤ В пароле должны присутствовать символы трех категорий из числа следующих четырех: <ul style="list-style-type: none"> <li>1. Прописные буквы английского алфавита от А до Z;</li> <li>2. Строчные буквы английского алфавита от А до Z;</li> </ul> </li> </ul>
Параметр	Значение
	3. Символы не принадлежащие алфавитно-цифровому набору (например, !, \$, #, %). Проверка соблюдения этих требований выполняется при изменении или создании паролей.
Хранить пароли всех пользователей в домене, используя обратимое шифрование.	Определяет, следует ли в системах Windows 2000, WindowsXP хранить пароли, используя обратимое шифрование. Эта политика обеспечивает поддержку приложений, использующих протоколы, которым для проверки подлинности нужно знать пароль пользователя. Хранить пароли, зашифрованные обратимыми методами, это всё равно, что хранить их открытым текстом. Поэтому данную политику следует использовать лишь в исключительных случаях, если потребности приложения оказываются важнее, чем защита пароля.

7. Ознакомьтесь со свойствами всех параметров.
8. Для изменения требуемого параметра выделите его и вызовите его свойства из контекстного меню после нажатия правой кнопки мыши (или дважды щёлкните на изменяемом параметре).
9. В результате этого действия появится одно из окон.
10. Измените, значение параметра и нажмите Ок.
11. Например (обязательно выполнить и сохранить), выберите параметр Требуется неповторяемости паролей и измените его значение на 1.
12. Для настройки Политики блокировки учетной записи выберите этот подпункт и откройте его.
13. Значения параметров данного подпункта Политики учетной записи приведены в таблице №2.

Таблица №2

Параметр	Значение
Пороговое значение блокировки	Определяет число неудачных попыток входа в систему, после которых учетная запись пользователя блокируется. Блокированную учетную запись нельзя использовать до тех пор, пока не будет сброшена администратором или пока не истечёт её интервал блокировки. Можно задать значение в диапазоне от 1 до 999 или запретить блокировку данной учетной записи, установив значение 0.
Блокировка учетной записи на	Определяет число минут, в течении которых учетная запись остаётся заблокированной, прежде чем будет автоматически разблокирована. Этот параметр может принимать значения от 1 до 99999 минут. Если установить
Параметр	Значение
	Значение 0, учетная запись будет заблокирована на всё время до тех пор, пока администратор не разблокирует её явным образом. Если пороговое значение блокировки определено, данный интервал блокировки должен быть больше или равен интервалу сброса.
Сброс	Определяет число минут, которые должны пройти после неудачной

счетчика блокировки через	попытки входа в систему, прежде чем счетчик неудачных попыток будет сброшен в 0. Этот параметр может принимать значения от 1 до 99999 минут. Если определено пороговое значение блокировки, данный интервал сброса не должен быть больше интервала Блокировка учетной записи на.
---------------------------	--

14. Ознакомьтесь со свойствами всех параметров.
15. Для изменения параметров воспользуйтесь алгоритмом, описанным в пунктах 8-10.

#### **Задания для самостоятельной работы:**

1. Измените параметр «**Пароль должен отвечать требованиям сложности**» Политики паролей на «**Включен**» (рисунок 3) и после этого попробуйте изменить пароль своей учетной записи. Зафиксируйте все сообщения системы, проанализируйте и введите допустимый пароль. Этот пароль является результатом выполнения Вашего задания.
2. После успешного выполнения первого задания, измените пароль Вашей учетной записи, а в качестве нового пароля укажите прежний пароль. Все сообщения зафиксируйте, проанализируйте и объясните поведение системы безопасности.
3. Проведите эксперименты с другими параметрами Политики учетных записей.

#### **Контрольные вопросы:**

1. Что такое аутентификация и идентификация?
2. Для чего применяются эти механизмы?
3. Что можно настроить с помощью вкладки **Локальные политики безопасности**?

### **Практическая работа №27**

#### **Настройка параметров безопасности информационной системы**

**Цель работы:** Научиться работать с редактором шаблонов безопасности, создавать их, редактировать и сохранять.

**Задание:** Загрузить редактор Шаблона безопасности, редактировать шаблон безопасности и сохранить его под новым именем.

#### **Алгоритм выполнения работы.**

##### **А. Загрузка оснастки Шаблона безопасности.**

1. Выберите кнопку Пуск панели задач.
2. Перейдите к пункту Выполнить.
3. В открывшемся поле ввода введите команду mms.
4. В результате откроется консоль управления mms.
5. В меню Консоль выберите команду Добавить или удалить оснастку (рис. 1) и нажмите кнопку Добавить.
6. В поле Оснастка дважды щелкните Шаблоны безопасности.
7. Нажмите кнопку Закрывать.
8. В списке выбранных оснасток для новой консоли появится элемент Шаблоны безопасности.
9. Если на консоль не нужно добавлять другие оснастки, нажмите кнопку ОК.
10. Чтобы сохранить эту консоль, в меню консоль выберите команду Сохранить и укажите имя оснастки Шаблоны безопасности.
11. Закройте окно Консоли и выберите команду Пуск и далее Все программы.
12. Найдите пункт Администрирование и выберите подпункт Шаблоны безопасности (Теперь оснастка Шаблоны безопасности доступна из меню Пуск).
13. Для просмотра значений имеющихся шаблонов в окне оснастки откройте, например, узел Шаблоны безопасности, щелчком выберите шаблон безопасности compatws (рис. 2) и просмотрите его папки Политика учетных записей, Локальная политика и др.
14. Помимо раскрытого шаблона безопасности compatws.inf существуют и другие стандартные шаблоны, конфигурации которых позволяют получить различные по надежности системы безопасности.

##### **В. Редактирование и сохранение шаблона безопасности.**

1. Щелкните на одном из стандартных шаблонов безопасности (например, compatws), которые Вы видите в окне оснастки Шаблоны безопасности.
2. Если Вы хотите модифицировать какую-либо настройку безопасности, дважды щелкните на ней и отредактируйте значения параметров (рис. 3), например, политика учетных записей- политика паролей- минимальная длина паролей.
3. Для сохранения откорректированного стандартного шаблона безопасности под другим именем выполните следующие действия.
4. Укажите откорректированный стандартный шаблон (например, compatws), и нажмите правую кнопку мыши.
5. В появившемся контекстном меню выберите команду Сохранить как.
6. Введите с клавиатуры новое имя файла, например custom. По умолчанию шаблоны безопасности располагаются в каталоге  

$$\%SystemRoot%\Secunfy\Templates.$$
7. Пользовательский шаблон будет добавлен в определенную заранее конфигурацию безопасности и сохраните под введенным Вами именем.  
 Настроив шаблон безопасности для одной ПЭВМ, Вы можете перенести его на другие ПЭВМ Вашей рабочей группы. Шаблоны безопасности являются гибким и удобным инструментом для настройки системы безопасности операционной системы WindowsXP.

#### **Задания для самостоятельной работы:**

Создайте на базе существующего **Шаблона безопасности новый шаблон** и дайте ему имя ПЗ-8. После этого зафиксируйте список шаблонов, скопируйте изображение экрана в буфер и далее в файл для отчета.

#### **Контрольные вопросы:**

1. Для чего используются шаблоны безопасности?
2. В каком месте на диске хранятся (по умолчанию) шаблоны безопасности?
3. Какие разделы включает стандартный Шаблон безопасности?

## **Лабораторная работа № 29 Обновление и восстановление информационных систем**

**Цель работы:** изучение основных методов резервного копирования и восстановления информации. Освоение практических навыков администрирования при аварийно-восстановительных работах в информационных системах.

**Задание (перед выполнением ознакомьтесь с материалом по теме - файл «ЭИС. Теоретический материал для практических работ»).**

#### **Описание оболочки эмулятора архиватора базы данных**

В начале работы необходимо запустить файл "SQLServer2000Emu.exe". При этом на экране появится заставка симулятора Microsoft SQL Server 2000. При нажатии кнопки "Начать" высветится главное диалоговое окно. В верхней части окна расположено главное меню. Оно включает следующие пункты.

- **"Файл"** -содержит пункты по работе с файлами(создание базы данных,изменение ее структуры посредством реализации транзакций, просмотр базы данных и устройств резервного копирования, ввод транзакций вручную при осуществлении восстановления до точки сбоя);
- **"Работа"** -содержит пункты для создания устройства резервного копирования,выполнения резервного копирования и восстановления базы данных;
- **"Справка"** -сведения о программе.

1. Меню **"Файл"** включает в себя следующие пункты.

1.1 **"Создание Базы данных"**. После выбора этого пункта на экране появляется поле для ввода имени базы данных (по умолчанию используется имя "DataBase"). Этот пункт позволяет сформировать по случайному закону файлы исходной распределенной базы данных, размещенные на пяти рабочих станциях. Список формируемых файлов отображается в

специальном окне. Название файла, например F\_2\_5 – , соответствует тому, что на второй рабочей станции создан файл 5 распределенной базы данных.

1.2. **"Выполнение транзакций"**. Каждый раз при выборе этого пункта в базу данных будут вноситься изменения. Для запуска процесса генерации изменений необходимо в поле окна ввода указать число выполняемых транзакций. После нажатия кнопки "ОК" начнется формирование транзакций (по случайному закону). По каждой транзакции выводится следующая информация: номер транзакции, время ее окончания, имя файла, в котором производятся изменения, вид изменения.

1.4. **"Ввод транзакций вручную"**. Этот пункт используется только после того, как база данных была восстановлена после сбоя с помощью резервных копий, и далее необходимо внести последние изменения, которые не были внесены в копию базы данных при резервировании. При выборе пункта появляется окно для ввода транзакций.

Это окно содержит следующие элементы: "	<i>Имя базы</i>
<i>данных</i>	" -
для указания имени восстановленной базы данных; "	
<i>Номер станции</i>	

" - для выбора номера рабочей станции; "

*Имя файла*

" - для выбора файла, в котором производятся изменения; "

*Транзакция*

" - непосредственно само изменение; "

*Выполнить*

" – кнопка запуска выполнения транзакции над базой данных; "

*ОК*

" - завершение внесения изменений. Система отслеживает правильность ввода транзакции и сообщает об этом.

1.5. **"Просмотр"**. Этот пункт позволяет посмотреть содержимое исходной базы, восстановленной базы данных и устройства резервного копирования. После того как сделан выбор и запущен на восстановление файл резервного копирования, его структура отображается на экране в специальном окне.

1.6. **"Выход"**. Завершение работы с программой. При выходе выдается сообщение о выполнении или невыполнении лабораторной работы.

2. Меню **"Работа"** включает в себя следующие пункты.

2.1. **"Новое устройство резервного копирования"** - позволяет добавить новое устройство резервного копирования. При выборе этого пункта открывается окно свойств устройства резервного копирования, в котором в поле "

*Имя устройства*

" необходимо ввести имя добавляемого устройства, которое затем будет использоваться при выполнении резервного копирования. Поле "

*Имя файла*

" служит для указания пути и имени файла на диске, который будет выступать в качестве устройства резервного копирования. Имя файла и путь к нему можно задавать напрямую, записав его в окне ввода, или посредством обращения к дереву каталога компьютера, нажав клавишу "...".

2.2 **"Резервное копирование Базы данных"** - позволяет выполнить основные виды резервного копирования выбранной базы данных. Этот пункт позволяет управлять процедурой резервного копирования базы данных и содержит подпункты **"Резервное копирование"** и **"Мастер копирования"**.

2.2.1 Открывающееся при выборе пункта **"Резервное копирование"** окно содержит две вкладки: **"Общие"** и **"Параметры"**.

Элементами вкладки **"Общие"** являются:

- **"База данных"** - в этом раскрывающемся списке содержатся названия выбранных для резервирования баз данных;

- "*Имя*" - в этом поле необходимо ввести имя, которое будет присвоено создаваемой резервной копии базы данных;

- "*Описание*" - это поле предназначено для краткого комментария к создаваемой резервной копии. Данная информация имеет значение только для пользователя и носит вспомогательный характер;

- "*Резервная копия*" - это группа переключателей, с помощью которых можно установить необходимый тип резервного копирования; "*Расположение резервной копии*"

- с помощью этой группы переключателей выбирается тип носителя (лента или диск);

- "*Перезаписать*" – данная группа переключателей определяет, будет ли созданная резервная копия при сохранении на носителе записываться поверх имеющихся данных (переключатель "*Записать поверх существующей резервной копии*") или добавляться в конец копии (переключатель "*Добавить к существующей резервной копии*")

*Добавить к существующей резервной копии*

- "*Расписание*" - этот флаг позволяет задать расписание, по которому в последующем будет выполняться резервное копирование. Если он не установлен, то копирование будет выполнено один раз сразу же после щелчка на кнопке "ОК".

Элементами вкладки "*Параметры*" являются:

- "*Проверять резервную копию после создания*" – при установке флага производится проверка созданной копии на предмет соответствия размещения файлов копии, поименованных в каталоге;

- "*Проверять имя и срок хранения набора носителей*" - при установке этого флага будет осуществляться проверка возможности перезаписи резервной копии взамен уже существующей. Если попытка перезаписи делается на копию, срок хранения которой еще не истек, то пользователю выдается сообщение об ошибке и предлагается изменить параметры сохранения резервной копии. Если флаг снят, станет возможной перезапись резервной копии на носитель с еще актуальной резервной копией;

- "*Срок хранения резервной копии*" - с помощью данной группы переключателей выбирается, до какого времени резервная копия будет считаться актуальной и защищаться от перезаписи: "*После*" - в этом поле указывается количество дней, по истечении которых окончится срок хранения резервной копии

- в этом поле указывается конкретная дата, до которой архив будет защищен от перезаписи. Остальные элементы вкладки доступны только при работе с лентой.

2.2.2. "*Мастер копирования*" - вызов мастера резервного копирования. Мастер резервного копирования позволяет отработать процедуры архивирования данных с использованием диалоговых окон, существующих в СУБД SQL Server 2000. Мастер содержит несколько окон. В первом окне мастера приведена информация о том, какие сведения должны быть указаны в процессе его работы. Второе окно мастера позволяет выбрать базу данных для резервирования. В третьем окне мастера предлагается ввести имя (поле "*Имя*") и описание (поле "*Описание*") создаваемой резервной копии. Четвертое окно предназначено для выбора типа создаваемой резервной копии. Следующее окно мастера необходимо для выбора типа устройства резервного копирования (лента, файл или непосредственно устройство резервного копирования) и задания свойств (добавить к имеющейся копии, записать поверх, извлечь ленту после копирования, считать и проверить целостность резервной копии после резервного копирования). Шестое окно мастера позволяет установить флаг проверки имени набора носителей и срок хранения резервной копии, а также задать расписание и срок хранения копии. Последнее окно мастера содержит сводную информацию о параметрах, определяемых в ходе работы мастера.

2.3. "*Восстановление Базы данных*". Этот пункт служит для восстановления испорченной базы данных с помощью созданных ранее резервных копий. Открывающееся окно содержит следующие элементы:

- "*Восстановить*" - эта группа переключателей предназначена для выбора типа устройства или архива, с которого предполагается восстановить резервную копию (база данных, группы файлов или файлы, устройство);

- "*Тип восстановления*"- этот раскрывающийся список содержит список всех копий баз данных. Выбрав ту или иную копию и указав имя копии и устройство, на котором оно

расположено, после нажатия кнопки ОК можно с помощью таблицы, расположенной в нижней части вкладки, ознакомиться со списком резервных копий, созданных ранее для этой базы данных.

Если сбой произошел после создания копии и внесения дополнительных изменений в БД (выполнения транзакций), следует осуществить ввод транзакций (не сохраненных в копии) вручную, выбрав в меню "Файл" пункт "Ввод транзакций вручную". При этом вводятся все транзакции, включая и ту, на которой произошел сбой. Если после ручного ввода транзакций база данных примет вид, какой она имела на момент сбоя, то появится сообщение об успешном выполнении восстановления, иначе высветится сообщаемое о том, что база данных не восстановлена должным образом.

#### **Порядок выполнения работы**

1. Запустить файл "SQLServer2000Emu.exe".
2. Создать исходную базу данных (в имени базы данных должны присутствовать номер группы и варианта задания, например "Database035\_02"). Номер варианта определяется номером бригады в списке журнала преподавателя.
3. Сформировать файлы на рабочих станциях для исходной базы данных.
4. Внести изменения в базу данных (количество транзакций для каждого из вариантов задать, как  $15+N$ , где  $N$  – номер варианта).
5. Создать новое устройство резервного копирования (в имени файла-копии также отразить номер группы и вариант, например "Databackup0\_035\_2").
6. Создать полную резервную копию базы данных.
7. Просмотреть и зафиксировать содержимое исходной и резервной БД.
8. Внести новые изменения в базу данных (число транзакций -  $10+N$ ).
9. Создать дифференцированную резервную копию базы данных с использованием мастера резервного копирования.
10. Запротоколировать изменения, внесенные в базу данных между полным и дифференцированным режимами копирования.
11. Внести изменения в базу данных ( $10+N$  транзакций).
12. Создать резервную копию журнала транзакций на новом устройстве резервного копирования.
13. Просмотреть и зафиксировать содержимое журнала транзакций. Внести изменения в базу данных ( $5+N$  транзакций).
14. Просмотреть и зафиксировать параметры введенных транзакций (от начала последнего ввода транзакций до момента сбоя в системе включительно).
15. Выполнить восстановление базы данных, осуществляя последовательно восстановление с использованием полной, дифференциальной копий и копии журнала транзакций.
16. Ввести вручную зафиксированные транзакции, чтобы привести базу данных к виду, в котором она находилась на момент сбоя.
17. Проверить правильность восстановления информации.

#### **Содержание отчета**

Отчет о практической работе должен содержать:

- 1) краткие теоретические сведения;
- 3) описание содержимого базы данных (исходной, после полного и дифференциального резервного копирования) и журнала транзакций (после сохранения и до момента сбоя в информационной системе); описание условий, при которых производится копирование информации.

#### **Контрольные вопросы**

1. Какую задачу решает процедура резервного копирования и восстановления информации?
2. Что понимается под термином "транзакция"?
3. Перечислите виды архивирования и восстановления информации.
4. Как осуществляется оформление сопровождающей архив документации?
5. Какие требования учитываются при разработке стратегии архивирования?

6. Опишите процедуру восстановления информации до момента сбоя в системе.
7. Опишите принципы работы мастера резервного копирования СУБД SQL Server 2000.
8. Как осуществляется хранение резервных копий?
9. Какие особенности следует учитывать при выборе стратегии восстановления информации?
10. Почему достоверность информации, сохраняемой в копиях, зависит от интервала времени, через который производится архивирование?

## **МДК.01.02 Методы и средства проектирования информационных систем**

### **Практическая работа №1**

Выделение жизненных циклов проектирования информационных систем.

**Цель работы:** закрепить теоретические знания по теме 3.1., на примере описания готового проекта провести анализ модели анализ жизненного цикла.

- Задание:**
- Изучить теоретический материал;
  - Соотнести этапы жизненного цикла проекта ИС с каскадной моделью;
  - Соотнести этапы жизненного цикла проекта ИС со спиральной моделью;
  - Сделать вывод о целесообразности этапов различных моделей жизненного цикла;
  - Зафиксировать информацию в текстовом файле для отчета.

#### **Теоретический материал.**

Одним из базовых понятий методологии проектирования ИС является понятие жизненного цикла ее программного обеспечения (ЖЦ ПО). ЖЦ ПО -это непрерывный процесс, который начинается с момента принятия решения о необходимости его создания и заканчивается в момент его полного изъятия из эксплуатации.

Основным нормативным документом, регламентирующим ЖЦ ПО, является международный стандарт ISO/IEC 12207 (ISO – International Organization of Standardization - Международная организация по стандартизации, IEC - International Electrotechnical Commission –Международная комиссия по электротехнике). Он определяет структуру ЖЦ, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания ПО.

Структура ЖЦ ПО по стандарту ISO/IEC 12207 базируется на трех группах процессов:

1. основные процессы ЖЦ ПО (приобретение, поставка, разработка, эксплуатация, сопровождение);
2. вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, решение проблем);
3. организационные процессы (управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого ЖЦ, обучение).

Результатом выполнения данной практической работы является анализ этапов жизненного цикла готовой информационной системы. Для этого необходимо найти в сети Интернет проект разработки информационной системы по одной из тем (на выбор студента): поддержки курьерской доставки, расчета заработной платы, автоматизации

учебного процесса, коммерческого банка. Затем, ориентируясь на модели жизненного цикла, выделить в данном проекте этапы (стадии) проектирования ИС.

**Итог работы:** файл-отчет, конспект.

### **Практическая работа №2,3**

Проведение анализа предметной области.

**Цель работы:** закрепить теоретические знания по теме 3.1, проанализировать предметную область, описав предприятие (организацию).

- Задание:**
- Изучить теоретический материал;
  - Дать характеристику понятию «предметная область»;
  - Собрать предварительную информацию об исследуемом предприятии (организации);
  - Выполнить анализ предметной области;
  - Составьте отчет о структурном разбиении предметной области.

#### **Теоретический материал.**

В данном материале приводятся примеры описания предметной области. В качестве предметной области приведено описание работы видеобиблиотеки, получающей запросы на фильмы от клиентов и фильмы (носители – ленты, диски) возвращаемые клиентами. Запросы рассматриваются администрацией видеобиблиотеки с использованием информации о клиентах, фильмах и носителях. При этом проверяется и обновляется список арендованных носителей, проверяются записи о пользователях библиотеки. Администрация контролирует возврат носителей, используя информацию о фильмах, лентах и дисках, обновляемый список арендованных носителей. Обработка запросов на фильмы и возвратов носителей включает следующие действия: если клиент не является пользователем библиотеки, он не имеет права на аренду. Если требуемый фильм имеется в наличии, администрация информирует клиента об арендной плате. Если клиент просрочил срок возврата имеющихся у него носителей, ему не разрешается брать новые фильмы. Когда носитель возвращается, администрация рассчитывает арендную плату плюс пени за несвоевременный возврат.

Видеобиблиотека получает от поставщиков новые носители. При их поступлении в библиотеку, необходимая информация о новых носителях фиксируется. Информация о пользователях библиотек содержится отдельно от записей об аренде носителей.

Администрация библиотеки регулярно готовит отчеты за определенный период времени о пользователях библиотеки, поставщиках носителей, о выдаче носителей и об их приобретении.

Из описания предметной области следует, что в процессе работы библиотеки участвуют следующие группы людей: клиенты (пользователи), поставщики и руководство. Эти группы являются внешними объектами и обозначаются обычными прямоугольниками, а процессы – окружностями.

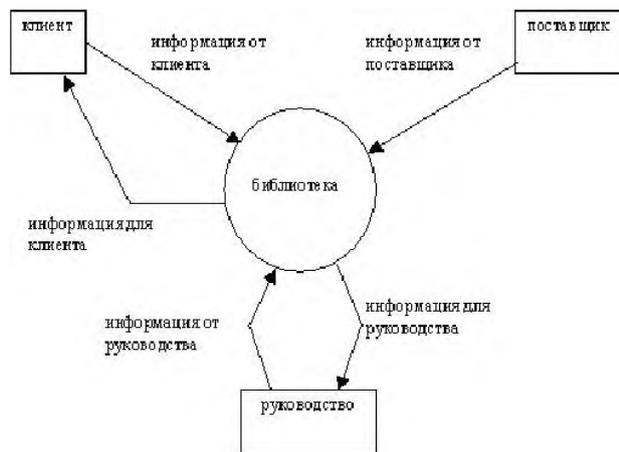


Рис. 1. Начальная контекстная диаграмма.

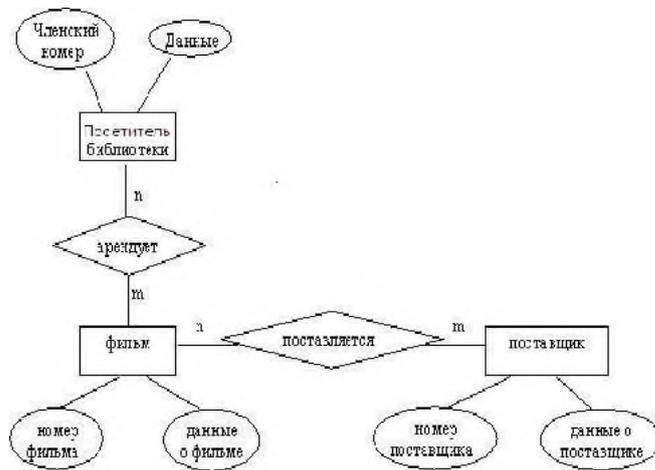


Рис. 3. Диаграмма "сущность-связь".

В процессе анализа строится глобальная модель данных, представляемая в виде диаграммы "сущность-связь" (рис. 3).

Результатом выполнения данной практической работы является описание предметной области организации. Описание должно включать следующие разделы:

1. *Краткое описание организации:*

- а) Название;
- б) Характеристика деятельности.

2. *Ресурсы организации:*

- а. Персонал – перечень должностей, например, Генеральный директор, Зам. генерального директора (2 чел.), Зав. отделом (3 чел.), Инженер (7 чел.) и т.д.;
- б. Здания, помещения (площадь, цель использования);
- с. Оборудование.

3. *Схемы и диаграммы:*

- а. Организационная диаграмма (структурно-функциональная схема организации) – иерархия управления в организации (Рис. 1). На ней должны быть отражены данные из п.2.а;
- б. Структурно-функциональная диаграмма (структурно-функциональная схема процессов) – структура подразделений, отражающая их взаимодействие между собой и с внешней средой в рамках деятельности организации – (Рис. 2);

**Итог работы:** файл-отчет, конспект.

### Практическая работа №4

Выбор модели построения информационной системы и программных средств.

**Цель работы:** закрепить теоретические знания по теме 3.1, выбрать модель построения информационной системы индивидуального проектного задания и программные средства, составить план реализации индивидуального проектного задания.

**Задание:**

- Изучить теоретический материал;
- Дать характеристику моделям построения ИС;
- Дать характеристику программным средствам для построения ИС;
- Выбрать оптимальную модель построения и программные средства для проектирования информационной системы предприятия (организации);
- Составить отчет в текстовом редакторе о целесообразности сделанного выбора.

## Теоретический материал.

Выявляемая в процессе исследования деятельности предприятия информация в большинстве своем не структурирована и требует формализации.

Такая формализация достигается путем построения моделей.

*Моделью* принято называть символическое описание системы, позволяющее получить информацию и ответить на все вопросы относительно системы. Процесс построения моделей называют моделированием.

Можно выделить различные *виды* моделей в зависимости от их назначения:

### 1. Познавательные и прагматические

*Познавательные* являются формой организации и представления знаний, средством соединения новых знаний с имеющимися. (При расхождении между моделью и реальностью изменяется модель.)

*Прагматические* являются средством управления, средством организации практических действий или их результата. (При расхождении между моделью и реальностью изменения должны приближать реальность к модели.)

### 2. Статические и динамические

*Статические* – модели конкретного состояния объекта.

*Динамические* – модели, в которых происходят какие бы то ни было изменения во времени.

*Статические модели* описывают содержательную сторону системы. Они могут быть:

- функциональными, т.е. описывать принципы функционирования системы;
- информационными, т.е. описывать структуру информации, на основе которой функционирует система;
- структурными, т.е. описывать структуру системы.

*Динамические модели* позволяют моделировать поведение системы во времени, учитывая фактор ее развития.

Рассмотрим наиболее общие модели систем.

### 1. Модель «Черного ящика» (рис.1).

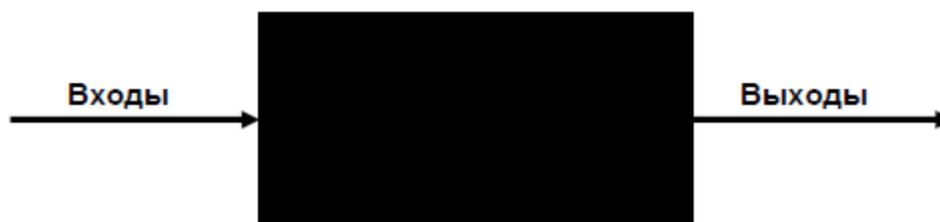


Рис.1. Схема модели «черного ящика»

Модель «черного ящика» представляет собой полностью непрозрачную систему, функционирование которой нам неизвестно. Однако мы можем сказать как наша система взаимодействует с внешней средой. Взаимодействие с внешней средой описывается в терминах входов и выходов. Входы реализуют возможность использования системы. Выходы соответствуют целям, для реализации которых была создана система. Данная модель отражает два важных свойства системы: целостность и обособленность от среды.

### 2. Модель состава (рис.2).

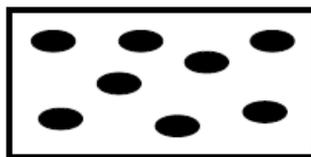


Рис.2. Схема модели состава

Данная модель позволяет нам судить о составе системы без рассмотрения взаимосвязи ее элементов. Части системы, которые мы рассматриваем как неделимые, называются элементами. Части системы, состоящие более чем из одного элемента, называются подсистемами (рис.3).

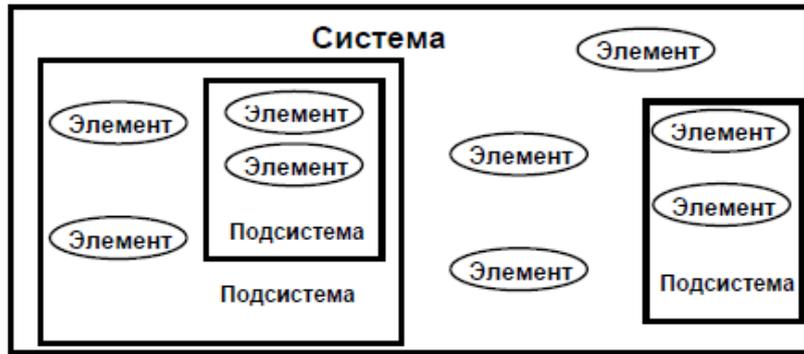


Рис.3. Схема многоуровневой системы

### 3. Модель структуры (рис. 4).

После определения состава системы необходимо установить связи между элементами (отношения). Совокупность необходимых и достаточных для достижения цели отношений между элементами называется *структурой системы*.

Абстрагируясь от содержательной стороны структурных схем (рис.4а), получают схемы, в которых обозначается только наличие элементов и связей между ними (рис.4б). Такие схемы называются *графами*.

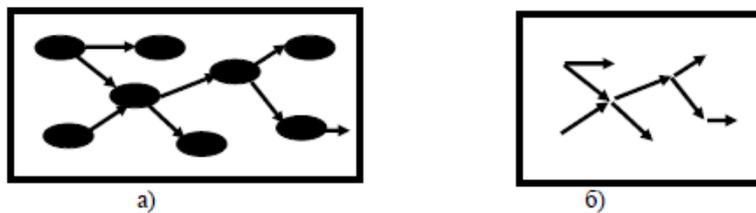


Рис.4. Схема модели структуры: а) модель с элементами; б) модель без элементов

### 4. Структурная схема системы.

Объединение рассмотренных трех моделей образует модель, которую будем называть *структурной схемой системы* (рис.5).

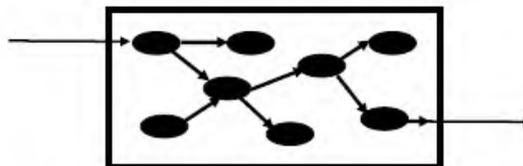


Рис.5. Модель структурной схемы системы

Итак, обобщая все сказанное выше и добавляя к рассмотрению признак времени, представим рассмотренные типы моделей в статическом (рис.6) и динамическом (рис.7) вариантах.

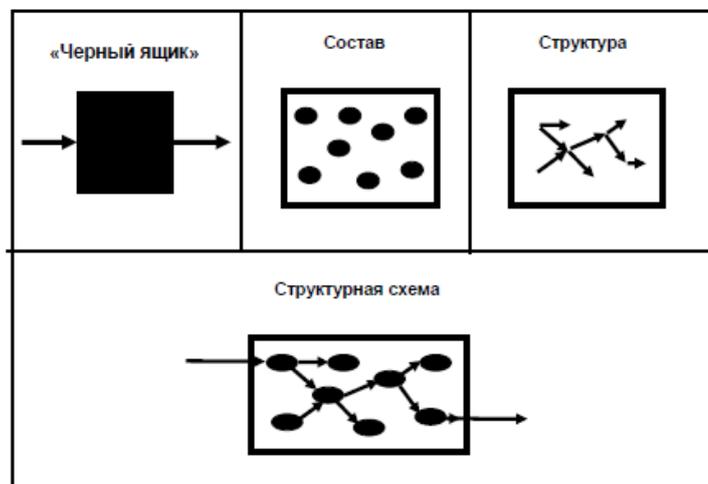


Рис.6. Типы моделей для статического варианта

<p>«Черный ящик»</p> <p>вход: начальное состояние</p> <p>Выход: конечное (желаемое) состояние</p>	<p>Состав</p> <p>Перечень действий, необходимых для перевода начального состояния в конечное</p>	<p>Структура</p> <p>Последовательность действий и продолжительность каждого действия</p>
<p>Сетевой график всего процесса</p>		

Рис.7. Типы моделей для динамического варианта

Приведем некоторые основополагающие рекомендации по структурированию моделей деятельности.

1. Основной принцип заключается в том, что структурирование должно осуществляться в соответствии с деятельностью и бизнес-процессами предприятия, а не в соответствии с его оргштатной структурой. Именно бизнес-процессы представляют ценность для клиента, и именно их улучшением предстоит в дальнейшем заниматься консультанту. Модель, основанная на оргштатной структуре, может продемонстрировать лишь хаос, царящий в организации (о котором, в принципе, руководству и так известно, иначе оно не воспользовалось бы услугами консультантов), на ее основе возможно внести предложения только об изменении этой структуры. С другой стороны, модель, основанная на бизнес-процессах, содержит в себе (не всегда в явном виде) и оргштатную структуру предприятия.

2. Верхний уровень модели должен отражать только контекст системы - взаимодействие моделируемого единственным контекстным процессом предприятия с внешним миром и ничего более.

3. На втором уровне модели должны быть отражены основные деятельности предприятия и их взаимосвязи.

4. Каждая из деятельностей, в свою очередь, должна быть детализирована на бизнес-процессы.

Сочетание различных признаков классификации методов проектирования обуславливает характер используемой технологии проектирования АИС. Выделяются два основных класса технологии проектирования: *каноническая* и *индустриальная*. Индустриальная технология проектирования в свою разбивается на два подкласса: *автоматизированное* (использование CASE-технологий) и *типичное* (параметрически-ориентированное или модельно-ориентированное) проектирование.

CASE-технология представляет собой методологию проектирования ИС, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать эту модель на всех этапах разработки и сопровождения ИС и разрабатывать приложения в соответствии с информационными потребностями пользователей. Большинство существующих CASE-средств основано на методологиях структурного (в основном) 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

CASE-средства позволяют создавать не только продукт, практически готовый к применению, но и обеспечить “правильный” процесс его разработки. Основная цель технологии – отделить проектирование программного обеспечения от его кодирования, сборки, тестирования и максимально “скрыть” от будущих пользователей все детали разработки и функционирования ПО. При этом значительно повышается эффективность работы проектировщика: сокращается время разработки, уменьшается число программных ошибок, программные модули можно использовать при следующих разработках.

В качестве примеров популярных CASE-средств укажем программные средства компании Computer Associates, IBM-Rational Software и Oracle:

- *BPwin* – моделирование бизнес-процессов;
- *ERwin* – моделирование баз данных и хранилищ данных;

- *ERwinExaminer* – проверка структуры СУБД и моделей, созданных в *Erwin*;
- *ModelMart* – среда для командной работы проектировщиков;
- *ParadigmPlus* – моделирование приложений и генерация объектного кода;
- *RationalRose* – моделирование бизнес-процессов и компонентов приложений;
- *RationalSuiteAnalystStudio* – пакет для аналитиков данных;
- *OracleDesigner* (входит в *Oracle9i DeveloperSuite*) – высокофункциональное средство проектирования программных систем и баз данных, реализующее технологию CASE и собственную методологию Oracle – CDM. Сложное CASE-средство, его имеет смысл использовать при ориентации на линейку продуктов Oracle.

Результатом выполнения данной практической работы является описание выбранной модели построения индивидуального проекта ИС, описание программных CASE-средств для проекта с обоснованием целесообразности выбора. Допускается выбор средств не из предложенного выше списка.

**Итог работы:** файл-отчет.

### Практическая работа №5,6

Формирование требований к автоматизированной системе.

**Цель работы:** закрепить теоретические знания по теме 3.1, для выбранного варианта информационной системы определить набор требований и спецификаций на создание информационной системы.

- Задание:**
- Изучить теоретический материал;
  - Дать характеристику учетным записям пользователя;
  - Проанализировать предметную область любой организации;
  - Выполнить структурное разбиение предметной области на отдельные подразделения;
  - Определить задачи и функции системы в целом и функции каждого подразделения;
  - Зафиксировать информацию в файле для отчета.

#### Теоретический материал.

Для успешной реализации базы данных объект проектирования должен быть прежде всего адекватно описан, должны быть построены полные и непротиворечивые функциональные модели базы данных. Опыт проектирования информационных систем показывает, что это логически сложная, трудоемкая и длительная по времени работа, требующая высокой квалификации участвующих в ней специалистов.

При проектировании информационной системы необходимо провести анализ целей этой системы и выявить требования к ней отдельных пользователей. Информация для построения модели информационной системы берется на основе проведения всестороннего обследования организации, для которой выполняется разработка информационной системы. Сбор данных начинается с изучения сущностей предметной области, процессов, использующих эти сущности, и связей между ними

Формирование требований включает этапы:

1. *Определение целей и показателей, на достижение которых направлено внедрение информационной системы.* Необходимо цели внедрения информационной системы и определить показатели достижения этих целей. Для этого необходимо составить таблицу по шаблону (таблица 1).

*Таблица 1. Цели внедрения информационной системы*

*и их показатели*

№	Цели	Показатели	Единица измерения показателя	Периодичность фиксации	Желаемый тренд
1					
2					
...					

*Например:* Основными целями внедрения информационной системы call-центра автосалона является повышение удовлетворенности клиентов и оптимизация расходов на телефонную связь. В таблице 2 представлены показатели достижения этой цели.

*Таблица 2. Цели и показатели бизнес-процессов call-центра*

№	Цели	Показатели	Единица измерения показателя	Периодичность фиксации	Желаемый тренд
1	Повышение удовлетворенности клиентов	Процент обработанных вызовов	%	1 раз в месяц	увеличение
		Процент не довольных клиентов	%	1 раз в месяц	уменьшение
2	Оптимизация расходов на телефонную связь	Средняя продолжительность звонка	мин	1 раз в месяц	уменьшение

2. *Проектирование новых (усовершенствование старых) бизнес-процессов предприятия, на автоматизацию которых направлена информационная система, – модель ТО-ВЕ.*

Если внедрение информационной системы на предприятие меняет структуру его текущих бизнес-процессов, то необходимо разработать модель ТО-ВЕ. Как правило, модель «как должно быть» изменяет регламент выполнения операций отдельными должностными лицами, поэтому изменения вносятся только на диаграммы нижних уровней.

Для проектирования бизнес-процессов «ТО-ВЕ» используются следующие методологии: IDEF0, DFD, IDEF3, EPC, Процесс, Процедура, BPMN.

3. *Выделение бизнес-процессов нижнего уровня, подлежащих автоматизации с указанием:*

- владельца процесса,
- исполнителя процесса,
- входных данных для реализации процесса,
- результат процесса,
- функция информационной системы, отвечающая за автоматизацию данного бизнес-процесса.

Необходимо внести в таблицу (таблица 3) данные обо всех автоматизируемых операциях.

Таблица 3. Автоматизируемые операции

№ бизнес-процесса	Название бизнес-процесса	Исполнитель	Входы (данные вносимые в информационную систему)	Выходы (данные и отчеты, генерируемые системой)	Функция информационной системы

Например: Автоматизируемые процессы для корпоративного портала представлены в таблице 4.

№ бизнес-процесса	Название бизнес-процесса	Исполнитель	Входы (данные вносимые в информационную систему)	Выходы (данные и отчеты, генерируемые системой)	Функция (модуль) информационной системы
		ров	телефон, пол, возраст, email		
A4.1.1	Авторизация	Сотрудник (любая должность)	логин, пароль		1. Учет сотрудников
A4.1.3	Формирование заявки	Сотрудник (любая должность)	текст заявки, исполнитель заявки	код заявки	2. Управление заявками
A5.2	Отчет о проектах	Директор		отчет о проектах	3. Управление проектами
...					

4. *Определение функций будущей информационной системы.* При необходимости определение модулей информационной системы и функций каждого из модулей.

Необходимо составить таблицу по шаблону (таблица 5).

Таблица 5. Модули и функции информационной системы

№ п/п	Модули	Функции
1		
2		
...		

Например: Модули и функции информационной системы управления торговлей (таблица 6).

Таблица 6. Модули и функции информационной системы управления торговлей

№ п/п	Модули	Функции
1	Управление продажами	Учет клиентов
		Учет заказов
		Формирование накладных
		Формирование счетов
2	Управление закупками	Учет поставщиков
		Формирование заказов на поставку
		Формирование актов приемки товаров
...		

### 5. Формирование характеристик автоматизированных рабочих мест.

Необходимо заполнить таблицу по шаблону (таблица 7). Данные komponуются в соответствии с таблицей 3.

*Таблица 7. Автоматизируемые рабочие места*

№ п/п	Должность	Процесс	Функция

Результатом выполнения данной практической работы является выполнение следующих этапов:

1. Проанализировать предметную область любой организации (гипотетической или реальной; например: оптовые продажи, склад, ломбард, аптека, реализация мягких игрушек и пр.), уточнив и дополнив ее, руководствуясь собственным опытом, консультациями и другими источниками (сеть Интернет).

2. Выполнить структурное разбиение предметной области на отдельные подразделения (отделы, службы, подсистемы, группы и пр.) согласно выполняемым ими функциям.

3. Определить задачи и функции системы в целом и функции каждого подразделения (подсистемы).

4. Выполнить словесное описание работы каждого подразделения (подсистемы), алгоритмов и сценариев выполнения ими отдельных работ.

5. Оформить отчет со следующими разделами:

- исходное задание;
- расширенное описание предметной области с учетом сделанных дополнений;
- состав подразделений (подсистем) информационной системы;
- перечень функций и задач системы в целом и каждого подразделения (подсистемы) в отдельности, отношения их между собой;
- подробное описание работы каждого подразделения (подсистемы);
- описание отдельных сценариев работ подразделений (подсистем);
- входная и выходная информация для каждого подразделения (подсистемы);

**Итог работы:** файл-отчет.

### **Практическая работа №7**

Моделирование бизнес-процессов средствами BPwin. Анализ среды.

**Цель работы:** закрепить теоретические знания по теме 3.1, провести анализ среды выбранного CASE-средства для проектирования бизнес-процессов предприятия.

- Задание:**
- Изучить теоретический материал;
  - Дать характеристику CASE-средствам;
  - Изучить интерфейс программы BPwin;
  - Сделать вывод о возможностях программы;
  - Зафиксировать информацию в файле для отчета.

#### **Теоретический материал.**

Пакет BPwin позволяет строить модели в нотациях IDEF0, IDEF3 и DFD. Он имеет простой и интуитивно понятный интерфейс пользователя, помогающий аналитику создавать сложные модели с минимальными усилиями.

При запуске VPwin в окне программы доступны главное меню, основная панель инструментов и рабочая область, в левой части которой — навигатор модели — ModelExplorer, а в правой область построения модели (рис.1).

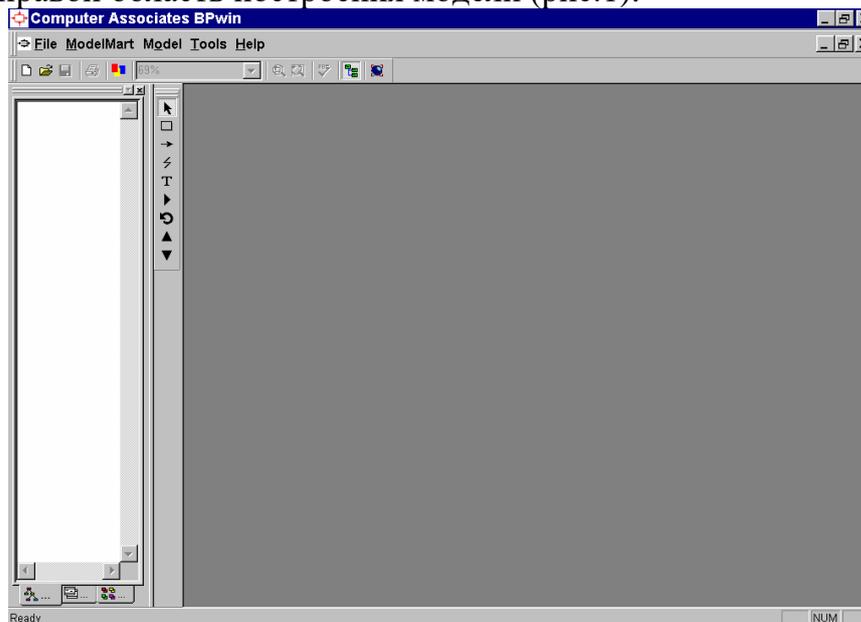


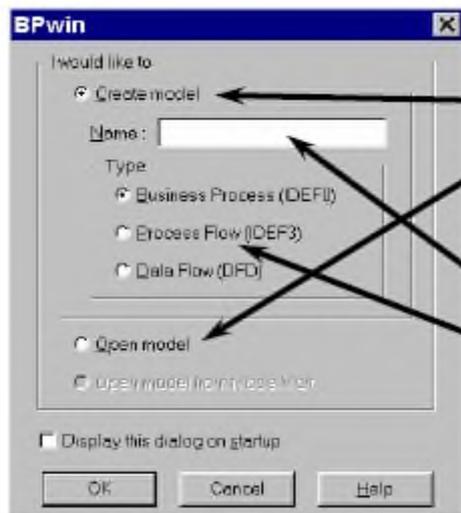
Рис. 1. Интерфейс пакета VPwin

Функции пакета доступны как с панели инструментов, так и из основного меню программы. Однако содержимое главного меню существенно расширяется после того, как модель открыта или создана. Набор функций в пунктах меню также зависит от того, в какой нотации создается модель (табл. 1).

Таблица 1

Описание элементов управления Vpwin Пункт меню	Описание
File/New	Создать новую модель
File/Open	Открыть модель
File/Save	Сохранить модель
File/Printsetup	Напечатать модель
ModelMart/Open	Открыть модель
ModelMart/Versions	Версия модели
ModelMart/LibraryManager	Менеджер библиотеки моделей
ModelMart/Connection	Связь
ModelMart/SessionManager	Менеджер сессии
ModelMart/SecurityManager	Менеджер доступа
ModelMart/Synchronizer	Синхронизатор
Model/Newmodelproperties	Свойства новой модели
Tools/Preferences	Настройка свойств системы
Tools/ Spellingoptions	Настройка проверки правописания
Help/HelpTopics	Справочная система – поиск по указателям
Help/VpwinOn-LineTutorial	Обучающая система
What'sNew	Что нового в этой версии
Howtousehelp	Как использовать помощь
AboutVpwin	О пакете

При создании новой модели необходимо заполнить диалог (рис.9). В нем указывают, будет ли:



- создана модель заново;
  - открыта из файла либо из репозитория ModelMart.
- Если модель создается заново, то необходимо:
- внести ее имя;
  - выбрать методологию, по которой она будет построена.

Модель в Bpwin, построенная на основе нотаций IDEF0 и IDEF3, рассматривается как совокупность «работ», каждая из которых оперирует с некоторым набором данных. Работа изображается в виде прямоугольников или функциональных блоков, данные — в виде стрелок или интерфейсных дуг. Если дважды щелкнуть по любому объекту модели левой кнопкой мыши (или один раз правой и затем выбрать из контекстного меню), появляется диалог ActivityProperties, каждая страница которого соответствует редактору какого-либо свойства объекта. Например, для установки шрифта (Font) (в том числе его размера и стиля) и цвета (Color) объекта вызываются соответствующие страницы диалога ActivityProperties

Результатом выполнения данной работы является описание функциональных возможностей программы Bpwin

**Итог работы:** файл-отчет.

### Практическая работа №8,9

Моделирование бизнес-процессов средствами Bpwin. Построение модели IDEF0.

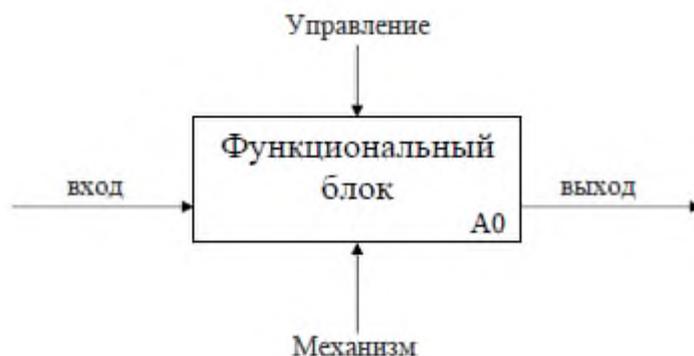
**Цель работы:** закрепить теоретические знания по теме 3.1, построить модель декомпозиции бизнес-процессов предприятия.

**Задание:**

- Изучить теоретический материал;
- Выполнить построение модели IDEF0 по образцу (файлы «Практическая работа. Моделирование бизнес-процессов.pdf» «Пособие по УИРС.pdf»);
- Выполнить декомпозицию IDEF0;
- Выполнить построение и декомпозицию IDEF0 по индивидуальному проекту;
- Сделать вывод о функциональном назначении модели построения IDEF0;
- Зафиксировать информацию в файле для отчета.

**Теоретический материал.**

В IDEF0 система представляется как совокупность взаимодействующих работ (или функций). Связи между работами определяют технологический процесс или структуру взаимосвязи внутри организации. Модель SADT представляет собой серию диаграмм, разбивающих сложный объект на составные части.



Основными понятиями методологии функционального моделирования работ являются:

*Работы (activity)* - поименованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. На диаграмме работы изображаются прямоугольниками.

*Вход (Input)* - материал или информация, которые используются работой для получения результата (стрелка, входящая в левую грань).

*Управление (Control)* - правила, стратегии, стандарты, которыми руководствуется работа (стрелка, входящая в верхнюю грань). В отличие от входной информации управление не подлежит изменению.

*Выход (Output)* - материал или информация, которые производятся работой (стрелка, исходящая из правой грани). Каждая работа должна иметь хотя бы одну стрелку выхода, так как работа без результата не имеет смысла и не должна моделироваться.

*Механизм (Mechanism)* - ресурсы, которые выполняют работу (персонал, станки, устройства - стрелка, входящая в нижнюю грань).

*Вызов (Call)* представляет собой взаимодействие одной модели работ с другой (стрелка, исходящая из нижней грани).

Различают в IDEF0 пять типов связей работ.

*Связь по входу (input-output)* имеет место, когда выход вышестоящей работы направляется на вход следующей работы.

*Связь по управлению (output-control)* обозначает ситуацию, когда выход вышестоящей работы направляется на управление следующей работы. Связь показывает доминирование вышестоящей работы.

*Обратная связь по входу (output-inputfeedback)* имеет место, когда выход нижестоящей работы направляется на вход вышестоящей. Используется для описания циклов.

*Обратная связь по управлению (output-controlfeedback)* обозначает ситуацию, когда выход нижестоящей работы направляется на управление вышестоящей. Является показателем эффективности бизнес-процесса.

*Связь выход-механизм (output-mechanism)* имеет место, когда выход одной работы направляется на механизм другой и показывает, что работа подготавливает ресурсы для проведения другой работы.

Из перечисленных блоков строятся диаграммы работ, описывающие принципы функционирования системы.

Результатом выполнения данной практической работы является построение серии диаграмм работ для всей информационной системы в целом и для отдельных сценариев работ, отражающих логику и взаимоотношение подразделений (подсистем).

Варианты индивидуальных проектов ИС:

1. Предприятие – книжный магазин. Задача – автоматизировать процесс организации продаж.

2. Предприятие – туристическая фирма. Задача – автоматизировать процесс выбора, бронирования, сопровождения туристических туров.
3. Предприятие – сервис по ремонту компьютерного оборудования. Задача – разработать информационную систему учета оказания услуг по ремонту компьютерной техники.
4. Предприятие – автосервис. Задача – разработать информационную систему учета оказания услуг автосервиса.
5. Предприятие – автовокзал. Задача – автоматизировать процесс организации продаж билетов.
6. Предприятие – магазин канцтоваров. Задача – автоматизировать процесс организации продаж через Интернет.
7. Предприятие – завод. Задача – разработать информационную систему кадрового учета.
8. Предприятие – образовательная школа. Задача – разработать информационную систему «Электронный журнал».
9. Предприятие – банк. Задача – автоматизировать работу сотрудников кредитного отдела.
10. Предприятие – завод. Задача – информационная система планирования поставок материальных ресурсов.
11. Предприятие – больница. Задача – информационная система учета пациентов.
12. Предприятие – компания-перевозчик. Задача – автоматизировать процесс управления перевозками.
13. Предприятие – магазин. Задача – разработать информационную систему складского учета.
14. Предприятие – медицинская лаборатория. Задача – разработка информационной системы учета медицинских анализов.
15. Предприятие – агентство организации праздников. Задача – информационная система управления отношениями с клиентами.
16. Предприятие – магазин бытовой техники. Задача – автоматизировать работу службы поддержки и обслуживания клиентов.
17. Предприятие – производство и продажа одежды. Задача – автоматизация деятельности маркетинга.
18. Предприятие – институт. Задача – разработать информационную систему технической поддержки.

**Итог работы:** файл-отчет.

### **Практическая работа №10,11**

Моделирование бизнес-процессов средствами Vpwin. Диаграммы дерева узлов и FEO.  
Слияние моделей.

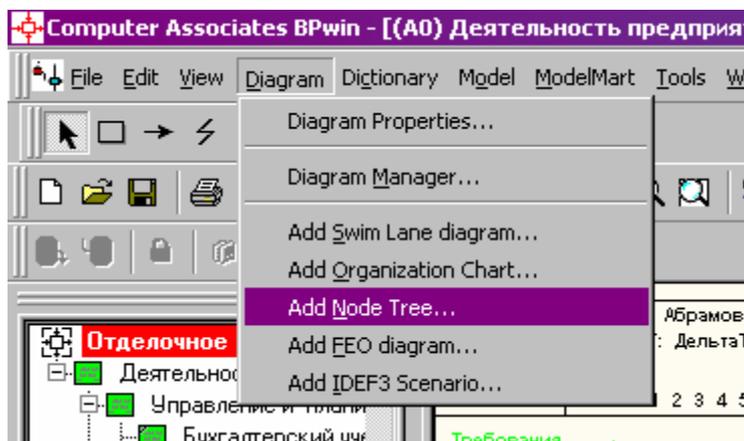
**Цель работы:** закрепить теоретические знания по теме 3.1, построение диаграммы дерева узлов.

- Задание:**
- Изучить теоретический материал;
  - Выполнить построение модели FEO по образцу (файлы «Практическая работа. Моделирование бизнес-процессов.pdf» «Пособие по УИРС.pdf»);
  - Выполнить слияние моделей индивидуального проекта;
  - Сделать вывод о проделанной работе;
  - Зафиксировать информацию в файле для отчета.

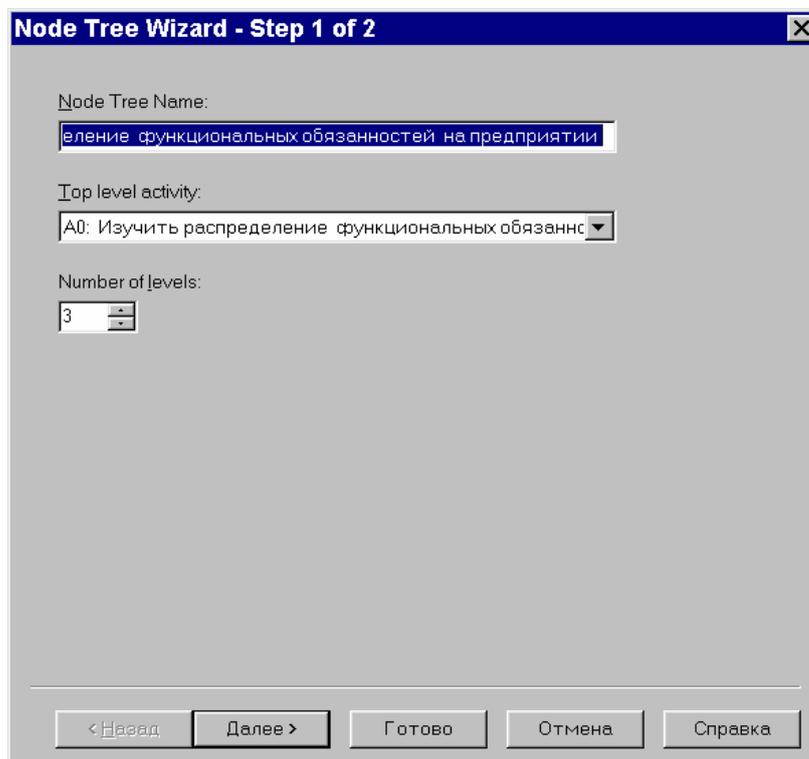
## Теоретический материал.

Процесс создания модели работ является итерационным, следовательно, работы могут менять свое расположение в дереве узлов многократно. Чтобы не запутаться и проверить способ декомпозиции, следует после каждого изменения создавать диаграмму дерева узлов. Впрочем, BPwin имеет мощный инструмент навигации по модели ModelExplorer, который позволяет представить иерархию «работ» и диаграмм в удобном и компактном виде, однако этот инструмент не является составляющей стандарта IDEF0.

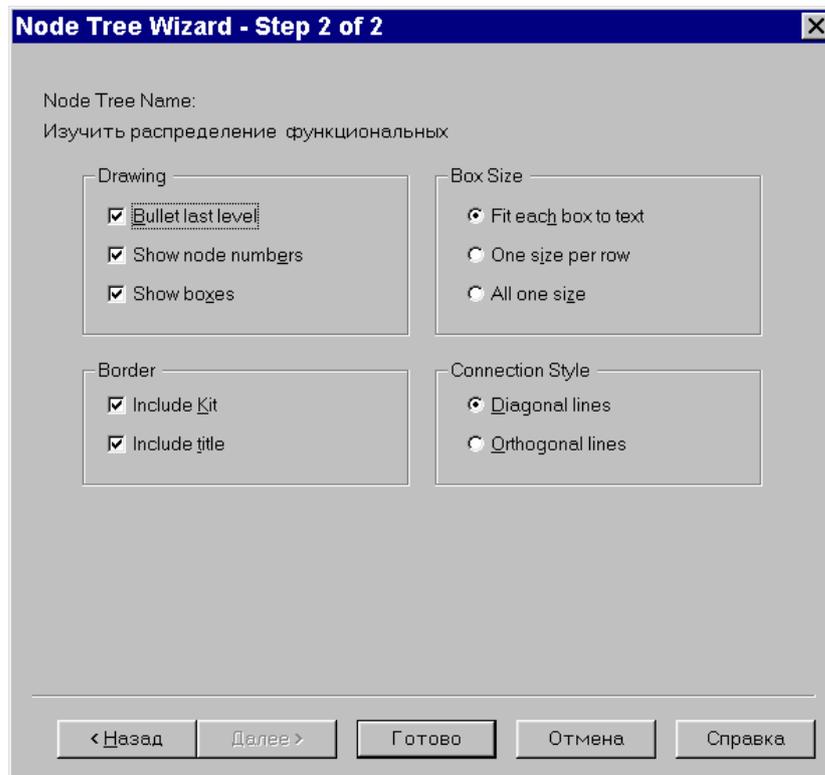
Для создания диаграммы дерева узлов следует выбрать в меню Diagram пункт AddNodeTree



Диалог формирования диаграммы предлагает два шага для настройки дерева узлов. В диалогe NodeTreeWizard на первом шаге указывают глубину — дерева - NumberofLevels (по умолчанию 3) и корень дерева (по умолчанию — родительская «работа» текущей диаграммы)

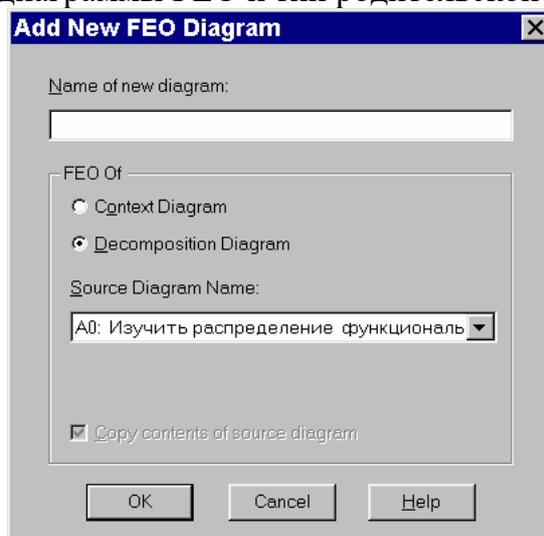


По умолчанию нижний уровень декомпозиции показывается в виде списка, остальные «работы» - в виде прямоугольников. Для отображения всего дерева в виде прямоугольников на втором шаге следует выключить опцию BulletLastLevel



### Диаграммы "только для экспозиции" (FEO)

Эти 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. Например, «работа» на диаграмме FEO может не иметь стрелок управления и выхода. С целью обсуждения определенных аспектов модели с экспертом предметной области может быть создана диаграмма только с одной «работой» и одной стрелкой, поскольку стандартная диаграмма декомпозиции содержит множество деталей, не относящихся к теме обсуждения и дезориентирующих эксперта. Но если FEO используется для иллюстрации альтернативных точек зрения (альтернативный контекст), рекомендуется все-таки придерживаться синтаксиса IDEF0. Для создания диаграммы FEO следует выбрать пункт меню Diagram – Add FEO Diagram. В возникающем диалоге AddNew FEO Diagram следует указать имя диаграммы FEO и тип родительской диаграммы



Результатом выполнения работы являются диаграммы, построенные по тематике индивидуальных проектов.

**Итог работы:** файл-отчет.

## Практическая работа №812

Моделирование бизнес-процессов средствами VPwin. Создание отчетов.

**Цель работы:** закрепить теоретические знания по теме 2.3, провести сравнительный анализ элементов популярных стеков протоколов.

- Задание:**
- Изучить теоретический материал;
  - Создать отчет в программе VPwin (файлы «Практическая работа. Моделирование бизнес-процессов.pdf» «Пособие по УИРС.pdf»);
  - Сделать вывод о проделанной работе;
  - Зафиксировать информацию в виде таблиц в файле для отчета.

### Теоретический материал.

VPwin имеет мощный инструмент генерации. Отчеты по модели вызываются из пункта меню Report. Всего имеется семь типов отчетов:

1. ModelReport. Он включает информацию о контексте модели: имя модели, точку зрения, область, цель, имя автора, дату создания и др.

2. DiagramReport. Отчет по конкретной диаграмме. Включает список объектов (работ, стрелок, хранилищ данных, внешних ссылок и т. д.).

3. DiagramObjectReport. Наиболее полный отчет по модели. Может включать полный список объектов модели (работ, стрелок с указанием их типа и др.) и свойства, определяемые пользователем.

4. ActivityCostReport. Отчет о результатах стоимостного анализа.

5. ArrowReport. Отчет по стрелкам. Может содержать информацию из словаря стрелок, информацию о работе-источнике, работе-назначении стрелки и информацию о разветвлении и слиянии стрелок.

6. DataUsageReport. Отчет о результатах связывания модели процессов и модели данных.

7. ModelConsistencyReport. Отчет, содержащий список синтаксических ошибок модели.

Синтаксические ошибки IDEF0 с точки зрения VPwin разделяются на три типа:

Во-первых, это ошибки, которые программа выявить не в состоянии. Например, синтаксис IDEF0 требует, чтобы имя работы было выражено отглагольным существительным или глагольной формой, выражающей действие («Изготовление изделия», «Обслуживание клиента» и т.д.)

Ошибки второго типа VPwin просто не допускает. Например, каждая грань работы предназначена для определенного типа стрелок. VPwin просто не позволит создать на диаграмме IDEF0 внутреннюю стрелку, выходящую из левой грани работы и входящую в правую грань.

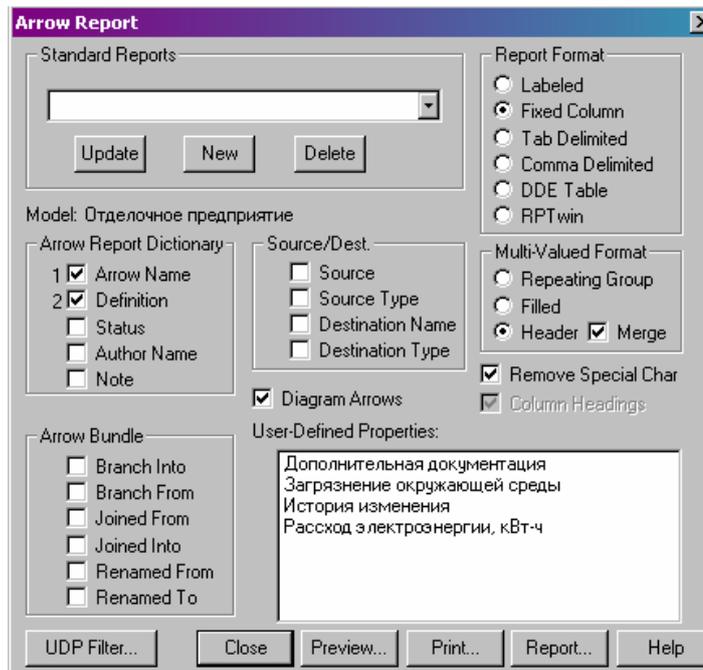
Третий тип ошибок VPwin позволяет допустить, но детектирует их. Полный их список можно получить в отчете ModelConsistencyReport.

Список ошибок может содержать, например, неименованные работы и стрелки (unnamedarrow, unnamedactivity), несвязанные стрелки (unconnectedborderarrow), неразрешенные стрелки (unresolved (squareunneled) arrowconnections).

При выборе пункта меню, который соответствует какому-либо отчету, появляется диалог настройки отчета. Для каждого из семи типов отчетов он выглядит по-своему. Рассмотрим типичный диалог ArrowReport.

Раскрывающийся список StandartReports позволяет выбрать один из стандартных отчетов. Стандартный отчет — это запоминаемая комбинация переключателей, флажков и

других элементов управления диалога. Для создания собственного стандартного отчета необходимо задать опции отчета, ввести имя отчета в поле списка выбора и щелкнуть по кнопке New.



Программа сохраняет информацию о стандартном отчете в файле BPWINRPT.INI. Все определения этого файла доступны из любой модели. Единственное ограничение — свойства, определяемые пользователем (UserDefinedProperties). Они сохраняются в виде указателя и поэтому доступны только из "родной" модели. Стандартный отчет можно изменить (кнопка Update) или удалить (кнопка Delete).

В правом верхнем углу диалога находится группа управляющих элементов для выбора формата отчета. Доступны следующие форматы:

- Labeled — отчеты включают метку поля, затем, в следующей строке, печатается содержимое поля.
- FixedColumn — каждое поле печатается в собственной колонке.
- Tab-CommaDelimited — каждое поле печатается в собственной колонке. Колонки разделяются знаком табуляции или запятыми.
- RPTwin — отчет создается в формате специализированного генератора отчетов. Опция Ordering (на отчете по стрелкам отсутствует) сортирует данные по какому-либо значению. Опция Multi-ValuedFormat регулирует вывод полей в отчете при группировке данных
- RepeatingGroup — детальные данные объединяются в одно поле, между значениями вставляется +.
- Filled — дублирование данных для каждого заголовка группы.
- Header (опция по умолчанию) — печатается заголовок группы, затем - детальная информация.

Результатом выполнения данной работы является отчет по индивидуальному проекту ИС.

**Итог работы:** файл-отчет.

### Практическая работа №13

Моделирование бизнес-процессов средствами Bpwin. Диаграммы потоков данных.

**Цель работы:** закрепить теоретические знания по теме 3.1, построить диаграммы потоков данных по индивидуальному проекту.

**Задание:**

- Изучить теоретический материал;
- Выполнить построение модели DFD(файлы «Практическая работа. Моделирование бизнес-процессов.pdf» «Пособие по УИРС.pdf»);
- Сделать вывод о проделанной работе;
- Зафиксировать информацию в виде таблиц в файле для отчета.

**Теоретический материал.**

Диаграммы потоков данных (DataFlowDiagrams - DFD) используются для описания движения документов и обработки информации как дополнение к IDEF0. В отличие от IDEF0, где система рассматривается как взаимосвязанные работы, стрелки в DFD показывают лишь то, как объекты (включая данные) движутся от одной работы к другой. DFD отражает функциональные зависимости значений, вычисляемых в системе, включая входные значения, выходные значения и внутренние хранилища данных. DFD - это граф, на котором показано движение значений данных от их источников через преобразующие их процессы к их потребителям в других объектах.

DFD содержит процессы, которые преобразуют данные, потоки данных, которые переносят данные, активные объекты, которые производят и потребляют данные, и хранилища данных, которые пассивно хранят данные.

Диаграмма потоков данных содержит:

- процессы, которые преобразуют данные;
- потоки данных, переносящие данные;
- активные объекты, которые производят и потребляют данные;
- хранилища данных, которые пассивно хранят данные.

Процесс DFD преобразует значения данных и изображается в виде эллипса, внутри которого помещается имя процесса.

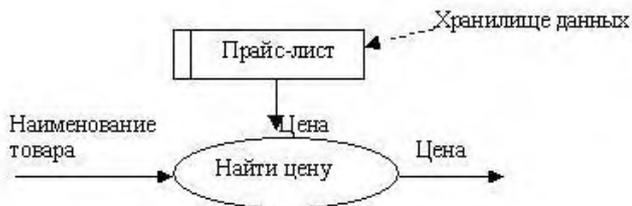


Поток данных соединяет выход объекта (или процесса) с входом другого объекта (или процесса) и представляет собой промежуточные данные вычислений. Поток данных изображается в виде стрелки между производителем и потребителем данных, помеченной именами соответствующих данных. Дуги могут разветвляться или сливаться, что означает соответственно разделение потока данных на части либо слияние объектов.

Активным объектом является объект, который обеспечивает движение данных, поставляя или потребляя их. Хранилище данных - это пассивный объект в составе DFD, в котором данные сохраняются для последующего доступа.



Хранилища данных. Хранилище данных - это пассивный объект в составе DFD, в котором данные сохраняются для последующего доступа. Хранилище данных допускает доступ к хранимым в нем данным в порядке, отличном от того, в котором они были туда помещены. Агрегатные хранилища данных, как, например, списки и таблицы, обеспечивают доступ к данным в порядке их поступления, либо по ключам.



Потоки управления. DFD показывает все пути вычисления значений, но не показывает в каком порядке значения вычисляются. Решения о порядке вычислений связаны с управлением программой, которое отражается в динамической модели. Эти решения, вырабатываемые специальными функциями, или предикатами, определяют, будет ли выполнен тот или иной процесс, но при этом не передают процессу никаких данных, так что их включение в функциональную модель необязательно. Тем не менее, иногда бывает полезно включать указанные предикаты в функциональную модель, чтобы в ней были отражены условия выполнения соответствующего процесса. Функция, принимающая решение о запуске процесса, будучи включенной в DFD, порождает в диаграмме поток управления и изображается пунктирной стрелкой.

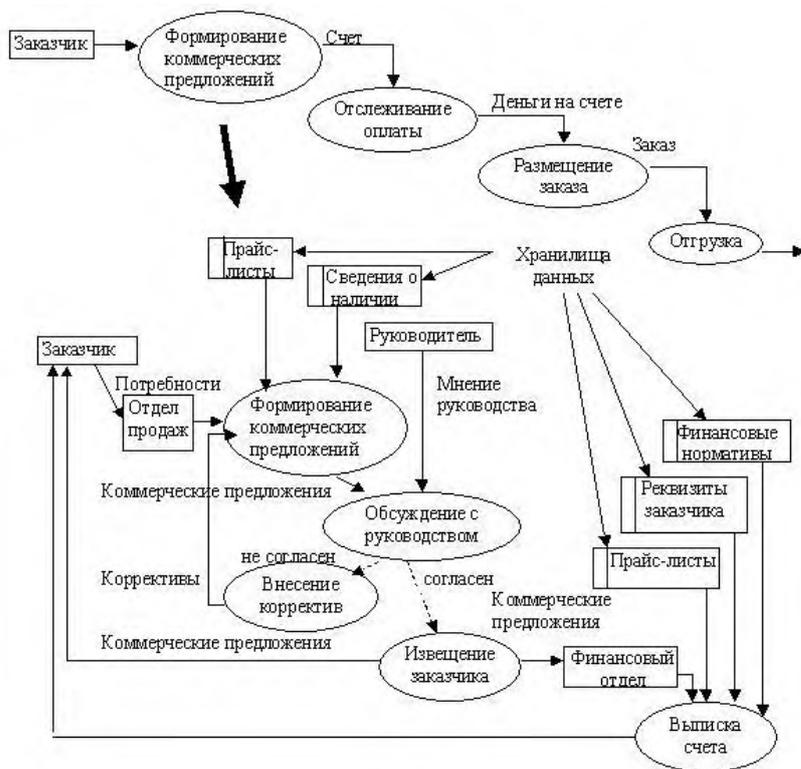


Первым шагом при построении иерархии DFD является построение контекстных диаграмм. Обычно при проектировании относительно простых информационных систем строится единственная контекстная диаграмма со звездобразной топологией, в центре которой находится так называемый главный процесс, соединенный с приемниками и источниками информации, посредством которых с системой взаимодействуют пользователи и другие внешние системы.

Если же для сложной системы ограничиться единственной контекстной диаграммой, то она будет содержать слишком большое количество источников и приемников информации, которые трудно расположить на листе бумаги нормального формата, и, кроме того, главный единственный процесс не раскрывает структуры распределенной системы.

При построении иерархии DFD переходить к детализации процессов следует только после определения содержания всех потоков и накопителей данных, которое описывается при помощи структур данных. Структуры данных конструируются из элементов данных и могут содержать альтернативы, условные вхождения и итерации. Условное вхождение означает, что данный компонент может отсутствовать в структуре. Альтернатива означает, что в структуру может входить один из перечисленных элементов. Итерация означает вхождение любого числа элементов в указанном диапазоне. Для каждого элемента данных может указываться его тип (непрерывные или дискретные данные). Для непрерывных данных может указываться единица измерения (кг, см и т.п.), диапазон значений, точность представления и форма физического кодирования. Для дискретных данных может указываться таблица допустимых значений.

Ниже приведена диаграмма потоков данных верхнего уровня с ее последующим уточнением:



Результатом выполнения данной работы является разработка DFD модели по образцу из файла «Практическая работа. Моделирование бизнес-процессов» и модели по варианту индивидуального проекта.

**Итог работы:** файл-отчет.

### Практическая работа №14

Моделирование бизнес-процессов средствами Bpwin. Описание процессов (IDEF3).

**Цель работы:** закрепить теоретические знания по теме 3.1, изучить и применить для разработки ИС метод описания процессов.

**Задание:**

- Изучить теоретический материал;
- Выполнить построение модели IDEF3 (файлы «Практическая работа. Моделирование бизнес-процессов.pdf» «Пособие по УИРС.pdf»);
- Сделать вывод о проделанной работе;
- Зафиксировать информацию в виде таблицы в файле для отчета.

#### Теоретический материал.

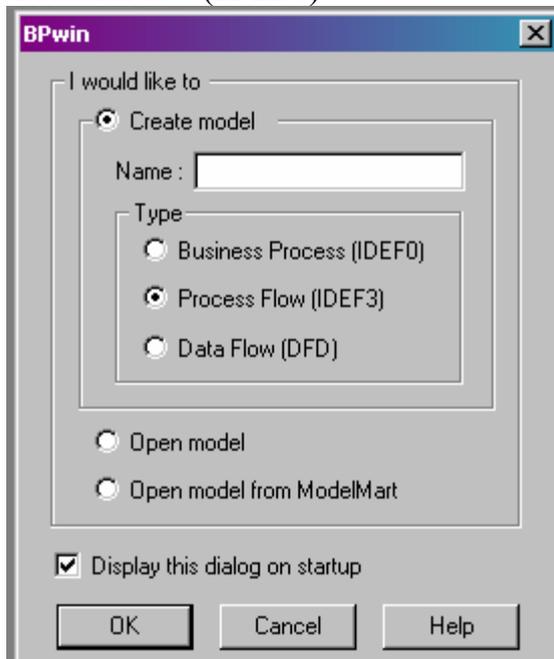
Наличие в диаграммах DFD элементов для описания источников, приемников и хранилищ данных позволяет более эффективно и наглядно описать процесс документооборота. Однако для описания логики взаимодействия информационных потоков более подходит IDEF3, называемая также workflow diagramming – методология проектирования, использующая графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов.

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 любой функции.

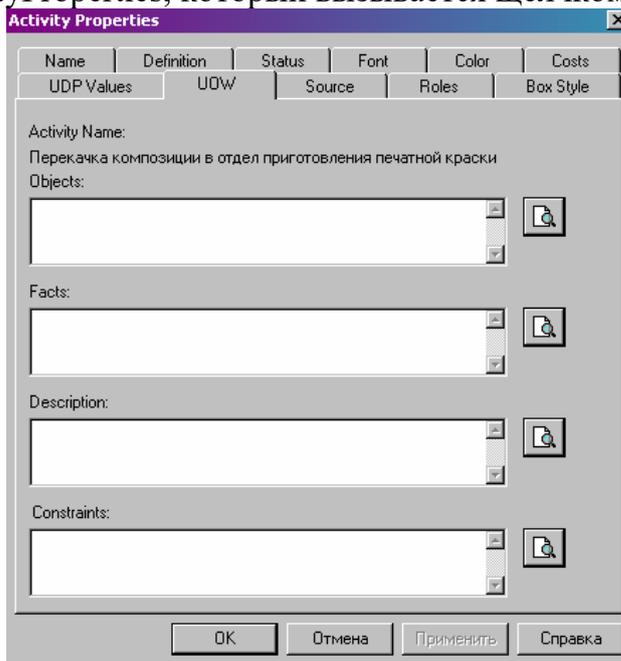
IDEF3 - это метод, имеющий основной целью дать возможность аналитикам описать ситуацию, когда процессы выполняются в определенной последовательности, а также описать объекты, участвующие совместно в одном процессе.

IDEF3 является частью структурного анализа и дополняет IDEFO. IDEF3 содержит все необходимое для построения моделей, которые в дальнейшем могут быть использованы для имитационного анализа.

Для того чтобы создать модель IDEF3, нужно в диалоге BPwin поле Type выбрать опцию ProcessFlow (IDEF3)



Центральными компонентами модели IDEF3 являются единицы работы - UnitofWork. Каждая UOW должна иметь ассоциированный документ, который включает текстовое описание компонентов работы: объектов (Objects) и фактов (Facts), связанных с работой, ограничений (Constraints), накладываемых на работу, и дополнительное описание работы (Description). Эта информация заносится во вкладку UOW диалога ActivityProperties, который вызывается щелчком правой кнопки мыши по работе



Связи показывают взаимоотношения работ. Все связи однонаправленны и могут быть направлены куда угодно.

Результатом выполнения работы является разработка диаграммы IDEF3 по образцу из файла «Пособие по УИРС» и диаграммы IDEF3 по заданию индивидуального проекта.

**Итог работы:** файл-отчет.

### Практическая работа №15

Составление технического задания на разработку информационной системы.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомиться с процедурой разработки технического задания на создание программного продукта.

- Задание:**
- Изучить теоретический материал;
  - Изучить информацию в файле «Пример технического задания»;
  - Разработать техническое задание для индивидуального проекта;
  - Сделать вывод о проделанной работе;
  - Зафиксировать информацию для отчета в виде файла.

#### Теоретический материал.

*Техническое задание (ТЗ)* – это завершающий предпроектную стадию документ, который содержит цели и обоснование проектирования, а также определяет основные требования к ИС и исходные данные необходимые при разработке.

Техническое задание составляется по результатам предпроектного обследования объекта автоматизации. В настоящее время при составлении технического задания обычно руководствуются требованиями следующих ГОСТов:

- 34.602-89 «Техническое задание на создание автоматизированной системы» – описывает состав и содержание ТЗ, которые распространяются на автоматизированную (информационную) систему в целом, в том числе:
  - общесистемные требования к ИС;
  - требования к компонентному составу ИС;
  - требования к интеграции компонентов ИС между собой и с другими системами;
  - требования к составу и содержанию работ по внедрению ИС;
  - ...
- 19.201-78 «Техническое задание. Требования к содержанию и оформлению» – входит в единую систему программной документации и устанавливает порядок построения и оформления технического задания на программное изделие для ЭВМ.

При проектировании сложных систем обычно разрабатывают общее техническое задание на ИС в целом (в соответствии с требованиями ГОСТа 34.602-89), а также дополнительные технические задания на части системы:

- на создание информационно-вычислительной сети,
- на отдельные подсистемы ИС,
- на элементы программного обеспечения ИС – программные компоненты и/или комплексы (в соответствии с требованиями ГОСТа 19.201-78).

При составлении технического задания целесообразно учитывать следующие практические рекомендации:

- a. Все изменения в структуре ТЗ (по сравнению с ГОСТ) должны быть обязательно согласованы с заказчиком.
- b. При составлении ТЗ целесообразно использовать методику «дробления и детализации». Это значит, что структура документа (разбиение на разделы и

- подразделы) должна быть тщательно проработана, так чтобы заинтересованное лицо могло быстро найти необходимые ему сведения относительно ИС по содержанию ТЗ.
- с. Сведения о функциональном назначении должны давать ответ на вопрос «для чего?», а сведения об эксплуатационном назначении – на вопрос «для кого?». Описание функционального назначения разработки должно содержать общие сведения без указания отдельных деталей.
  - d. Требования к программе желательно составлять на основе ГОСТов и нормативно-технической документации заказчика.
  - e. При составлении требований к программе целесообразно использовать метод «шаблонного построения фраз», например
    - При изложении требований к функциональным и иным характеристикам: «Программа должна обеспечивать возможность ...» или «Требования к ... не предъявляются».
    - При изложении требований к квалификации персонала: «Каждый пользователь должен обладать практическими навыками работы с графическим пользовательским интерфейсом ОС»;
    - и. т.п.
  - f. Требования к программным изделиям должны носить *императивный* характер. Если какие-либо требования (из перечисленных в ГОСТе) не предъявляются, об этом следует указывать специально.
  - g. Требования к пользовательскому интерфейсу рекомендуется оформлять в разделе «Специальные требования».
  - h. Минимальный состав программной документации, который должен упоминаться в соответствующем разделе ТЗ, включает в себя:
    - Спецификация программной документации;
    - Техническое задание;
    - Программа и методики испытаний;
    - Руководства администратора и оператора
  - i. В раздел «Технико-экономические показатели» можно включать оценку потребности в программном изделии и приблизительную оценку стоимости и трудоёмкости разработки.
  - j. Стадии и этапы разработки обычно излагаются в форме таблицы:

Содержание	Сроки	Исполнители	Отчёт
...	...	...	...

- k. В разделе «Порядок контроля и приёмки» рекомендуется указывать:
  - Какие функции программного изделия подлежат испытанию;
  - В какие сроки и чьими силами разрабатываются программные испытания;
  - Срок проведения испытания;
  - Оформление испытания;
  - Иные условия (например, на какой технике проводятся испытания)

В соответствии с полученным вариантом задания необходимо разработать проект технического задания на программное изделие для ИС. Руководствоваться требованиями Единой системы программной документации (ЕСПД), в частности, ГОСТом 19.201-78.

Результатом выполнения данной стадии является техническое задание, оформленное в соответствии со стандартами

**Итог работы:** файл-отчет.

### **Практическая работа №16,17**

Разработка модели фрагмента ИС с применением UML.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомиться с методологиями моделирования прецедентов и деятельности на основе языка UML.

- Задание:**
- Изучить теоретический материал;
  - Изучить интерфейс программного продукта MSVisio для построения схем UML;
  - Построить схему UML по проектируемой информационной системе;
  - Сделать вывод о проделанной работе;
  - Зафиксировать информацию в виде скриншотов в файле для отчета.

#### **Теоретический материал.**

UML (Universal Modeling Language) - универсальный язык моделирования, который был разработан компанией Rational Software с целью создания наиболее оптимального и универсального языка для описания как предметной области, так и конкретной задачи в программировании. Визуальное моделирование в UML можно представить как некоторый процесс поуровневого спуска от наиболее общей и абстрактной концептуальной модели системы к логической, а затем и к физической модели соответствующей системы. Любая задача, таким образом, моделируется при помощи некоторого набора иерархических диаграмм, каждая из которых представляет собой некоторую проекцию системы.

Диаграмма (Diagram) - это графическое представление множества элементов. Чаще всего она изображается в виде связанного графа с вершинами (сущностями) и ребрами (отношениями).

В UML определено восемь видов диаграмм:

- диаграмма прецедентов (Usecasediagram) - диаграмма поведения, на которой показано множество прецедентов и актеров, а также отношения между ними;
- диаграмма деятельности (Activitydiagram) - диаграмма поведения, на которой показан автомат и подчеркнуты переходы потока управления от одной деятельности к другой;
- диаграмма классов (Classdiagram) - структурная диаграмма, на которой показано множество классов, интерфейсов, коопераций и отношения между ними;
- диаграмма состояний (Statechartdiagram) - диаграмма поведения, на которой показан автомат и подчеркнуто поведение объектов с точки зрения порядка получения событий;
- диаграмма последовательностей (Sequencediagram) - диаграмма поведения, на которой показано взаимодействие и подчеркнута временная последовательность событий;
- диаграмма кооперации (Collaborationdiagram) - диаграмма поведения, на которой показано взаимодействие и подчеркнута структурная организация объектов, посылающих и принимающих сообщения;
- диаграмма компонентов (Componentdiagram) - диаграмма поведения, на которой показан автомат и подчеркнуто поведение объектов с точки зрения порядка получения событий;
- диаграмма развертывания (Deploymentdiagram) - структурная диаграмма, на которой показаны узлы и отношения между ними.

#### **Диаграмма прецедентов**

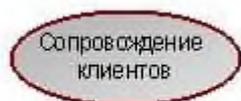
Диаграммы прецедентов применяются для моделирования вида системы с точки зрения внешнего наблюдателя. На диаграмме прецедентов графически показана совокупность прецедентов и Субъектов, а также отношения между ними.

Рассмотрим основные элементы диаграммы прецедентов.

Субъект (actor) - **любая сущность, взаимодействующая с системой извне [2] или множество логически связанных ролей, исполняемых при взаимодействии с прецедентами [1]**. Стандартным графическим обозначением субъекта на диаграммах является фигурка "человечка", под которой записывается конкретное имя субъекта, однако субъектом может быть не только человек, но и техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик.



Прецеденты (usecase) - **это описание множества последовательностей действий (включая их варианты), которые выполняются системой для того, чтобы актер получил результат, имеющий для него определенное значение**. При этом ничего не говорится о том, каким образом будет реализовано взаимодействие субъектов с системой, это одна из важнейших особенностей разработки прецедентов. Стандартным графическим обозначением прецедента на диаграммах является эллипс, внутри которого содержится краткое название прецедента или имя в форме глагола с пояснительными словами.



Сущность концепции прецедентов подразумевает несколько важных пунктов.

- Прецедент представляет собой **завершенный** фрагмент функциональных возможностей (включая основной поток логики управления, его любые вариации (подпотоки) и исключительные условия (альтернативные потоки)).
- Фрагмент **внешне наблюдаемых функций** (отличных от внутренних функций).
- **Ортогональный** фрагмент функциональных возможностей (прецеденты могут при выполнении совместно использовать объекты, но выполнение каждого прецедента независимо от других прецедентов).
- Фрагмент функциональных возможностей, **инициируемый субъектом**. Будучи инициирован, прецедент может взаимодействовать с другими субъектами. При этом возможно, что субъект окажется только на принимающем конце прецедента, опосредованно инициированного другим субъектом.
- Фрагмент функциональных возможностей, который предоставляет субъекту **ощутимый полезный результат** (и этот результат достигается в пределах одного прецедента).

Между субъектами и прецедентами - основными компонентами диаграммы прецедентов - могут существовать различные отношения, которые описывают взаимодействие экземпляров одних субъектов и прецедентов с экземплярами других субъектов и прецедентов. В языке UML имеется несколько стандартных видов отношений между субъектами и прецедентами:

*Отношение ассоциации (association)* - **определяет наличие канала связи между экземплярами субъекта и прецедента (или между экземплярами двух субъектов)**. Обозначается сплошной линией, возможно наличие стрелки и указание мощности связи.

*Отношение расширения (extend)* - определяет взаимосвязь экземпляров отдельного прецедента с более общим прецедентом, свойства которого определяются на основе способа совместного объединения данных экземпляров. Обозначается пунктирной линией со стрелкой, направленной от того прецедента, который является расширением для исходного прецедента, и помечается ключевым словом "extend" ("расширяет").

*Отношение включения (include)* - указывает, что некоторое заданное поведение для одного прецедента включает в качестве составного компонента поведение другого прецедента. Данное отношение является направленным бинарным отношением в том смысле, что пара экземпляров прецедентов всегда упорядочена в отношении включения. Обозначается пунктирной линией со стрелкой, направленной от базового прецедента к включаемому, и помечается ключевым словом "include" ("включает").

*Отношение обобщения (generalization)* - служит для указания того факта, что некоторый прецедент А может быть обобщен до прецедента В. В этом случае прецедент А будет являться специализацией прецедента В. При этом В называется предком или родителем по отношению к А, а прецедент А - потомком по отношению к прецеденту В. Следует подчеркнуть, что потомок наследует все свойства и поведение своего родителя, а также может быть дополнен новыми свойствами и особенностями поведения. Графически данное отношение обозначается сплошной линией со стрелкой в форме незакрашенного треугольника, которая указывает на родительский прецедент.

#### **Пример. Магазин видеопродукции.**

Магазин продает видеокассеты, DVD-диски, аудио-кассеты, CD-диски и т.д. , а также предлагает широкой публике прокат видеокассет и DVD-дисков.

Товары поставляются несколькими поставщиками. Каждая партия товара предварительно заказывается магазином у некоторого поставщика и доставляется после оплаты счета. Вновь поступивший товар маркируется, заносится в базу данных и затем распределяется в торговый зал или прокат.

Видеоносители выдаются в прокат на срок от 1 до 7 дней. При прокате с клиента взимается залоговая стоимость видеоносителя. При возврате видеоносителя возвращается залоговая стоимость минус сумма за прокат. Если возврат задержан менее чем на 2 дня, взимается штраф в размере суммы за прокат за 1 день\* кол-во дней задержки. При задержке возврата более чем на 2 дня - залоговая сумма не возвращается. Клиент может взять одновременно до 4 видеоносителей (прокат-заказ ). На каждый видеоноситель оформляется квитанция.

Клиенты могут стать членами видео-клуба и получить пластиковые карточки. С членов клуба не берется залог (за исключением случая описанного ниже), устанавливается скидка на ставку проката и покупку товаров. Члены клуба могут делать предварительные заказы на подбор видеоматериалов для проката или покупки.

Каждый член клуба имеет некоторый статус. Первоначально - "новичок". При возврате в срок 5 прокат-заказов , статус меняется на "надежный". При задержке хотя бы одного видеоносителя более чем на 2 дня , статус "новичок" или "надежный" меняется на "ненадежный" и клиенту высылается предупреждение. При повторном нарушении правил статус меняется на "нарушитель". Члены клуба со статусом "надежный" могут брать до 8 видеоносителей одновременно, все остальные - 4. С членов клуба со статусом "нарушитель" берется залоговая сумма.

Клиенты при покупке товара или получении видеоносителя в прокат могут расплачиваться наличными или кредитной картой.

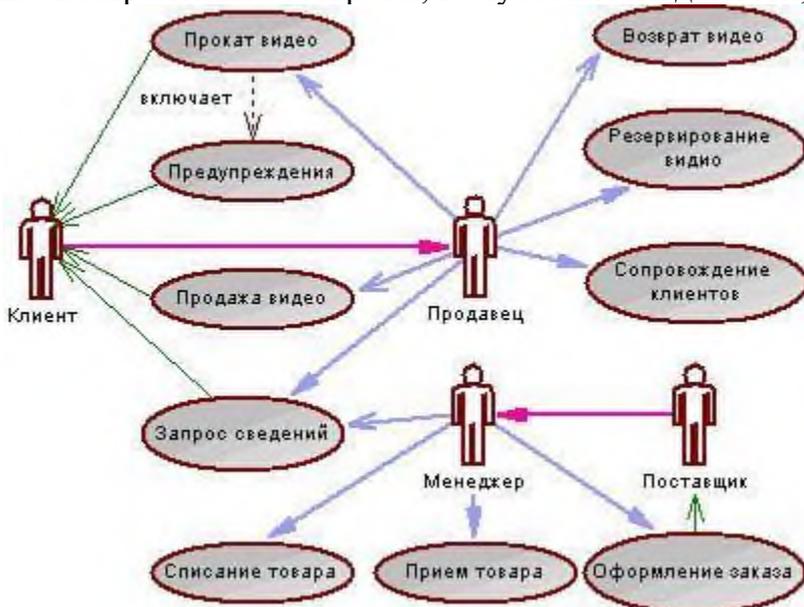
Прокатные видеоносители через определенное количество дней проката списываются и утилизируются по акту. Списываются также товары и прокатные видеоносители, у которых обнаружился брак.

На рисунке ниже приведена диаграмма прецедентов для рассматриваемого примера. В этом примере можно выделить следующие субъекты и соответствующие им прецеденты:

- **Менеджер** - изучает рынок видеопродукции, анализирует продажи (прецедент "Запрос сведений"), работает с поставщиками: составляет заявки на поставки товара (прецедент "Оформление заказа" ), оплачивает и принимает товар (прецедент "Прием товара"), списывает товар (прецедент "Списание товара").
- **Продавец** - работает с клиентами: продает товар (прецедент "Продажа видео"), оформляет членство в клубе (прецедент "Сопровождение клиентов"), резервирует (прецедент "Резервирование видео"), выдает в прокат (прецедент "Прокат видео") и принимает назад видеоносители (прецедент "Возврат видео"), отвечает на вопросы клиента (прецедент "Запрос сведений").
- **Поставщик** - оформляет документы для оплаты товара (прецедент "Оформление заказа"), поставляет товар (прецедент "Прием товара")
- **Клиент** - покупает (прецедент "Продажа видео"), берет на прокат и возвращает видеоносители (прецеденты "Прокат видео" и "Возврат видео"), вступает в клуб (прецедент "Сопровождение клиентов"), задает вопросы (прецедент "Запрос сведений").

Последние два субъекта **Поставщик** и **Клиент** не будут иметь непосредственного доступа к разрабатываемой системе (второстепенные субъекты), однако именно они являются основным источником событий, инициализирующих прецеденты, и получателями результата работы прецедентов

От прецедента "Прокат видео" к прецеденту "Предупреждения" установлено отношение включения на том основании, что каждый выданный видеоноситель должен быть проверен на своевременный возврат и, в случае необходимости, выдано предупреждение клиенту.



Дальнейшее развитие модели поведения системы предполагает спецификацию прецедентов. Для этого традиционно используют два способа. Первый - описание с помощью текстового документа. Такой документ описывает, что должна делать система, когда субъект инициировал прецедент. Типичное описание содержит следующие разделы:

- Краткое описание
- Участвующие субъекты
- Предусловия, необходимые для инициирования прецедента
- Поток событий (основной и, возможно, подпотоки, альтернативный)
- Постусловия, определяющие состояние системы, по достижении которого прецедент завершается.

Описательная спецификация прецедента "Прокат видео"

Раздел	Описание
Краткое описание	Клиент желает взять на прокат видеокассету или диск, которые

	<p>снимаются с полки магазина или были предварительно зарезервированы клиентом. При условии, что у клиента нет невозвращенных в срок видеоносителей, сразу после внесения платы фильм выдается напрокат. Если невозвращенные в срок видеоносители есть, клиенту выдается напоминание о просроченном возврате.</p>
<p>Субъекты Предусловия</p>	<p>Продавец, Клиент. В наличии имеются видеокассеты или диски, которые можно взять напрокат. У клиентов есть клубные карточки. Устройство сканирования работает правильно. Работники за прилавком знают, как обращаться с системой.</p>
<p>Основной поток</p>	<p>Клиент может назвать номер заказа или взять видеоноситель с полки. Видеоноситель и членская карточка сканируются и продавцу не сообщается никаких сведений о задержках, так, что он не задает клиенту соответствующих вопросов. Если клиент имеет статус &lt;надежный&gt;, он может взять до 8 видеоносителей, во всех остальных случаях - до 4-х. Если статус клиента определен как &lt;нарушитель&gt;, его просят внести задаток. Клиент расплачивается наличными или кредитной картой. После получения суммы, информация о наличии фильмов обновляется и видеоносители передаются клиенту вместе с квитанциями на прокат. О прокате каждого видеоносителя делается отдельная запись с указанием идентификационного номера клиента, даты проката, даты возврата, идентификационного номера продавца, полученной суммы.</p>
<p>Альтернативный поток</p>	<p>Прецедент генерирует предупреждения о просроченном возврате клиенту, если видеофильм не был возвращен в течение двух дней по истечении даты возврата и изменяет статус клиента на &lt;ненадежный&gt; (первое нарушение) или &lt;нарушитель&gt; (повторное нарушение). У клиента нет членской карточки. В этом случае прецедент &lt;Сопровождение клиента&gt; может быть активизирован для выдачи новой карточки. Видеофильмы не выдаются, поскольку у клиента есть невозвращенные в срок видеоносители. Попытка взять на прокат слишком много видеоносителей. Видеоноситель или кредитная карта не могут быть отсканированы из-за их повреждения</p>
<p>Постусловия</p>	<p>У клиента не хватило наличных или платеж по кредитной карте отклонен. Видеофильмы сданы напрокат, и база данных соответствующим образом обновлена</p>

**Диаграммы деятельности** (Activitydiagram), называемые также диаграммами активности или диаграммами видов деятельности, были введены в язык UML сравнительно недавно. Диаграмма деятельности - это, по существу, блок-схема, которая показывает, как поток управления переходит от одной деятельности к другой, при этом внимание фиксируется на результате деятельности. Результат может привести к изменению состояния системы или возвращению некоторого значения.

Диаграмма деятельности отличается от традиционной блок-схемы

- более высоким уровнем абстракции;
- возможностью представления с помощью диаграмм деятельности управления параллельными потоками наряду с последовательным управлением.

Основными направлениями использования диаграмм деятельности являются

- визуализация особенностей реализации операций классов;
- отображение внутрисистемной точки зрения на прецедент.

В последнем случае диаграммы деятельности применяют для описания шагов, которые должна предпринять система после того, как инициирован прецедент

Разработка диаграммы деятельности преследует цели:

- детализировать особенности алгоритмической и логической реализации выполняемых системой операций и прецедентов;
- выделить последовательные и параллельные потоки управления;
- подготовить детальную документацию для взаимодействия разработчиков системы с ее заказчиками и проектировщиками.

Графически диаграмма деятельности представляется в форме графа деятельности, вершинами которого являются состояния действия или состояния деятельности, а дугами - переходы от одного состояния действия/деятельности к другому. Каждая диаграмма деятельности должна иметь единственное начальное и единственное конечное состояния (на практике иногда можно видеть несколько конечных состояний на одной диаграмме, но это одно и то же состояние, изображенное несколько раз для лучшей читабельности диаграммы). Саму диаграмму деятельности принято располагать таким образом, чтобы действия следовали сверху вниз. В этом случае начальное состояние будет изображаться в верхней части диаграммы, а конечное - в ее нижней части.

Рассмотрим основные элементы диаграммы деятельности.

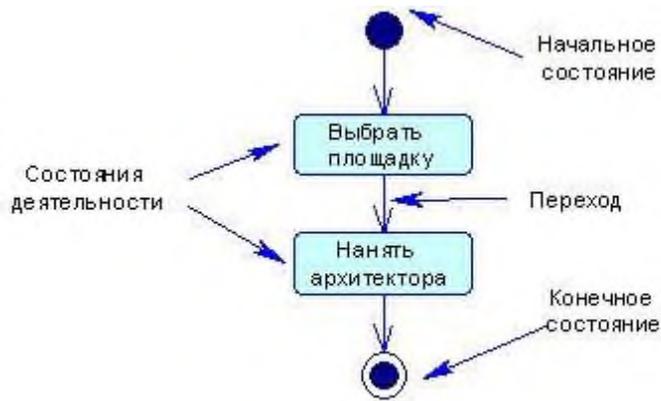
*Состояние деятельности (Activity, Process)*- это **продолжающийся во времени неатомарный шаг вычислений в автомате**. Состояния деятельности могут быть подвергнуты дальнейшей декомпозиции, вследствие чего выполняемую деятельность можно представить с помощью других диаграмм деятельности. Состояния деятельности не являются атомарными, то есть могут быть прерваны. Предполагается, что для их завершения требуется заметное время.

*Состояния действия (actionstate)* - **состояние, которое представляет вычисление атомарного действия**, как правило - вызов операции. Состояния действия не могут быть подвергнуты декомпозиции. Они атомарны, то есть внутри них могут происходить различные события, но выполняемая в состоянии действия работа не может быть прервана. И наконец, обычно предполагается, что длительность одного состояния действия занимает неощутимо малое время. Действие может заключаться в вызове другой операции, посылке сигнала, создании или уничтожении объекта либо в простом вычислении - скажем, значения выражения.

Состояния деятельности и состояния действия имеют одинаковое стандартное графическое обозначение - прямоугольник с закругленными краями. Внутри такого символа записывают произвольное выражение (action-expression), которое должно быть уникальным в пределах одной диаграммы деятельности.

*Начальное и конечное состояния* на диаграммах деятельности изображаются как закрашенный кружок и закрашенный кружок внутри окружности, соответственно.

Ниже приведен пример диаграммы деятельности:

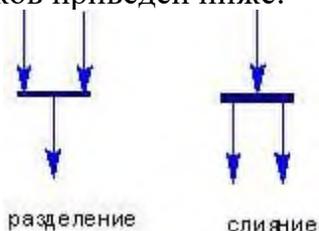


< P >

*Переход (Transitions)* - отношение между двумя состояниями, показывающее, что объект, находящийся в первом состоянии, должен выполнить некоторые действия и перейти во второе состояние. Когда действие или деятельность в некотором состоянии завершается, поток управления сразу переходит в следующее состояние действия или деятельности. Для описания этого потока и используются переходы, показывающие путь из одного состояния действия или деятельности в другое. В UML переход представляется простой линией со стрелкой.

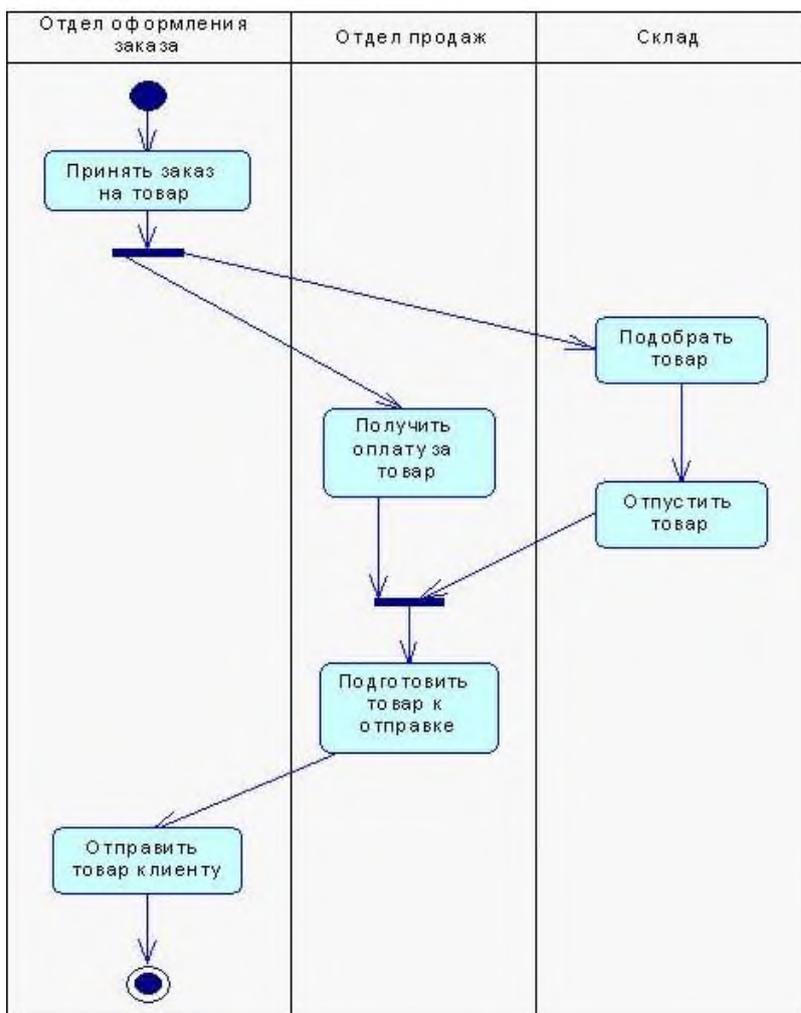
*Ветвления.* Простые последовательные переходы встречаются наиболее часто, но их одних недостаточно для моделирования любого потока управления. Как и в блок-схему, в диаграмму деятельности может быть включено *ветвление или множественный переход со сторожевыми условиями*. Ветвление описывает различные пути выполнения в зависимости от значения некоторого булевского выражения. Графически точка ветвления представляется ромбом. В точку ветвления может входить ровно один переход, а выходить - два или более. Для каждого исходящего перехода задается булевское выражение, которое вычисляется только один раз при входе в точку ветвления. Ни для каких двух исходящих переходов сторожевые условия не должны одновременно принимать значение "истина", иначе поток управления окажется неоднозначным. Но эти условия должны покрывать все возможные варианты, иначе поток остановится.

*Разделения и слияния.* Простые и ветвящиеся последовательные переходы в диаграммах деятельности используются чаще всего. Однако часто возникает потребность изображения параллельных потоков, и это особенно характерно для моделирования бизнес-процессов. В UML для обозначения разделения и слияния таких параллельных потоков выполнения используется синхронизационная черта, которая рисуется в виде жирной вертикальной или горизонтальной линии. При этом *разделение (concurrentfork)* имеет один входящий переход и несколько выходящих, *слияние (concurrentjoin)*, наоборот, имеет несколько входящих переходов и один выходящий. Пример разделения и слияния потоков приведен ниже:

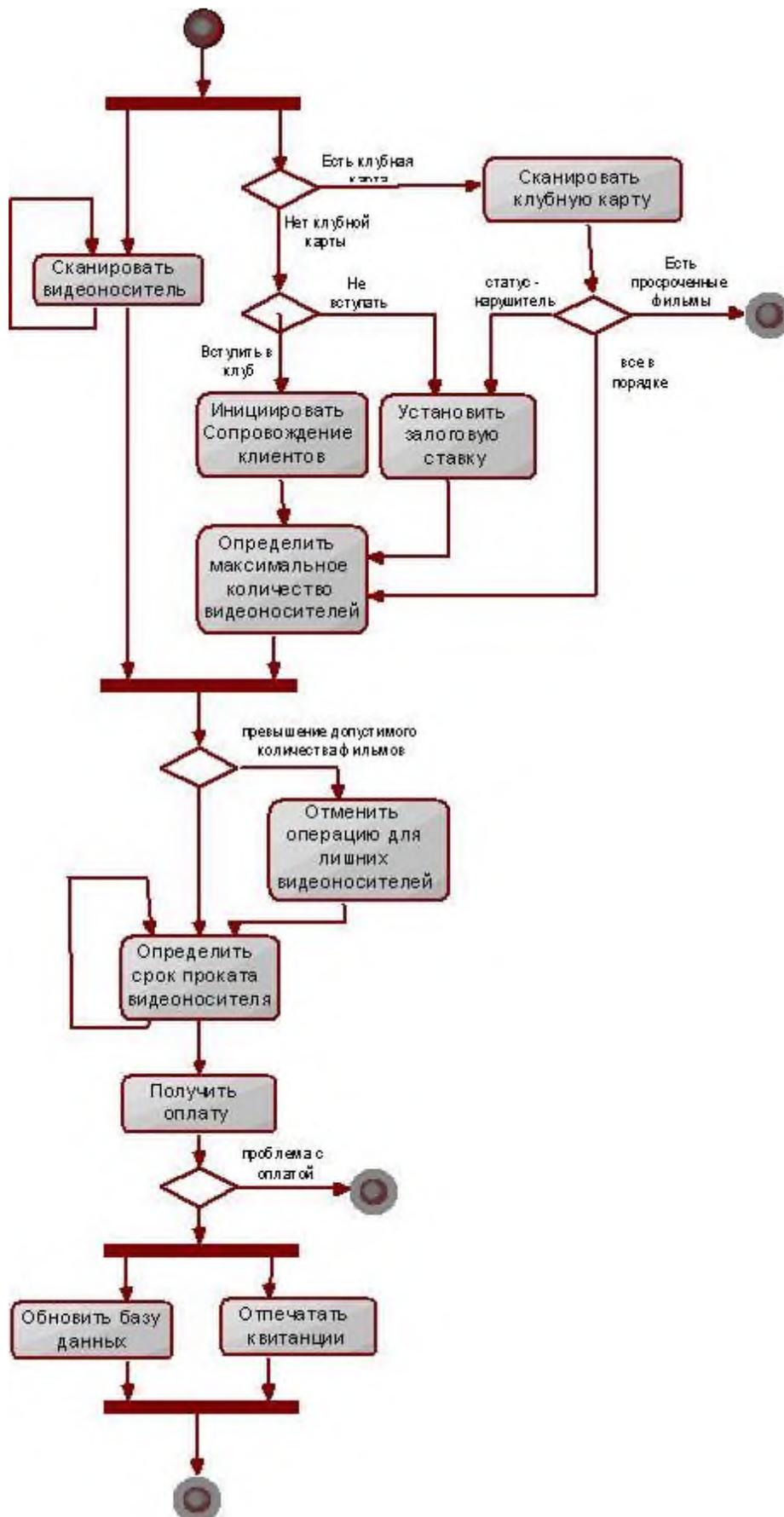


*Дорожки.* При моделировании течения бизнес-процессов иногда бывает полезно разбить состояния деятельности на диаграммах деятельности на группы, каждая из которых представляет отдел компании, отвечающий за ту или иную работу. В UML такие группы называются дорожками (Swimlanes), поскольку визуально каждая группа отделяется от соседних вертикальной чертой, как плавательные дорожки в бассейне (см. рис. 3). Дорожки - это разновидность пакетов, описывающие связанную совокупность

работ.Каждой присутствующей на диаграмме дорожке присваивается уникальное имя. Никакой глубокой семантики дорожка не несет, разве что может отражать некоторую сущность реального мира. Каждая дорожка представляет сферу ответственности за часть всей работы, изображенной на диаграмме. На диаграмме деятельности, разбитой на дорожки, каждая деятельность принадлежит ровно одной дорожке, но переходы могут пересекать границы дорожек.Имеется некоторая связь между дорожками и параллельными потоками выполнения. Концептуально деятельность внутри каждой дорожки обычно - но не всегда - рассматривается отдельно от деятельности в соседних дорожках. Это разумно, поскольку в реальном мире подразделения организации, представленные дорожками, как правило, независимы и функционируют параллельно.



На следующем рисунке приведена диаграмма деятельности прецедента "Прокат видео" для рассмотренного в лабораторной работе © 4 примера "Магазин видеопродукции"



<P>

Результатом выполнения данной работы будет диаграммы прецедентов и деятельности по заданию индивидуального проекта, выполненная в программе MSVisio.

**Итог работы:** файл-отчет.

## Практическая работа №18

Разработка фрагмента информационной системы.

**Цель работы:** закрепить теоретические знания по теме 3.2, разработать фрагмент информационной системы с использованием системы управления базами данных.

- Задание:**
- Изучить теоретический материал;
  - Разработать фрагмент информационной системы по заданию индивидуального проекта;
  - В проект включить реализацию всех автоматизируемых бизнес-процессов в соответствии с предметной областью;
  - Разработать базу данных по фрагменту информационной системы в программе MSAccess;
  - Зафиксировать информацию в файле для отчета.

### Теоретический материал.

Access является наиболее сложной программой из всех офисных приложений MicrosoftOffice. Чтобы начать работу с этой программой, вначале необходимо создать структуру базы данных, подробно ее описать, а затем создать различные формы.

ACCESS – это реляционная СУБД. Это означает, что с ее помощью можно работать одновременно с несколькими таблицами базы данных, эти таблицы между собой связаны. Таблицу ACCESS можно связать с данными, хранящимися на другом компьютере. Данные ACCESS очень просто комбинировать с данными EXCEL, WORD и другими программами Office.

Access во многом похож на Excel. Основное различие между таблицей БД и электронной таблицей – в системе адресации: в электронной таблице адресуется каждая ячейка, а в таблице БД – только поля текущей записи.

### База данных состоит из следующих компонентов:

**Таблицы** – основные объекты базы данных (БД). В них хранятся данные. Реляционная база данных может иметь много взаимосвязанных таблиц. Сведения по разным вопросам следует хранить в разных таблицах. В БД столбцы называются *полями*, а строки – *записями*. Для работы таблицу необходимо открыть. Перед окончанием работы ее следует закрыть, предварительно сохранив все изменения.

**Запрос** – это средство, с помощью которого извлекается из базы данных информация, отвечающая определенным критериям. Результаты запроса представляют не все записи из таблицы, а только те, которые удовлетворяют запросу.

**Формы** – Обеспечивают более наглядную работу с таблицами, с помощью форм в базу вводят новые данные или просматривают имеющиеся.

**Отчеты** – средство представления данных таблиц. Отчеты могут быть оформлены надлежащим образом и распечатаны в том виде, в котором требуется пользователю.

### Этапы проектирование базы данных:

#### 1. Анализ требований к базе данных.

На основе пункта 5 из работы №1, пункта 3 из работы №2 и пункта 2 из работы №3 необходимо определить какую информацию необходимо хранить в базе данных.

Здесь необходимо перечислить информационные объекты, с которыми работает система (клиенты, товары, заказы, поставщики, сотрудники, и т.д.). Кроме того, необходимо перечислить основные типы запросы к базе данных.

#### 2. Концептуальное проектирование базы данных (разработка инфологической модели базы данных).

Необходимо:

1) определить сущности, с которыми будет работать информационная система. Для каждой сущности задать уникальное имя, список атрибутов, идентификатор;

2) определить связи между сущностями. Для каждой связи определить степень связи и класс принадлежности;

3) определить внешние ограничения целостности.

Результатом этого этапа должна стать инфологическая модель базы данных.

**3. Выбор системы управления базами данных.**

Проанализировав требования к системе необходимо указать, какая СУБД будет выбрана для реализации базы данных и обосновать этот выбор.

**4. Датологическое (логическое) проектирование.**

Необходимо нормализовать (минимум до третьей нормальной формы) инфологическую модель. Можно использовать как метод декомпозиции отношений, так и метод синтеза отношений.

Далее необходимо определить ограничения целостности:

- внешние;
- внутренние;
- специально конструируемые.

Результатом этого этапа должна стать нормализованная логическая модель, выполненная в соответствии с требованиями выбранной СУБД.

**5. Физическое проектирование.**

На этом этапе необходимо:

- 1) реализовать базу данных в выбранной СУБД;
- 2) определить и перечислить все основные транзакции базы данных
- 3) провести денормализацию данных с целью увеличения производительности процесса обработки транзакций. Здесь необходимо указать используемые методы денормализации и представить итоговую схему базы данных.

Результатом выполнения данной работы будет являться база данных для индивидуального проекта, спроектированная на основе анализа предметной области, выделенных бизнес-процессов и итоговых диаграмм прецедентов и деятельности.

База данных должна содержать 3 запроса с условием (используя различные операторы сравнения и логические операторы: =, <, >, <>, Between, In, Like, And, Or, Not ), запрос с вычисляемым полем, запрос с параметром, перекрёстный запрос, кнопочную форму и отчеты по всем запросам

**Итог работы:** файл-отчет.

### **Практическая работа №19**

Использование языка запросов баз данных для манипулирования данными.

**Цель работы:** закрепить теоретические знания по теме 3.2, научиться применять операторы языка запросов баз данных для манипулирования данными.

- Задание:**
- Изучить теоретический материал;
  - Выполнить задания из файла «Базы данных. Эффективная работа в Access. Использование языка структурированных запросов»;
  - Изменить данные в базе данных индивидуального проекта с использованием операторов манипулирования данными;
  - Зафиксировать информацию в файле для отчета.

**Теоретический материал.**

#### **Описание основных операторов SQL**

SQL состоит из набора команд манипулирования данными в реляционной базе данных, которые позволяют создавать объекты реляционной базы данных,

модифицировать данные в таблицах (вставлять, удалять, исправлять), изменять *схемы отношений* базы данных, выполнять вычисления над данными, делать выборки из базы данных, поддерживать безопасность и целостность данных.

Весь набор команд SQL можно разбить на следующие группы:

- команды определения данных (*DDL- DataDefininionLanguage*);
- команды манипулирования данными (*DML-DataManipulationLanguage*);
- команды выборки данных (*DQL - DataQueryLanguage*);
- команды управления транзакциями;
- команды управления данными.

При выполнении каждая команда SQL проходит четыре фазы обработки:

- фаза *синтаксического разбора*, которая включает проверку синтаксиса команды, проверку имен таблиц и колонок в базе данных, а также подготовку исходных данных для оптимизатора;
- фаза оптимизации, которая включает подстановку действительных имен таблиц и колонок базы данных в представление, идентификацию возможных вариантов выполнения команды, определение стоимости выполнения каждого варианта, выбор наилучшего варианта на основе внутренней статистики;
- фаза генерации исполняемого кода, которая включает построение выполняемого кода команды;
- фаза выполнения команды, которая включает выполнение кода команды.

В настоящее время оптимизатор является составной частью любой промышленной реализации SQL. Работа оптимизатора основана на сборе статистики о выполняемых командах и выполнении эквивалентных алгебраических преобразований с отношениями базы данных. Такая статистика сохраняется в системном каталоге базы данных. Системный каталог является словарем данных для каждой базы данных и содержит информацию о таблицах, представлениях, индексах, колонках, пользователях и их *привилегиях доступа*. Каждая база данных имеет свой системный каталог, который представляет совокупность предопределенных таблиц базы данных.

Команды манипулирования данными

DELETE	Удаляет одну или более строк из таблицы базы данных
INSERT	Вставляет одну или более строк в таблицу базы данных
UPDATE	Обновляет значения колонок в таблице базы данных

Следует помнить, что SQL является единственным средством общения всех категорий пользователей с реляционными базами данных.

Результатом выполнения данной работы будет являться база данных для индивидуального проекта.

**Итог работы:** файл-отчет.

## Практическая работа №20

Определение ограничения целостности данных.

**Цель работы:** закрепить теоретические знания по теме 3.2, закрепить знания по формированию первичных и внешних ключей для обеспечения целостности данных информационной системы.

- Задание:**
- Изучить теоретический материал;
  - Настроить систему первичных и внешних ключей для таблиц базы данных индивидуального проекта;
  - Определить значение свойства "Условие на значение" для каждой таблицы;

- Зафиксировать информацию в файле для отчета.

### **Теоретический материал.**

ограничения целостности можно определить как специальные средства в базах данных, главное назначение которых - не дать попасть в базу недопустимым данным (например, предупредить ошибки пользователей при вводе данных).

Все ограничения целостности можно разделить на три большие категории:

- первая категория - средства обеспечения доменной целостности. Они отвечают за то, чтобы в соответствующем поле базы данных были допустимые значения. Например, фамилия, как правило, должна состоять из букв, а почтовый индекс - из цифр. В базах данных такая целостность обычно обеспечивается условиями на значение, запретом пустых значений, триггерами и хранимыми процедурами, а также ключами;
- вторая категория - сущностная целостность. Главная задача здесь - сделать так, чтобы данные об одной сущности не попали в базу данных два раза. Обеспечивается ограничением уникальности и первичным ключом;
- третья категория - ссылочная целостность, обеспечивается системой первичных и внешних ключей. Например, при помощи этих средств можно гарантировать, что у нас не будет заказов, оформленных на покупателей, которых нет в базе данных.

Еще две большие категории, на которые можно поделить средства обеспечения целостности - средства декларативного и процедурного характера. Средства декларативного характера создаются как составные части объектов при их определении в базе данных (например, условие на значение при определении таблицы в базе данных). Средства процедурного характера (триггеры и хранимые процедуры) реализуются как отдельные программные модули. В общем случае декларативные ограничения менее функциональны, но более экономны с точки зрения ресурсов и наоборот.

Надо сказать, что наличие развитой системы ограничений целостности во многом определяет зрелость базы данных. Обычно проще сразу позаботиться о том, чтобы в базу данных не попадали неверные значения, чем потом их убирать из базы данных.

Средства обеспечения сущностной целостности очевидны: это **-первичные ключи и ограничения уникальности**. Первичный ключ можно определять при создании таблицы или потом при помощи команды `SQLALTER TABLE`:

```
ALTER TABLE doc_exe  
ADD column_b INT IDENTITY  
CONSTRAINT column_b_pk PRIMARY KEY
```

В Microsoft Access средств реализации целостности на уровне ядра намного меньше. Практически это - только система первичных и внешних ключей плюс условия на проверку в свойствах столбцов. На практике получается так, что больших требований к ядру при работе с Access обычно предъявлять не приходится - все проверки реализуются на клиенте, например, средствами VBA.

Физически настройка параметра производится в режиме конструктора для таблицы. Для поля необходимо просто определить значение свойства "Условие на значение". Удобнее всего это сделать при помощи построителя выражений (кнопка для его вызова появляется в правой части соответствующего поля).

Результатом выполнения данной работы будет являться база данных для индивидуального проекта. База данных должна быть приведена к 3 НФ.

**Итог работы:** файл-отчет.

## Практическая работа №21

Построение архитектурной схемы организации (модель «как есть»).

**Цель работы:** закрепить теоретические знания по теме 3.2, построить схему организации индивидуального проекта по модели «как есть».

- Задание:**
- Изучить теоретический материал;
  - С учетом разработанной информационной системы по индивидуальному проекту построить новую модель «как есть» для организации;
  - Сравнить изменения с исходной моделью организации;
  - Сделать вывод о преимуществах внедрения информационной системы в организацию;
  - Зафиксировать информацию в файле для отчета.

### Теоретический материал.

Данная модель позволяет систематизировать протекающие в данный момент процессы, а также используемые информационные объекты. На основе этого выявляются узкие места в организации и взаимодействии бизнес-процессов, определяется необходимость тех или иных изменений в существующей структуре.

Такую модель часто называют функциональной и выполняют с использованием различных графических нотаций и case-средств. На этапе построения модели AS-IS важным считается строить максимально приближенную к действительности модель, основанную на реальных потоках процессов, а не на их идеализированном представлении.

Проектирование информационных систем и управление процессами подразумевает построение модели AS-IS и дальнейший переход к модели TO-BE, что является залогом автоматизации "правильных", усовершенствованных процессов.

Процесс моделирования какой-либо системы в IDEF0 начинается с определения контекста, т. е. наиболее абстрактного уровня описания системы в целом. В контекст входит определение субъекта моделирования, цели и точки зрения на модель. Под субъектом понимается сама система, при этом необходимо точно установить, что входит в систему, а что лежит за ее пределами, другими словами, определить компоненты системы и внешние воздействия.

На определение субъекта системы существенно влияет позиция, с которой рассматривается система, и цель моделирования - вопросы, на которые построенная модель должна дать ответ. Другими словами, первоначально необходимо определить область моделирования. Описание области как системы в целом, так и ее компонентов является основой построения модели. Цель моделирования. Модель не может быть построена без четко сформулированной цели.

Обычно сначала строится модель существующей организации работы - AS-IS (как есть). На основе модели AS-IS достигается консенсус между различными единицами бизнеса по тому, "кто что сделал" и что каждая единица бизнеса добавляет в процесс. Модель AS-IS позволяет выяснить, "что мы делаем сегодня" перед тем, как перейти на то, "что мы будем делать завтра".

Анализ функциональной модели позволяет понять, где находятся наиболее слабые места, в чем будут состоять преимущества новых бизнес-процессов и насколько глубоким изменениям подвергнется существующая структура организации бизнеса.

Детализация бизнес-процессов позволяет выявить недостатки организации даже там, где функциональность на первый взгляд кажется очевидной.

Признаками неэффективной деятельности могут быть бесполезные, неуправляемые и дублирующиеся работы, неэффективный документооборот оказывается в нужном месте в нужное время, отсутствие обратных связей по управлению и т. д.

Найденные в модели AS-IS недостатки можно исправить при создании модели TO-BE (как будет) - модели новой организации бизнес-процессов. Модель нужна TO-BE для анализа альтернативных/лучших путей выполнения работы и документирования того, как компания будет делать бизнес в будущем.

Результатом выполнения данной работы будет построение архитектурной схемы организации с учетом внедрения информационной системы по индивидуальному проекту.

**Итог работы:** файл-отчет.

## Практическая работа №22

Оформление программной и технической документации с использованием стандартов оформления программной документации.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомление с процедурой составления технологической документации к разработанному программному продукту.

- Задание:**
- Изучить теоретический материал;
  - Оформить по шаблонам необходимую программную и техническую документацию для информационной системы по индивидуальному проекту;
  - Сделать вывод о целесообразности оформления и ведения документации;
  - Зафиксировать информацию в файле для отчета.

### Теоретический материал.

*Документация по сопровождению ПП (system documentation)* описывает ПП с точки зрения ее разработки.

В случае необходимости модернизации ПП к этой работе привлекается специальная команда разработчиков-сопроводителей.

Этой команде придется иметь дело с такой же документацией, которая определяла деятельность команды первоначальных (основных) разработчиков ПП.

Команда разработчиков-сопроводителей должна будет изучать технологическую документацию, чтобы понять строение и процесс разработки модернизируемого ПП, и внести необходимые изменения, повторяя в значительной степени технологические процессы, спомощью которых создавался первоначальный ПП.

Документация по сопровождению ПП можно разбить на две группы:

(1) документация, определяющая строение программ и структур данных ПП и технологию их разработки;

(2) документацию, помогающую вносить изменения в ПП.

Документы установления достоверности ПП включают, прежде всего, документацию по тестированию (схема тестирования и описание комплекта тестов), но могут включать и результаты других видов проверки ПС, например, доказательства свойств программ.

Документация второй группы содержит руководство по сопровождению ПП (*system maintenance guide*), которое описывает известные проблемы вместе с ПП, описывает, какие части системы являются аппаратно- и программно-зависимыми, и как развитие ПП принято в расчет в его строении (конструкции).

Результатом выполнения данной работы является технологическая документация к разработанной информационной системе по индивидуальному проекту, оформленная в соответствии со стандартами ГОСТ.

**Итог работы:** файл-отчет.

### Практическая работа №23

Использование нормативных документов при разработке информационной системы.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомление с нормативными документами при разработке информационной системы.

- Задание:**
- Изучить теоретический материал;
  - Найти в сети Интернет сведения о стандартах ISO/IEC (ИСО/МЭК) в области разработки и документирования программных средств;
  - Сделать вывод о применении нормативных документов при разработке ИС;
  - Зафиксировать информацию в файле для отчета.

#### Теоретический материал.

При разработке промышленных информационных систем одним из ключевых требований является соответствие разрабатываемых систем и их компонентов, а также документации действующим государственным и международным стандартам.

Согласно Закону Российской Федерации «О техническом регулировании» стандарт – документ, в котором в целях добровольного многократного использования устанавливаются характеристики продукции, правила осуществления и характеристики процессов производства, эксплуатации, хранения, перевозки, реализации и утилизации, выполнения работ или оказания услуг. Стандарт также может содержать требования к терминологии, символике, упаковке, маркировке и правилам ее нанесения.

Деятельность по установлению правил и характеристик, направленная на достижение упорядоченности в сферах производства и обращения продукции, а также повышение ее конкурентоспособности, работ или услуг, называется стандартизацией.

Результатом выполнения данной работы является анализ нормативной документации при разработке информационной системы.

**Итог работы:** файл-отчет.

### Практическая работа №24

Тестирование и контрольный расчет программы.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомление с процедурой тестирования,.

- Задание:**
- Изучить теоретический материал;
  - Провести тестирование системы по моделям белого ящика, черного ящика;
  - Оформить документацию по результатам тестирования к разработанному программному продукту;
  - Сделать вывод о преимуществах внедрения информационной системы в организацию;
  - Зафиксировать информацию в файле для отчета.

#### Теоретический материал.

За функциональные и исполнительные тесты несут ответственность разработчики заказчик, он больше влияет на составление тестов испытаний и инсталляции системы.

Для этих целей, как правило, создается служба проверяющих - команда тестировщиков, которая не зависит от штата разработчиков.

Так как тесты и тестовые сценарии являются прямым отражением требований и проекта в целом, перспективы управления конфигурацией системы определяются именно

этой командой. Обнаруживаемые в программе ошибки и изменения в системе отражаются в документации, требованиях, проекте, а также в описаниях входных и выходных данных или в других разрабатываемых артефактах. Вносимые изменения в процессе разработки приводят к модификации тестовых сценариев или в большей части к изменению планов тестирования. Специалисты по управлению конфигурацией учитывают эти изменения и координируют составление тестов.

Для проведения тестирования создается план (*TestPlan*), в котором описываются стратегии, ресурсы и график тестирования отдельных компонентов и системы в целом. В плане отмечаются работы для разных членов команды, которые выполняют определенные роли в этом процессе. План включает также определение роли тестов в каждом процессе, степень покрытия программы тестами и процент тестов, которые выполняются со специальными данными.

Тестовые инженеры создают множество тестовых сценариев (*TestCases*), каждый из которых проверяет результат взаимодействия между актором и системой на основе пред- и постусловий использования таких сценариев.

Оценка тестов (*TestEvaluation*) заключается в оценке результатов тестирования, степени покрытия программ сценариями и статуса полученных ошибок.

Все способы тестирования информационных систем объединяются базой данных, где помещаются результаты тестирования системы. В ней содержатся все компоненты, тестовые контрольные данные, результаты тестирования и информация о документировании процесса тестирования.

База данных проекта поддерживается специальными инструментальными средствами типа CASE, которые обеспечивают ведение анализа ПО, сборку данных об их объектах, потоках данных и тому подобное.

При тестировании выполняются разные *виды расчетов* характеристик этого процесса и способы планирования и управления.

1. Расчет продолжительности выполнения функций путем сбора средних показателей скорости выполнения операторов без выполнения программы на машине. Выявляются компоненты, которые требуют большого времени выполнения в реальной среде.
2. Управление выполнением тестирования путем подбора тестов проверки, их выполнения, селекции результатов тестирования и проведения сопоставления с эталонными значениями. Результаты данного процесса отображаются на дисплее, например, в графической форме (пути прохождения по графу программы), в виде диаграмм UML, данных об отказах и ошибках или конкретных значений исходных параметров программы. Эти данные анализируются разработчиками для формулирования выводов о направлениях дальнейшей проверки правильности программы или их завершения.
3. Планирование тестирования предназначено для распределения сроков работ по тестированию, распределения тестировщиков по отдельным видам работ и составления ими тестов проверки системы. Определяются стратегия и пути тестирования. В диалоге запрашиваются данные о реальных значениях процесса выполнения системы, структуре ветвления вершин графа и параметрах циклов. Проверенные циклы, как правило, изымаются из путей выполнения программы. При планировании путей выполнения создаются соответствующие тесты, критерии и входные значения.

Документирование результатов тестирования в соответствии с действующим стандартом ANSI/IEEE 829 включает:

- описание задач, назначение и содержание ПС, а также перечень функций в соответствии с требованиями заказчика;

- технологии разработки системы;
- планов тестирования различных объектов, необходимых ресурсов, соответствующих специалистов для проведения тестирования и технологических способов;
- тестов, контрольных примеров, критериев и ограничений, оценки результатов программного продукта, а также процесса тестирования;
- учета процесса тестирования, составление отчетов об аномальных событиях, отказах и дефектах в итоговом документе системы.

Результатом выполнения данной работы является технологическая документация результатов тестирования к разработанной информационной системе по индивидуальному проекту, оформленная в соответствии со стандартом ANSI/IEEE 829.

**Итог работы:** файл-отчет.

### Практическая работа №25

Составление руководства пользователя к программе.

**Цель работы:** закрепить теоретические знания по теме 3.2, ознакомление с процедурой составления пользовательской (эксплуатационной) документации к информационной системе.

- Задание:**
- Изучить теоретический материал;
  - Составить эксплуатационную документацию к базе данных для индивидуального проекта;
  - Составить руководство пользования информационной системой;
  - Сделать вывод о целесообразности составления руководства пользователя;
  - Зафиксировать информацию в файле для отчета.

#### Теоретический материал.

**Эксплуатационная документация** должна обеспечивать отчуждаемость ПС от их первичных разработчиков, адекватно отражать требуемое внешнее качество и качество в использовании, а также возможность освоения и эффективного применения ПС достаточно квалифицированными специалистами. Она применяется непосредственными пользователями в соответствии с функциональным назначением ПС, а также заказчиками, покупателями и поставщиками программных продуктов. Состав этой документации формируется с использованием части технологических документов с учетом требований заказчиков или потенциальных пользователей ПС.

Содержание эксплуатационных документов должно предотвращать или исключать возможность некорректного использования комплекса программ пользователями за пределами условий эксплуатации, при которых поставщиком гарантируются требуемые и утвержденные характеристики качества функционирования ПС. При формировании эксплуатационных документов ПС, кроме базовых стандартов жизненного цикла могут использоваться ряд ведомственных нормативных документов и фирменных руководств.

Эксплуатационная документация включает в себя:

- Документация администрирования при применении ПС;
- Документация операторов-пользователей при применении программного средства;
- Документация обучения специалистов применению ПС.

**Документация администрирования** при эксплуатации информационной системы должна обеспечивать поддержку первичной инсталляции, штатного функционирования и восстановления программ и данных после сбоев и отказов. Управляющая

деятельность администратора состоит в манипулировании управляемыми объектами должна описываться, анализироваться и регламентироваться совокупностью требований и документов. Для этого необходима полная документация о компонентах информационной системы (компьютерах, сетевых устройствах), которые имеют свои особенности в управлении с помощью специальных программных компонентов, поддерживающих администрирование и управление системой. К основным функциям системы администрирования,

*Документы для которых подлежат разработке и оцениванию, относятся:*

- планирование использования памяти и производительности вычислительной системы в рабочем режиме применения ПС, оперативное управление и распределение ресурсов информационной системы;
- инсталляция и генерация рабочих версий ПС для оперативных пользователей;
- управление и учет внешней среды при выполнении адаптации и реконфигурации конкретного ПС;
- выявление, регистрация и накопление данных о сбоях и дефектах функционирования программ и данных;
- управление средствами защиты информации и санкционированного доступа пользователей, анализ попыток взлома системы защиты, восстановление программ и информации баз данных при искажениях;
- сбор и обобщение статистики о качестве функционирования ПС в составе системы обработки информации.

*Документация операторов-пользователей* должна обеспечивать корректную и квалифицированную эксплуатацию комплекса программ во всем диапазоне его характеристик, предписанных требованиями заказчика и зафиксированных метриками в использовании. Объектами разработки и оценивания являются документы на процедуры и компоненты интерфейса с внешней средой и с пользователями, определяющие инициализацию соответствующих операций, их ход и результаты, а также комфортность их выполнения.

Должно быть предусмотрено достаточное качество идентификации ошибочных действий и ситуаций, а также стандартизированной формы сообщений об ошибках пользователей.

Приобретение, поставка, разработка, функционирование и сопровождение программных средств в значительной степени зависит от квалификации специалистов. Поэтому эксплуатационной документацией обязательно должно поддерживаться эффективное обучение персонала с целью его подготовки к приобретению, поставке, применению и сопровождению программного средства.

Результатом выполнения данной работы является пользовательская документация к разработанной информационной системе по индивидуальному проекту, оформленная в соответствии со стандартами ГОСТ.

**Итог работы:** файл-отчет.

## 4. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРАКТИЧЕСКИХ РАБОТ

### Нормативно-техническая документация

1. ГОСТ 34.601 – 90 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
2. ГОСТ 34.602-89. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Техническое задание на создание автоматизированной системы
3. Стандарт ISO/IEC 12207:2008 «Information Technology — Software Life Cycle Processes» (информационные технологии – жизненный цикл программного обеспечения), ГОСТ Р ИСО/МЭК 12207-2010.

### Основные:

1. Перлова О.Н. Проектирование и разработка информационных систем : учебник для студ. учреждений сред. проф. образования / О.Н.Перлова, О.П.Ляпина, А.В.Гусева. — 2-е изд.,стер. — М. : Издательский центр «Академия», 2018. — 256 с.
2. Федорова Г.Н. Сопровождение информационных систем : учебник для студ. учреждений сред. проф. образования / Г.Н.Федорова. —М. : Издательский центр «Академия», 2018. — 320 с.
3. Советов Б. Я. Методы и средства проектирования информационных систем и технологий : учебник для студ. учреждений высш.образования / Б. Я. Советов, В. А. Дубенецкий, В. В. Цехановский. — М. : Издательский центр «Академия», 2018. —352 с.

### Дополнительные:

1. Гагарина, Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учебное пособие/ Гагарина Л.Г., Киселев Д.В., Федотова Е.Л. – М.: ФОРУМ: ИНФРА-М, 2007.
2. Емельянова, Н.З. Основы построения автоматизированных информационных систем: учебник/ Н.З. Емельянова, Т.Л. Партыка, И.И. Попов – М.: ФОРУМ: ИНФРА-М, 2005.

### Интернет – ресурсы:

1. Образовательный портал: Информационная система «Единое окно доступа к образовательным ресурсам» [Электронный ресурс]. Режим доступа: <http://window.edu.ru>, с регистрацией. – Заглавие с экрана

## 5. ЛИСТ ИЗМЕНЕНИЙ И ДОПОЛНЕНИЙ, ВНЕСЕННЫХ В МЕТОДИЧЕСКИЕ УКАЗАНИЯ

<b>№ изменения, дата внесения, № страницы с изменением</b>	
<b>Было</b>	<b>Стало</b>
<b>Основание:</b>	
<b>Подпись лица, внесшего изменения</b>	