

Министерство образования и науки Республики Татарстан
Государственное автономное профессиональное образовательное учреждение
«Сабинский аграрный колледж»

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПРОФЕССИОНАЛЬНОГО МОДУЛЯ
ПМ.06. СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ**

ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ
ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА ПО СПЕЦИАЛЬНОСТИ

09.02.07 Информационные системы и программирование

Квалификация: Специалист по информационным системам

Фонд оценочных средств рассмотрен и одобрен
цикловой комиссией *Программирования*

Председатель ЦК

Протокол № _____

от « 25 » *апреля* 2022 г.

УТВЕРЖДАЮ

Заместитель директора по ТО
ГАПОУ «Сабинский аграрный колледж»

Ибрагимов Р.М.

Ибрагимов Р.М.

« 25 » *апреля* _____ 2022 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПМ.06 Сопровождение информационных систем для
специальности: 09.02.07 Информационные системы и программирование, квалификация:
специалист по информационным системам

Фонд оценочных средств разработан на основе образования Федерального государственного
образовательного стандарта по специальности среднего профессионального образования
09.02.07 Информационные системы и программирование, утвержденным приказом Министер-
ства образования и науки РФ 09 декабря 2016 № 1548, зарегистрированным в Министерстве
юстиции Российской Федерации 26 декабря 2016 года, регистрационный № 44978, входящим в
укрупнённую группу ТОП-50 09.00.00 Информатика и вычислительная техника.

Разработчик: Мусин Б.Р., преподаватель ГАПОУ «МЦК - КТИТС»

Содержание

1.Паспорт фонда оценочных средств	4
1.1 Общие положения	4
1.2 Результаты освоения профессионального модуля, подлежащие проверке	4
1.3 Формы текущей и промежуточной аттестации	5
1. Комплект оценочных средств	6
2.1. Комплект оценочных средств для входного контроля	6
2.2. Комплект оценочных средств	10
3. Контрольно-оценочные материалы для зачета	17
3.1. Общие положения	17
3.2. Комплект оценочных материалов	18
3.3. Критерии оценивания	41

1. Паспорт комплекта оценочных средств

1.1. Общие положения

Фонд оценочных средств (ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу профессионального модуля ПМ 06 Сопровождение информационных систем

ФОС включает контрольные материалы для проведения текущего контроля и промежуточной аттестации в форме дифференцированного зачета

ФОС разработан на основе ФГОС образовательной программы по специальности 09.02.07 Информационные системы и программирование, квалификация: Специалист по информационным системам

Формой аттестации по профессиональному модулю является экзамен (квалификационный).

1.2. Планируемые результаты освоения профессионального модуля

Результатом освоения профессионального модуля является овладение обучающимися видом профессиональной деятельности:

- Разработка программных модулей программного обеспечения для компьютерных систем.
- Разработка и администрирование баз данных.
- Участие в интеграции программных модулей.
- Выполнение работ по одной или нескольким профессиям рабочих, должностям служащих (приложение к настоящему ФГОС СПО),

в том числе профессиональными (ПК) и общими (ОК) компетенциями:

Код компетенции	Формулировка компетенции	Знания, умения
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам	Умения: распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать задачу и/или проблему и выделять её составные части; определять этапы решения задачи; выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы; составить план действия; определить необходимые ресурсы; владеть актуальными методами работы в профессиональной и смежных сферах; реализовать составленный план; оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)

		<p>Знания: актуальный профессиональный и социальный контекст, в котором приходится работать и жить; основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте; алгоритмы выполнения работ в профессиональной и смежных областях; методы работы в профессиональной и смежных сферах; структуру плана для решения задач; порядок оценки результатов решения задач профессиональной деятельности</p>
ОК 02	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности	<p>Умения: определять задачи для поиска информации; определять необходимые источники информации; планировать процесс поиска; структурировать получаемую информацию; выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска</p> <p>Знания: номенклатура информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации</p>
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие.	<p>Умения: определять актуальность нормативно-правовой документации в профессиональной деятельности; применять современную научную профессиональную терминологию; определять и выстраивать траектории профессионального развития и самообразования</p> <p>Знания: содержание актуальной нормативно-правовой документации; современная научная и профессиональная терминология; возможные траектории профессионального развития и самообразования</p>
ОК 04	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	<p>Умения: организовывать работу коллектива и команды; взаимодействовать с коллегами, руководством, клиентами в ходе профессиональной деятельности</p> <p>Знания: психологические основы деятельности коллектива, психологические особенности личности; основы проектной деятельности</p>
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.	<p>Умения: грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе</p> <p>Знания: особенности социального и культурного контекста; правила оформления документов и построения устных сообщений.</p>
ОК 06	Проявлять гражданско-	<p>Умения: описывать значимость своей специальности</p>

	патриотическую позицию, продемонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.	Знания: сущность гражданско-патриотической позиции, общечеловеческих ценностей; значимость профессиональной деятельности по специальности
ОК 07	Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.	Умения: соблюдать нормы экологической безопасности; определять направления ресурсосбережения в рамках профессиональной деятельности по специальности Знания: правила экологической безопасности при ведении профессиональной деятельности; основные ресурсы, задействованные в профессиональной деятельности; пути обеспечения ресурсосбережения
ОК 08	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.	Умения: использовать физкультурно-оздоровительную деятельность для укрепления здоровья, достижения жизненных и профессиональных целей; применять рациональные приемы двигательных функций в профессиональной деятельности; пользоваться средствами профилактики перенапряжения характерными для данной специальности Знания: роль физической культуры в общекультурном, профессиональном и социальном развитии человека; основы здорового образа жизни; условия профессиональной деятельности и зоны риска физического здоровья для специальности; средства профилактики перенапряжения
ОК 09	Использовать информационные технологии в профессиональной деятельности	Умения: применять средства информационных технологий для решения профессиональных задач; использовать современное программное обеспечение Знания: современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языках.	Умения: понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы; строить простые высказывания о себе и о своей профессиональной деятельности; кратко обосновывать и объяснить свои действия (текущие и планируемые); писать простые связные сообщения на знакомые или интересующие профессиональные темы

		<p>Знания: правила построения простых и сложных предложений на профессиональные темы; основные общеупотребительные глаголы (бытовая и профессиональная лексика); лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности; особенности произношения; правила чтения текстов профессиональной направленности</p>
ОК 11	Планировать предпринимательскую деятельность в профессиональной сфере	<p>Умения: выявлять достоинства и недостатки коммерческой идеи; презентовать идеи открытия собственного дела в профессиональной деятельности; оформлять бизнес-план; рассчитывать размеры выплат по процентным ставкам кредитования; определять инвестиционную привлекательность коммерческих идей в рамках профессиональной деятельности; презентовать бизнес-идею; определять источники финансирования</p> <p>Знание: основы предпринимательской деятельности; основы финансовой грамотности; правила разработки бизнес-планов; порядок выстраивания презентации; кредитные банковские продукты</p>

1.1.2. Перечень профессиональных компетенций

Код	Наименование видов деятельности и профессиональных компетенций
ВД 6	Сопровождение информационных систем
ПК 6.1.	Разрабатывать техническое задание на сопровождение информационной системы
ПК 6.2	Выполнять исправление ошибок в программном коде информационной системы
ПК 6.3	Разрабатывать обучающую документацию для пользователей информационной системы
ПК 6.4	Оценивать качество и надежность функционирования информационной системы в соответствии с критериями технического задания
ПК 6.5	Осуществлять техническое сопровождение, обновление и восстановление данных ИС в соответствии с техническим заданием

1.1.3. В результате освоения профессионального модуля студент должен:

Иметь практический опыт	В инсталляции, настройка и сопровождение информационной системы; выполнении регламентов по обновлению, техническому сопровождению и восстановлению данных информационной системы
уметь	<ul style="list-style-type: none"> - осуществлять настройку информационной системы для пользователя согласно технической документации; - применять основные правила и документы системы сертификации Российской Федерации; - применять основные технологии экспертных систем; разрабатывать обучающие материалы для пользователей по эксплуатации информационных систем; - разрабатывать и внедрять информационную систему - выполнять сравнительный анализ методологий проектирования; - выполнять обслуживание информационной системы в соответствии с пользовательской документацией; - формировать отчеты об ошибках; - разрабатывать техническое задание на сопровождение информационной системы ; - обслуживать локальную сеть; - моделировать интеллектуальные системы.
знать	<ul style="list-style-type: none"> - регламенты и нормы по обновлению и техническому сопровождению обслуживаемой информационной системы; - политику безопасности в современных информационных системах; достижения мировой и отечественной информатики в области интеллектуализации информационных систем; - принципы работы экспертных систем - структуру и этапы проектирования информационной системы; - основные методологии разработки и проектирования информационных систем - стратегии, цели и сценарии внедрения. - особенности сопровождения информационных систем управления качеством, технической и технологической подготовки производства. - структуру и этапы проектирования информационной системы. - сохранение и восстановление баз данных - определение показателей безотказности системы - особенности информационного, программного и технического обеспечения различных видов АИС.

	<ul style="list-style-type: none">- особенности сопровождения информационных систем управления качеством, технической и технологической подготовки производства.- структуру и этапы проектирования информационной системы.- основные модели интеллектуальных систем- архитектуру интеллектуальных информационных систем.
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. Формы промежуточной аттестации по «ПМ.06. СОПРОВОЖДЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ»

Наименование ПМ 03.	Курс семестр	Форма контроля и оценивания			
		Итоговая аттестация	Промежуточная аттестация	Рубежный контроль	Текущий контроль
«Сопровождение информационных систем»	МДК.06.01 Внедрение ИС				
	Курс 4 Сем.7		экзамен		тестирование, практическая работа
	МДК.06.02 Инженерно-техническая поддержка сопровождения ИС				
	Курс 4 Сем.8		экзамен		тестирование, практическая работа
	МДК.06.03 Устройство и функционирование информационной системы				
	Курс 4 Сем.7		экзамен		тестирование, практическая работа
	МДК.06.04 Интеллектуальные системы и технологии				
	Курс 4 Сем.8		экзамен		тестирование, практическая работа
	Учебная практика				
	Курс 4 Сем.8		Дифференцированный зачет		
	Производственная практика (по профилю специальности)				
	Курс 4 Сем.8		Дифференцированный зачет		
	Курс 4 Сем.8		Экзамен квалификационный		

3. Результаты освоения профессионального модуля, подлежащие проверке

Контроль и оценка результатов освоения ПМ осуществляется преподавателем в процессе текущего (рубежного) контроля, промежуточной аттестации в форме дифференцированного зачета, экзамена и экзамена квалификационного.

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Критерии оценки	Методы оценки
Раздел модуля 1. Ввод информационных систем в эксплуатацию		
<i>ПК 6.1</i> Разрабатывать техническое задание на сопровождение информационной системы	Оценка « отлично » - проанализирована предметная область функционирования системы; выделены и определены признаки системы по нескольким основаниям классификации; указаны все функции предложенной информационной системы; сформировано и обосновано несколько предложений по расширению перечня выполняемых функций. Сформированы и обоснованы предложения по реинжинирингу системы Оценка « хорошо » - проанализирована предметная область функционирования	Экзамен в форме собеседования: практическое задание по формированию предложений на расширение функциональности информационной системы Формирование предложений о реинжиниринге информационной си-

	<p>системы; выделены и определены признаки системы и указана ее принадлежность по классификации; указаны основные функции предложенной информационной системы; сформированы и обоснованы предложения по расширению перечня выполняемых функций.</p> <p>Сформированы предложения по реинжинирингу системы</p> <p>Оценка «удовлетворительно» - проанализирована предметная область функционирования системы; указана ее принадлежность по классификации; указаны функции предложенной информационной системы; сформированы предложения по расширению перечня выполняемых функций.</p> <p>Внесено хотя бы одно предложение по реинжинирингу системы</p>	<p>стемы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p><i>ПК 6.3</i> Разрабатывать обучающую документацию для пользователей информационной системы.</p>	<p>Оценка «отлично» - обучающая документация разработана с учетом особенностей пользователей; документация имеет понятную и логичную структуру, содержит достаточное количество рисунков, схем, таблиц; содержание позволяет освоить работу с информационной системой в достаточном объеме для указанной категории пользователей; оформление полностью соответствует требованиям стандартов.</p> <p>Оценка «хорошо» - обучающая документация разработана с учетом особенностей пользователей; документация содержит достаточное количество рисунков, схем, таблиц; содержание позволяет освоить работу с информационной системой в достаточном объеме для указанной категории пользователей; оформление соответствует требованиям стандартов.</p> <p>Оценка «удовлетворительно» - обучающая документация разработана; документация содержит рисунки, схемы, таблицы; содержание позволяет освоить работу с информационной системой без учета указанной категории пользователей; оформление в основном соответствует требованиям стандартов.</p>	<p>Экзамен в форме собеседования: практическое задание по разработке обучающей документации для указанной категории пользователей</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p>Раздел модуля 2. Обеспечение эксплуатации информационных систем</p>		
<p><i>ПК 6.2</i> Выполнять исправление ошибок в программном коде информационной системы.</p>	<p>Оценка «отлично» - проанализированы функции системы, проверено и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование си-</p>	<p>Экзамен в форме собеседования: практическое задание по обнаружению и исправлению ошибок программного кода информационной систе-</p>

	<p>стемы после исправления и сделан вывод о работоспособности.</p> <p>Оценка «хорошо» - проверено функционирование системы и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование системы после исправления и сделан вывод о работоспособности.</p> <p>Оценка «удовлетворительно» - проверено функционирование системы и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены некоторые причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование системы после исправления и сделан вывод о работоспособности.</p>	<p>мы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p><i>ПК 6.4</i> Оценивать качество и надежность функционирования информационной системы в соответствии с критериями технического задания.</p>	<p>Оценка «отлично» - проанализировано техническое задание и выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; протоколы оформлены в соответствии с требованиями стандартов и/или руководящих документов; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «хорошо» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «удовлетворительно» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы</p>	<p>Экзамен в форме собеседования: практическое задание по оценке качества функционирования информационной системы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p><i>ПК 6.5</i> Осуществлять техническое сопровождение, обновление и восста-</p>	<p>Оценка «отлично» - внесены заданные изменения в базу данных информационной системы; проверено сохранение изменений; выполнено обновление системных компонент; предложен и обоснован</p>	<p>Экзамен в форме собеседования: практическое задание по выполнению обновления и ре-</p>

<p>новление данных ИС в соответствии с техническим заданием.</p>	<p>план резервного копирования базы данных; резервное копирование выполнено. Оценка «хорошо» - внесены заданные изменения в базу данных информационной системы, изменения сохранены; выполнено обновление системных компонент; предложен план резервного копирования базы данных; резервное копирование выполнено.</p> <p>Оценка «удовлетворительно» - внесены заданные изменения в базу данных информационной системы, изменения сохранены; предложен план резервного копирования базы данных; резервное копирование выполнено.</p>	<p>зервного копирования базы данных информационной системы</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p>Раздел модуля 3. Виды, характеристики и особенности функционирования информационных систем</p>		
<p>ПК 6.2 Выполнять исправление ошибок в программном коде информационной системы.</p>	<p>Оценка «отлично» - проанализированы функции системы, проверено и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование системы после исправления и сделан вывод о работоспособности.</p> <p>Оценка «хорошо» - проверено функционирование системы и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование системы после исправления и сделан вывод о работоспособности.</p> <p>Оценка «удовлетворительно» - проверено функционирование системы и выявлено несоответствие выполняемых функций описанию (спецификации, техническому заданию и т.п.); выявлены и устранены некоторые причины несоответствия (внесены исправления в программный код); продемонстрировано функционирование системы после исправления и сделан вывод о работоспособности</p>	<p>Экзамен в форме собеседования: практическое задание по обнаружению и исправлению ошибок программного кода информационной системы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p>ПК 6.4 Оценивать качество и надежность функционирования информационной системы в соответствии с критериями тех-</p>	<p>Оценка «отлично» - проанализировано техническое задание и выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; протоколы оформле-</p>	<p>Экзамен в форме собеседования: практическое задание по оценке качества функционирования информационной системы.</p> <p>Защита отчетов по</p>

<p>нического задания.</p>	<p>ны в соответствии с требованиями стандартов и/или руководящих документов; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «хорошо» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «удовлетворительно» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы.</p>	<p>практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p>Раздел модуля 4. Особенности технического сопровождения интеллектуальных систем</p>		
<p><i>ПК 6.1</i> Разрабатывать техническое задание на сопровождение информационной системы.</p>	<p>Оценка «отлично» - проанализирована предметная область функционирования системы; выделены и определены признаки системы по нескольким основаниям классификации; указаны все функции предложенной информационной системы; сформировано и обосновано несколько предложений по расширению перечня выполняемых функций.</p> <p>Сформированы и обоснованы предложения по реинжинирингу системы</p> <p>Оценка «хорошо» - проанализирована предметная область функционирования системы; выделены и определены признаки системы и указана ее принадлежность по классификации; указаны основные функции предложенной информационной системы; сформированы и обоснованы предложения по расширению перечня выполняемых функций.</p> <p>Сформированы предложения по реинжинирингу системы</p> <p>Оценка «удовлетворительно» - проанализирована предметная область функционирования системы; указана ее принадлежность по классификации; указаны функции предложенной информационной системы; сформированы предложения по расширению перечня выполняемых функций.</p> <p>Внесено хотя бы одно предложение по реинжинирингу системы</p>	<p>Экзамен в форме собеседования: практическое задание по формированию предложений на расширение функциональности информационной системы</p> <p>Формирование предложений о реинжиниринге информационной системы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p><i>ПК 6.4</i> Оценивать качество и</p>	<p>Оценка «отлично» - проанализировано техническое задание и выполнена про-</p>	<p>Экзамен в форме собеседования:</p>

<p>надежность функционирования информационной системы в соответствии с критериями технического задания.</p>	<p>верка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; протоколы оформлены в соответствии с требованиями стандартов и/или руководящих документов; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «хорошо» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы; сделан вывод о соответствии системы действующим стандартам качества.</p> <p>Оценка «удовлетворительно» - выполнена проверка функционирования информационной системы в соответствии с разделом технического задания; качественные характеристики информационной системы, полученные в результате проверки внесены в протоколы.</p>	<p>практическое задание по оценке качества функционирования информационной системы.</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p><i>ПК 6.5</i> Осуществлять техническое сопровождение, обновление и восстановление данных ИС в соответствии с техническим заданием.</p>	<p>Оценка «отлично» - внесены заданные изменения в базу данных информационной системы; проверено сохранение изменений; выполнено обновление системных компонент; предложен и обоснован план резервного копирования базы данных; резервное копирование выполнено.</p> <p>Оценка «хорошо» - внесены заданные изменения в базу данных информационной системы, изменения сохранены; выполнено обновление системных компонент; предложен план резервного копирования базы данных; резервное копирование выполнено.</p> <p>Оценка «удовлетворительно» - внесены заданные изменения в базу данных информационной системы, изменения сохранены; предложен план резервного копирования базы данных; резервное копирование выполнено.</p>	<p>Экзамен в форме собеседования:</p> <p>практическое задание по выполнению обновления и резервного копирования базы данных информационной системы</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Экспертное наблюдение за выполнением различных видов работ во время учебной/ производственной</p>
<p>ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.</p>	<p>– обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач;</p> <p>- адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач</p>	<p>Экспертное наблюдение за выполнением работ</p>
<p>ОП 02. Осуществлять поиск, анализ и</p>	<p>- использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические</p>	

интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.	издания по специальности для решения профессиональных задач	
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.	- демонстрация ответственности за принятые решения - обоснованность самоанализа и коррекция результатов собственной работы;	
ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	- взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик; - обоснованность анализа работы членов команды (подчиненных)	
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.	Демонстрировать грамотность устной и письменной речи, - ясность формулирования и изложения мыслей	
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.	- соблюдение норм поведения во время учебных занятий и прохождения учебной и производственной практик,	
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.	- эффективное выполнение правил ТБ во время учебных занятий, при прохождении учебной и производственной практик; - демонстрация знаний и использование ресурсосберегающих технологий в профессиональной деятельности	
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятель-	- эффективность использовать средств физической культуры для сохранения и укрепления здоровья при выполнении профессиональной деятельности.	

ности и поддержания необходимого уровня физической подготовленности.		
ОК 09. Использовать информационные технологии в профессиональной деятельности.	- эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту;	
ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.	- эффективность использования в профессиональной деятельности необходимой технической документации, в том числе на английском языке.	

4. Типовые задания для оценки освоения дисциплины

МДК.06.01 Внедрение информационных систем

Практическое занятие:

по теме: «Разработка сценария внедрения информационной системы для рабочего места»

Цель:

По завершению практического занятия студент должен уметь:

Продолжительность: 2 аудиторных часа (90 минут)

Необходимые принадлежности

Практическая работа направлена на ознакомление с процессом описания ИС, получение навыков по использованию основных методов анализа ИС, ознакомление с процессом разработки требований к информационной системе и составления технического задания (ТЗ) на разработку программного обеспечения, получение навыков по использованию основных методов формирования и анализа требований.

Требования к результатам выполнения практикума:

1. наличие описания информационной системы;
2. проведение анализа осуществимости выполнения проекта;
3. наличие диаграммы идентификации точек зрения и диаграммы иерархии точек зрения;
4. наличие пользовательских требований, четко описывающих будущий функционал системы;
5. наличие системных требований, включающих требования к структуре, программному интерфейсу, технологиям разработки, общие требования к системе (надежность, масштабируемость, распределённость, модульность, безопасность, открытость, удобство пользования и т.д.);
6. наличие составленного ТЗ.

3. Теоретические сведения

3.1. Общие сведения о требованиях к информационным системам

Проблемы, которые приходится решать специалистам в процессе создания программного обеспечения, очень сложны. Природа этих проблем не всегда ясна, особенно если разрабатываемая программная система инновационная. В частности, трудно чётко описать те действия, которые должна выполнять система.

Описание функциональных возможностей и ограничений, накладываемых на систему, называется **требованиями** к этой системе, а сам процесс формирования, анализа, документирования и проверки этих функциональных возможностей и ограничений – **разработкой требований**.

Требования подразделяются на пользовательские и системные. **Пользовательские требования** – это описание на естественном языке (плюс поясняющие диаграммы) функций, выполняемых системой, и ограничений, накладываемых на неё.

Системные требования – это описание особенностей системы (архитектура системы, требования к параметрам оборудования и т.д.), необходимых для эффективной реализации требований пользователя.

3.2. Первые шаги по разработке требований к информационным системам - анализ осуществимости

Разработка требований — это процесс, включающий мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Для новых программных систем процесс разработки требований должен начинаться с анализа осуществимости.

Началом такого анализа является общее описание системы и ее назначения, а результатом анализа — отчет, в котором должна быть четкая рекомендация, продолжать или нет процесс разработки требований проектируемой системы. Другими словами, анализ осуществимости должен осветить следующие вопросы.

1. Отвечает ли система общим и бизнес-целям организации-заказчика и организации-разработчика?
2. Можно ли реализовать систему, используя существующие на данный момент технологии и не выходя за пределы заданной стоимости?
3. Можно ли объединить систему с другими системами, которые уже эксплуатируются?

Выполнение анализа осуществимости включает сбор и анализ информации о будущей системе и написание соответствующего отчета. Сначала следует определить, какая именно информация необходима, чтобы ответить на поставленные выше вопросы.

Например, эту информацию можно получить, ответив на следующее:

1. Что произойдет с организацией, если система не будет введена в эксплуатацию?
2. Какие текущие проблемы существуют в организации и как новая система поможет их решить?
3. Каким образом система будет способствовать целям бизнеса?
4. Требуется ли разработка системы технологии, которая до этого не использовалась в организации?

Далее необходимо определить источники информации. Это могут быть менеджеры отделов, где система будет использоваться, разработчики программного обеспечения, знакомые с типом будущей системы, технологи, конечные пользователи и т.д.

После обработки собранной информации готовится отчет по анализу осуществимости создания системы. В нем должны быть даны рекомендации относительно продолжения разработки системы. Могут быть предложены изменения бюджета и графика работ по созданию системы или предъявлены более высокие требования к системе.

Различают четыре основных этапа процесса разработки требований:

1. анализ технической осуществимости создания системы,
2. формирование и анализ требований,
3. специфицирование требований и создание соответствующей документации,
4. аттестация этих требований.

На рис. 1 показаны взаимосвязи между этими этапами и результаты, сопровождающие каждый этап процесса разработки системных требований.

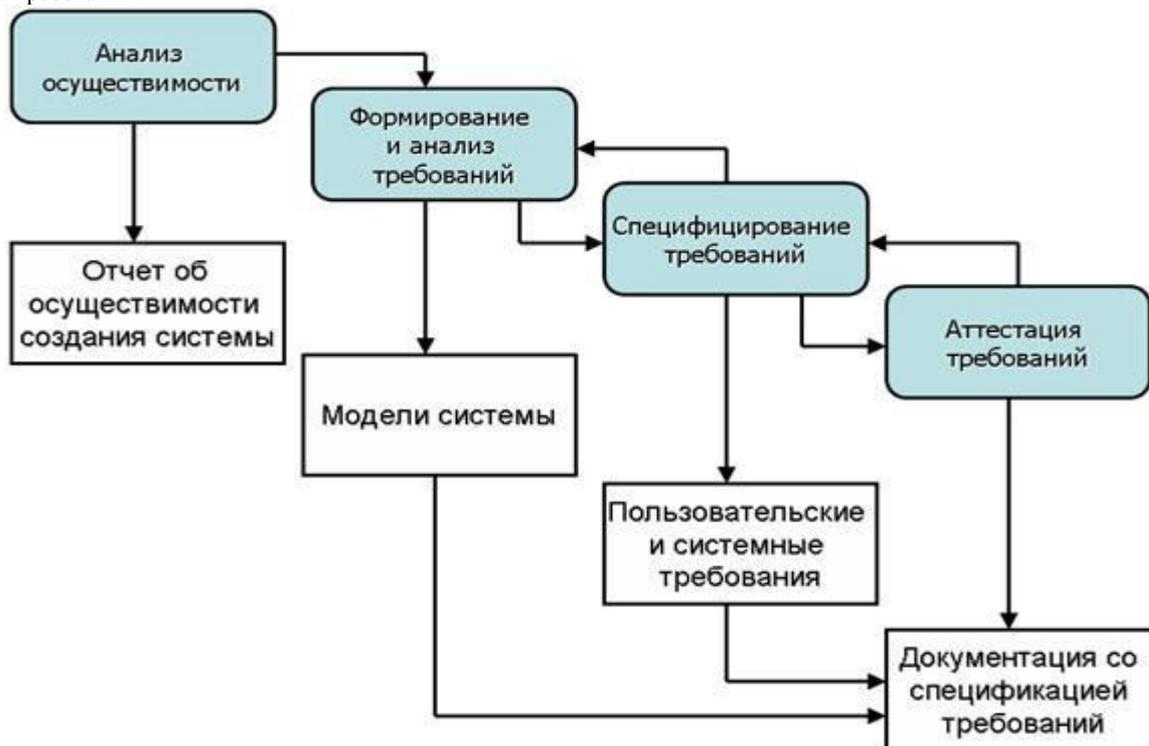


Рис. 1. Процесс разработки требований

Но поскольку в процессе разработки системы в силу разнообразных причин требования могут меняться, управление требованиями, т.е. процесс управления изменениями системных требований, является необходимой составной частью деятельности по их разработке.

3.4. Формирование и анализ требований

Следующим этапом процесса разработки требований является формирование (определение) и анализ требований. Обобщенная модель процесса формирования и анализа требований показана на рис. 2. Каждая организация использует собственный вариант этой модели, зависящий от “местных факторов”: опыта работы коллектива разработчиков, типа разрабатываемой системы, используемых стандартов и т.д.

Процесс формирования и анализа требований проходит через ряд этапов.

1. *Анализ предметной области.* Аналитики должны изучить предметную область, где будет эксплуатироваться система.
2. *Сбор требований.* Это процесс взаимодействия с лицами, формирующими требования. Во время этого процесса продолжается анализ предметной области.
3. *Классификация требований.* На этом этапе бесформенный набор требований преобразуется в логически связанные группы требований.
4. *Разрешение противоречий.* Без сомнения, требования многочисленных лиц, занятых в процессе формирования требований, будут противоречивыми. На этом этапе определяются и разрешаются противоречия различного рода.
5. *Назначение приоритетов.* В любом наборе требований одни из них будут более важны, чем другие. На этом этапе совместно с лицами, формирующими требования, определяются наиболее важные требования.
6. *Проверка требований.* На этом этапе определяется их полнота, последовательность и непротиворечивость.

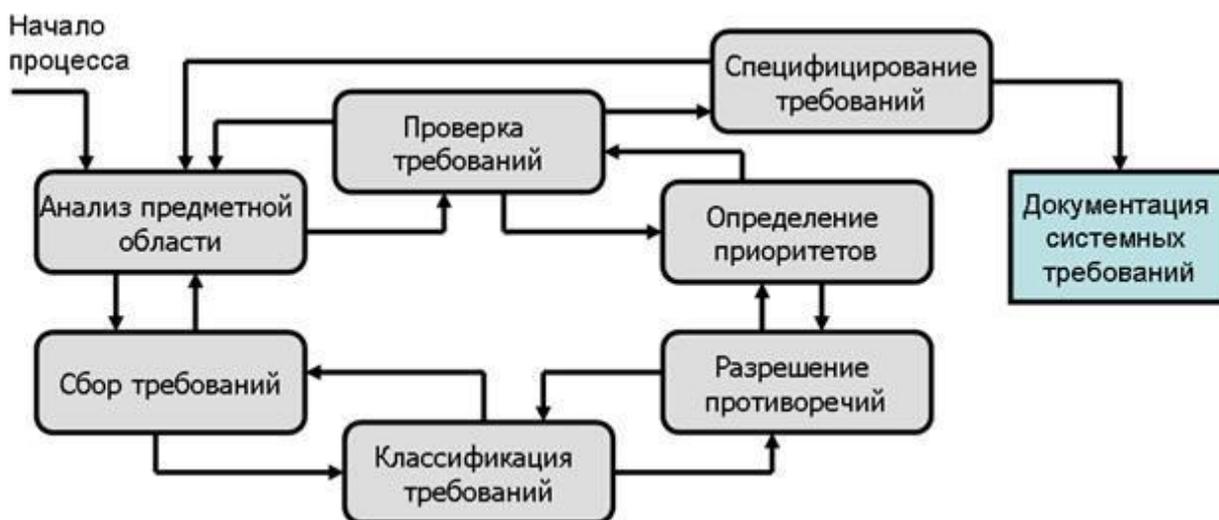


Рис. 2. Процесс формирования и анализа требований

Процесс формирования и анализа требований циклический, с обратной связью от одного этапа к другому. Цикл начинается с анализа предметной области и заканчивается проверкой требований. Понимание требований предметной области увеличивается в каждом цикле процесса формирования требований.

Рассмотрим три основных подхода к формированию требований: метод, основанный на множестве опорных точек зрения, сценарии и этнографический метод.

3.5. Опорные точки зрения

Подход с использованием различных *опорных точек зрения* к разработке требований признает различные (опорные) точки зрения на проблему и использует их в качестве основы построения и организации как процесса формирования требований, так и непосредственно самих требований.

Различные методы предлагают разные трактовки выражения "точка зрения". Точки зрения можно трактовать следующим образом.

1. *Как источник информации о системных данных.* В этом случае на основе опорных точек зрения строится модель создания и использования данных в системе. В процессе формирования требований отбираются все такие точки зрения (и на их основе определяются данные), которые будут созданы или использованы при работе системы, а также способы обработки этих данных.
2. *Как структура представлений.* В этом случае точки зрения рассматриваются как особая часть модели системы. Например, на основе различных точек зрения могут разрабатываться модели "сущность-связь", модели конечного автомата и т.д.
3. *Как получатели системных сервисов.* В этом случае точки зрения являются внешними (относительно системы) получателями системных сервисов. Точки зрения помогают определить данные, необходимые для выполнения системных сервисов или их управления.

Наиболее эффективным подходом к анализу таких систем является использование внешних опорных точек зрения. На основе этого подхода разработан метод **VORD** (Viewpoint-Oriented Requirements Definition — определение требований на основе точек зрения) для формирования и анализа требований.

Основные этапы метода VORD показаны на рис. 3.:

1. Идентификация точек зрения, получающих системные сервисы, и идентификация сервисов, соответствующих каждой точке зрения.
2. Структурирование точек зрения — создание иерархии сгруппированных точек зрения. Общесистемные сервисы предоставляются более высоким уровням иерархии и наследуются точками зрения низшего уровня.
3. Документирование опорных точек зрения, которое заключается в точном описании идентифицированных точек зрения и сервисов.
4. Отображение системы точек зрения, которая показывает системные объекты, определенные на основе информации, заключенной в опорных точках зрения.



Рис. 3. Метод VORD

Пример. Рассмотрим использование метода VORD на первых трех шагах анализа требований для системы поддержки заказа и учета товаров в магазине. В магазине для каждого товара фиксируется место хранения (определенная полка), количество товара и его поставщик. Система поддержки заказа и учета товаров должна обеспечивать добавление информации о новом товаре, изменение или удаление информации об имеющемся товаре, хранение (добавление, изменение и удаление) информации о поставщиках, включающей в себя название фирмы, ее адрес и телефон. При помощи системы составляются заказы поставщикам. Каждый заказ может содержать несколько позиций, в каждой позиции указываются наименование товара и его количество в заказе. Система по требованию пользователя формирует и выдает на печать следующую справочную информацию:

- список всех товаров;
- список товаров, имеющихся в наличии;
- список товаров, количество которых необходимо пополнить;
- список товаров, поставляемых данным поставщиком.

Первым шагом в формировании требований является идентификация опорных точек зрения. Во всех методах формирования требований, основанных на использовании точек зрения, начальная идентификация является наиболее трудной задачей. Один из подходов к идентификации точек зрения — метод "мозговой атаки", когда определяются потенциальные системные сервисы и организации, взаимодействующие с системой. Организуется встреча лиц, участвующих в формировании требований, которые предлагают свои точки зрения. Эти точки зрения представляются в виде диаграммы, состоящей из ряда круговых областей, отображающих возможные точки зрения (рис. 4). Во время "мозговой атаки" необходимо идентифицировать потенциальные опорные точки зрения, системные сервисы, входные данные, нефункциональные требования, управляющие события и исключительные ситуации.

Следующей стадией процесса формирования требований будет идентификация опорных точек зрения (на рис. 4 показаны в виде темных круговых областей) и сервисов (показаны в виде затененных областей). Сервисы должны соответствовать опорным точкам зрения. Но могут быть сервисы, которые не поставлены им в соответствие. Это означает, что на начальном этапе "мозговой атаки" некоторые опорные точки зрения не были идентифицированы.

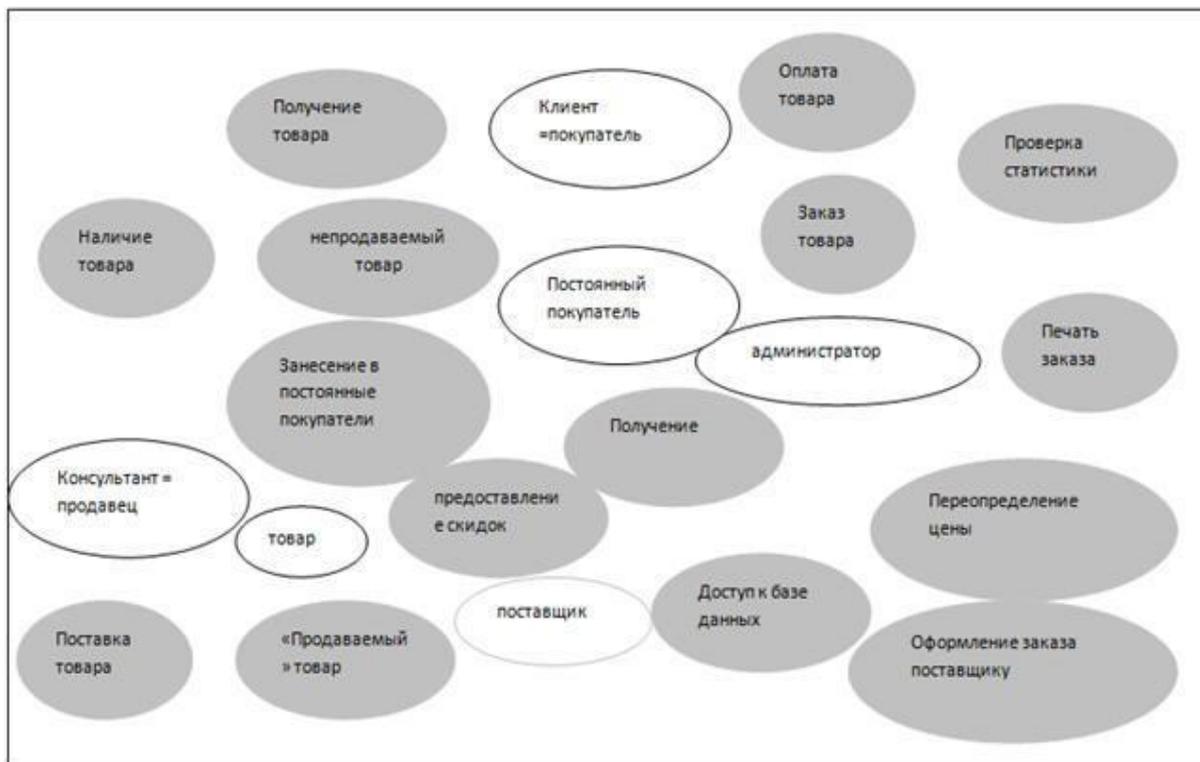


Рис. 4. Диаграмма идентификации точек зрения

В таблице 1 показано распределение сервисов для некоторых идентифицированных на рис. 4 точек зрения. Один и тот же сервис может быть соотнесен с несколькими точками зрения.

Таблица 1 - Сервисы, соотнесенные с точками зрения

клиент	покупатель	постоянный покупатель	товар	поставщик	продавец	администратор
Проверка наличия товара	Занесение в список постоянных клиентов	Получение скидки	Прием товара	Занесение в базу данных (название, адрес, телефон и т.д.)	Продажа товара	Доступ к базе данных
Покупка товара		Получение информации о новых поступлениях	Занесение в базу данных (данные о поставщике, колве, месте хранения и.д.)		Печать чека	Проверка статистики
Получение чека			Назначение цены		Доступ к каталогу	Переопределение цены
Заказ товара			Переопределение цены		Проверка наличия товара	Оформление заказа поставщику
Занесение покупателя и суммы покупки в базу данных			«Покупаемый» или «непокупаемый» товар		Оформление заказа покупателю	Печать заказа

Информация, извлеченная из точек зрения, используется для заполнения форм шаблонов точек зрения и организации точек зрения в иерархию наследования. Это позволяет увидеть общие точки зрения и повторно использовать информацию в иерархии наследования. Сервисы, данные и управляющая информация наследуются подмножеством точек зрения. На рис. 5 показана часть иерархии точек зрения для системы поддержки заказа и учета товаров.



Рис. 5. Иерархия точек зрения

Практическое занятие:

по теме: «Сравнительный анализ офисных пакетов»

Цель: научиться анализировать программы с офисных пакетов

По завершению практического занятия студент должен уметь: анализировать с офисных пакетов

Продолжительность: 2 аудиторных часа (90 минут)

Необходимые принадлежности

Компьютеры.

Задание

Сравнение свободных офисных пакетов с MS Office

Наименование пакета	Последняя версия	Лицензия	Стоимость	Операционная система	Поддержка Microsoft Office (doc, xls...)	Поддержка Microsoft Office Open XML (docx, xlsx, pptx)	Поддержка OpenDocument (odt, odp, ods, odg)	Поддержка Portable Document Format (pdf)
Microsoft Office	Microsoft Office Профессиональный плюс 2016 (июль 2015)	Microsoft EULA	Профессиональный плюс от 25 000 руб.	Windows, OS X, Windows Phone, Android, iOS	Да	Да	Да	Да
LibreOffice	5.2.2 (сентябрь 2016)	Mozilla Public License 2.0	Полностью Бесплатно	Windows, OS X, Linux, BSD, Unix, Solaris, illumos	Да	Да	Да	Да
Apache OpenOffice	4.1.3 (апрель 2016)	Apache License 2.0	Полностью Бесплатно	Windows, OS X, Linux, BSD, Unix, Solaris, illumos	Да	Да	Да	Только экспорт
Calligra Suite (Calligra Gemini)	2.9.6.0 (2015)	GNU LGPL	Полностью Бесплатно	Windows, OS X, Linux, BSD, Solaris, illumos	Только чтение	Да	Да	Только экспорт
ONLYOFFICE Desktop Editors	4.1.2.270 (2016)	GNU AGPL	Полностью Бесплатно	Windows, Linux, OS X, iOS	Да	Да	Да	Да
Функциональные возможности								
	Microsoft Office	LibreOffice	Apache OpenOffice	Calligra Suite	ONLYOFFICE Desktop Editors			
Текстовый процессор	Microsoft Word	LibreOffice Writer	OpenOffice Writer	Calligra Words	ONLYOFFICE Редактор документов			
Табличный процессор	Microsoft Excel	LibreOffice Calc	OpenOffice Calc		ONLYOFFICE Редактор таблиц			
Редактор презентаций	Microsoft PowerPoint	LibreOffice Impress	OpenOffice Impress	Calligra Stage	ONLYOFFICE Редактор презентаций			
Создание диаграмм	+	+	+	+	+			
Графический редактор	-	LibreOffice Draw	OpenOffice Draw					
Редактор формул	+	LibreOffice Math	OpenOffice Math		+			

Много лет назад был всего один офисный пакет с открытым исходным кодом для операционной системы Windows, который можно было сравнивать с Microsoft Office или с пакетами под лицензией Freeware. Сегодня таких проектов уже четыре.

Для того, чтобы выбрать один из них, стоит сравнить их между собой, а также с нынешним эталоном - Microsoft Office.

В сравнении будут участвовать следующие программные продукты:

- Microsoft Office Профессиональный плюс 2016;
- LibreOffice 5.2.2 x86 ([подробнее](#));
- Apache OpenOffice 4.1.3 x86 ([подробнее](#));
- Calligra Gemini 2.9.6.0 x64 ([подробнее](#));
- ONLYOFFICE Desktop Editors 4.1.2.270 x64 ([подробнее](#)).

Пункт сравнения	Microsoft Office Pro Plus 2016	LibreOffice	Apache OpenOffice	Calligra Gemini	ONLYOFFICE Desktop Editors
Последняя версия	Microsoft Office Профессиональный плюс 2016 (июль 2015)	5.2.2 (сентябрь 2016)	4.1.3 (октябрь 2016)	2.9.6.0 (2015)	4.1.2.270 (2016)
Лицензия	Microsoft EULA	Mozilla Public License 2.0	Apache License 2.0	GNU LGPL	GNU AGPL
Стоимость	Профессиональный плюс ~ 30 000 р Для дома и учебы ~ 5 000 р	Полностью Бесплатно	Полностью Бесплатно	Полностью Бесплатно	Полностью Бесплатно
Операционная система	Windows, OS X, Windows Phone, Android, iOS	Windows, OS X, Linux, BSD, Unix, Solaris/Illumos	Windows, OS X, Linux, BSD, Unix, Solaris/Illumos	Windows, OS X, Linux, BSD, Solaris/Illumos	Windows, Linux, OS X, iOS
Поддержка Microsoft Office (doc, xls, ppt)	Да	Да	Да	Только чтение	Да
Поддержка Microsoft Office Open XML (docx, xlsx, pptx)	Да	Да	Да	Да	Да
Поддержка OpenDocument (odt, odf, ods, odg)	Да	Да	Да	Да	Да
Поддержка Portable Document Format (pdf)	Да	Да	Только экспорт	Только экспорт	Да
Текстовый процессор	Microsoft Word	LibreOfficeWriter	OpenOffice Writer	Calligra Words	ONLYOFFICE Редактор документов
Табличный процессор	Microsoft Excel	LibreOffice Calc	OpenOffice Calc	-	ONLYOFFICE Редактор таблиц
Редактор презентаций	Microsoft PowerPoint	LibreOffice Impress	OpenOffice Impress	Calligra Stage	ONLYOFFICE Редактор презентаций
Создание диаграмм	+	+	+	+	+
Графический редактор	-	LibreOffice Draw	OpenOffice Draw	-	-
Редактор формул	+	LibreOffice Math	OpenOffice Math	+	+
Система Управления Базами Данных	Microsoft Access (Отсутствует в версии Для дома и учебы)	LibreOffice Base	OpenOffice Base	-	-

Редактор HTML	+	LibreOffice Writer	OpenOffice Writer	-	+
Совместная работа	+	-	-	-	+
Поддержка вкладок	-	-	-	-	+
Проверка орфографии (русский язык)	+	+	+	-	+
Шаблоны	+	+	+	+	-
Открытие и редактирование PDF файлов	Редактирование	Редактирование через LibreOffice Draw	Только экспорт	Только экспорт	Открытие и создание (без редактирования)
Расширения	+	+	+	-	+
Локализация (русский язык)	+	+	+	-	+
Оценка	17 зеленых, 3 красных и 1 желтый	19 зеленых, 2 красных	17 зеленых, 2 желтых и 2 красных	9 зеленых, 9 желтых и 3 красных	18 красных, 2 красных и 1 желтый

Практическое занятие:

по теме: «Сравнительный анализ методологий проектирования»

Цель: научиться анализировать методологий проектирования

По завершению практического занятия студент должен уметь: анализировать методологий проектирования

Продолжительность: 5 аудиторных часа (225 минут)

Необходимые принадлежности

Компьютеры.

Задание

Наибольшее распространение при этом получают готовые решения, предлагаемые компаниями 1С или SAP (Systems, Applications and Products in Data Processing), а также системы моделирования процессов в различных нотациях, на которых базируются программные средства проектирования. Подавляющее большинство современных программ позволяют осуществлять автоматическую генерацию программного кода спроектированной информационной системы, что позволяет получать готовый продукт, реализованный, как правило, в форме веб-приложения. При этом открытым остается вопрос качества генерируемого кода, его понятности и читабельности.

Наиболее широко распространены следующие нотации: стандарты IDEF (Integration Definition for Function Modeling), в частности IDEF0 и IDEF3, диаграммы потоков данных DFD (Data Flow Diagram), унифицированный язык моделирования UML (Unified Modeling Language) и нотация BPMN (Business Process Modeling Notation), применяемая для моделирования бизнес-процессов.

В настоящее время стандарт IDEF0 считается устаревающим и наиболее часто используется только лишь при описании системы в рамках предпроектного исследования.

Постановка задачи

Основной целью работы являлось исследование допустимости применения стандартов разного уровня абстракции для решения аналогичных задач.

В рамках проводимого анализа была поставлена задача проектирования АСОИУ (автоматизированной системы обработки информации и управления), основного процесса документооборота для регистратуры больницы, осуществляемого с помощью программных средств моделирования различных нотаций.

Для этого были произведены сопоставление и анализ стандартов UML и BPMN. Помимо теоретического анализа было проведено практическое сравнение нотаций на примере двух систем проектирования для заданного объекта автоматизации. Таким образом, ключевым методом исследования является сравнение различных этапов проектирования информационных систем в разных нотациях на примере двух программ, реализующих работу с этими стандартами.

Выбор данной предметной области обусловлен тем, что она имеет достаточно строгую и упорядоченную структуру, но при этом ее внутренние процессы не являются стандартными. При этом вышеуказанные стандарты для проектирования информационных систем в выбранной сфере, как правило, не используются.

В качестве основных критериев сравнения выбраны параметры, которые наиболее точно определяют качество и адекватность проектируемой модели, а также степень ее соответствия реальной системе, доступности, понятности и практической полезности. К таким критериям относятся, прежде всего, выразительная мощь стандарта визуального проектирования, структурированность, возможность создания готовой информационной системы, ее практического применения, удобство, скорость разработки и внедрения и т.д.

Рассмотрение стандартов

UML – согласно одному из определений – язык графического описания для объектного моделирования процессов, в частности производственных, а также бизнес-процессов, системного проектирования, процессов разработки различных систем, программного обеспечения, а также описания и отображения организационных структур [1].

UML является открытым стандартом, использующим визуальные графические обозначения для проектирования абстрактной модели системы, рассматривая ее с точки зрения конструктивного описания. При этом система рассматривается как набор взаимосвязанных сущностей – объектов [1].

Технология визуального моделирования предоставляет возможность упрощенной и наглядной работы со сложными системами, позволяя более детально рассматривать как систему в целом, так и ее отдельные компоненты.

Основным преимуществом унифицированного языка является то, что он, прежде всего, является объектно-ориентированным, в результате чего методы описания результатов анализа и проектирования системы структурно близки к методам непосредственного программирования на современных объектно-ориентированных языках [2].

Вторым плюсом применения визуальных моделей при проектировании АСОИУ является то, что они позволяют организовать эффективное взаимодействие между участниками процесса анализа и автоматизации системы: заказчиками, аналитиками и разработчиками.

BPМN (Business Process Modeling Notation) – спецификация, содержащая графическую нотацию описания бизнес-процессов на диаграммах, называемых BPD (Business Process Diagram). Данная нотация, подобно прочим, призвана обеспечить взаимопонимание между всеми участниками процесса анализа и автоматизации системы [5].

Нотации UML и BPМN не являются взаимоисключающими. Несмотря на идентичность некоторых функций, схемы процессов в этих нотациях отличаются по визуальному представлению информации.

Основным отличием данных стандартов является то, что UML рассматривает систему в виде взаимосвязанных объектов – классов, образующих ее, и их взаимодействия, в то время как в BPМN система описывается на более высоком абстрактном уровне – уровне бизнес-процессов. Главным в данной нотации являются процессы, а не объекты [5].

Следует отметить, что сравнение данных методологий уже неоднократно производилось. По результатам проведенных исследований отмечалось, что в плане выразительной мощи оба стандарта приблизительно одинаково эффективны. Однако, как уже было упомянуто, BPМN является более высокоуровневой нотацией, рассматривающей систему с точки зрения основного процесса. Это включает в себе определенный минус, поскольку описание организационной структуры и модели данных затрудняется.

Однако на сегодняшний день существуют BPMS (Business Process Modeling System), позволяющие описывать не только процессы организации, но также структуру и модель данных.

Выбор средств разработки

На сегодняшний день существует большое разнообразие программного обеспечения для работы с рассматриваемыми стандартами.

В ходе работы был проведен сравнительный анализ существующего программного обеспечения. По результату анализа выбор был сделан в пользу open-source системы ArgoUML. Данная система является кроссплатформенной, иными словами может работать практически на всех платформах [3].

Среди BPMS выбор был сделан в пользу системы Bizagi. Данная BPМ-система направлена на моделирование, исполнение, автоматизацию и анализ бизнес-процессов.

Основной ее особенностью среди многих прочих BPMS является возможность описания модели данных для будущей информационной системы.

Разработка системы

При проектировании АСОИУ выбранного к рассмотрению объекта первоначально была разработана действующая модель с помощью UML. По данной модели была осуществлена генерация программного кода и получено работоспособное приложение автоматизации документооборота медицинского учреждения.

Следующим шагом было осуществлено проектирование аналогичной системы средствами BPМN системы Bizagi.

Обе реализованные системы являются работоспособными. Следует отметить, что поскольку моделирование объекта автоматизации производилось упрощенно, разрабатываемые модели не включали в себя отражение специфических функций, присутствующих в системе. За счет этого можно было явно проследить параллели различных диаграмм нотаций UML и BPМN.

При проектировании информационной системы в среде ArgoUML был создан проект, содержащий две стандартные для UML диаграммы: классов и автоматов (состояний).

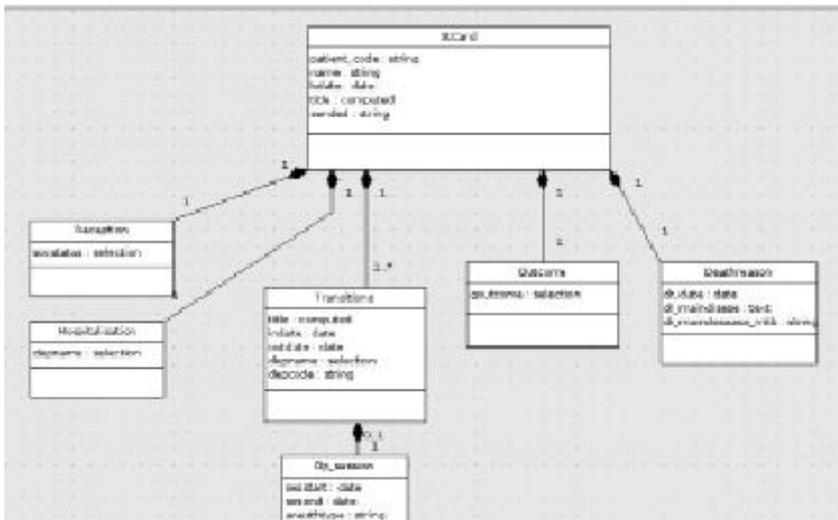


Рис. 1. Диаграмма классов

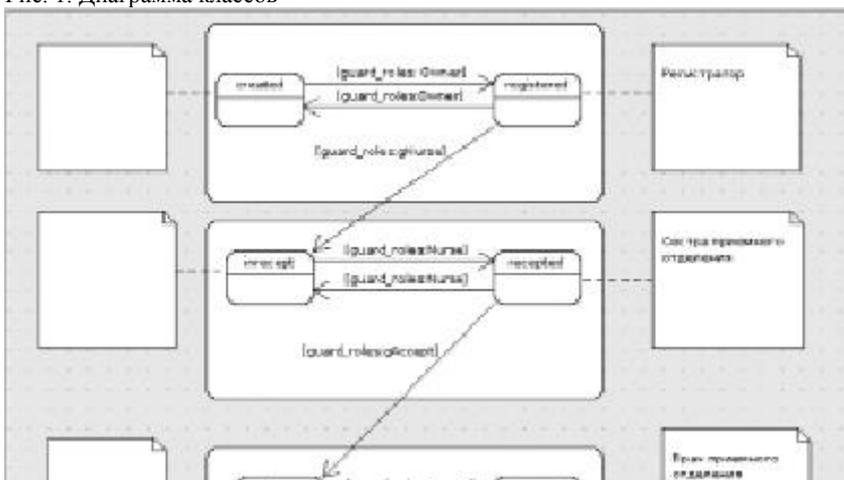


Рис. 2. Диаграмма состояний

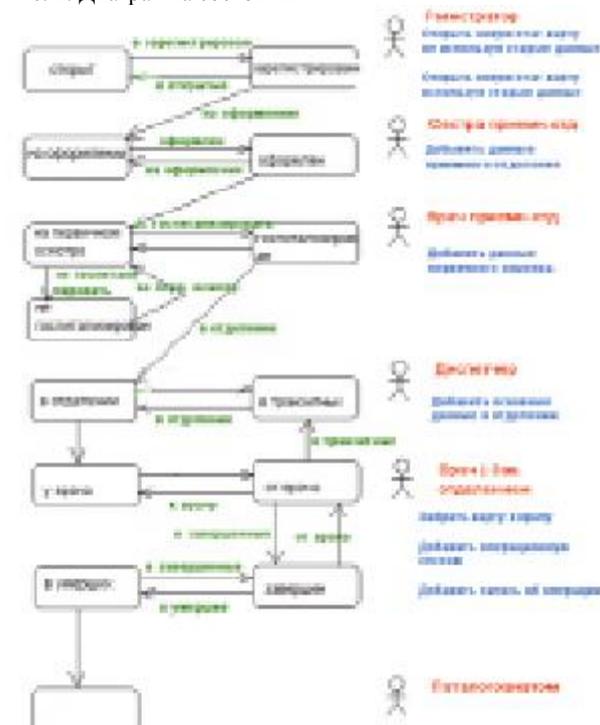


Рис. 3. Схема документооборота

Диаграмма классов UML позволяет обозначать отношения между классами и их экземплярами, абстрагируясь от предметной области и работая исключительно с сущностями, реализуя принцип объектно-ориентированного программирования. Полученная схема является моделью данных будущего приложения (рис. 1).

Эта диаграмма является основным уровнем описания структуры организации и работы системы [2].

Диаграмма состояний – это один из способов детального описания системы в определенные различные состояния и переходов между ними (рис. 2).

В рассматриваемом примере для реализации системы оборота медицинской карты данных достаточно использования этих двух диаграмм [6, 7].

Субъектами – участниками процесса документооборота являются: регистратор, сестра приемного отделения, врач приемного отделения, диспетчер отделения, лечащий врач, патологоанатом (рис. 3) [6].

По окончании проектирования двух диаграмм проект был экспортирован в формат XMI (XML Metadata Interchange – стандарт обмена метаданными с помощью языка расширенной разметки XML), и с помощью системы Plone было сгенерировано готовое приложение [4]. Полученное веб-приложение представляет собой интерфейс для взаимодействия с базой данных.

Далее было спроектировано аналогичное приложение средствами Bizagi Studio. Первым шагом была создана диаграмма процесса в нотации BPM (рис. 4).

Для выбранной предметной области можно четко проследить связь этой диаграммы и диаграммы состояний нотации UML. Субъекты, взаимодействующие в данной системе (роли), отражены на «дорожках» процесса.

Описанные функции субъектов в диаграмме состояний UML находят отражение в блоках задач и условных переходах. Процесс можно поделить на несколько крупных блоков: Регистрация, Осмотр, Лечение и Выписка.

Наиболее значимым при проектировании информационной системы, помимо моделирования процесса, является второй шаг разработки в среде Bizagi Studio – создание модели данных будущего приложения (рис. 5).

Данный шаг в рассматриваемой системе эквивалентен схеме взаимодействия объектов, описываемой на диаграмме классов UML.

Для каждой задачи впоследствии создаются формы интерфейса на третьем шаге проектирования. Логика задач и условных переходов описывается на четвертом шаге. После выполнения оставшихся шагов система автоматически генерирует программный код и создает приложение, доступное в браузере.

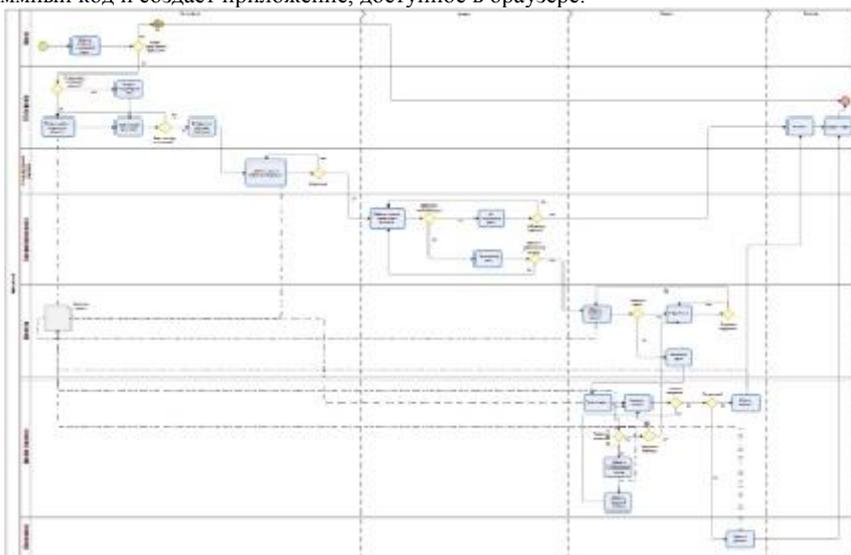


Рис. 4. Диаграмма процесса BPMN

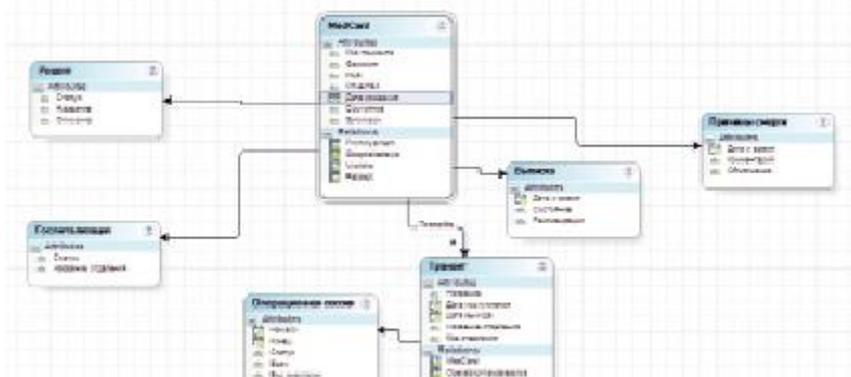


Рис. 5. Модель данных

Заключение

Реализованные приложения для оптимизации работы системы документооборота больницы в настоящее время обладают достаточно низким функционалом.

В рамках исследования отработана связка ArgoUML, ArchGen, Plone, с помощью которой была автоматически получена информационная система без написания программного кода [6]. Программа Bizagi позволяет создать приложение подобного функционала встроенными средствами.

На основании проделанной работы можно сделать вывод о том, что для простейших автоматизированных систем работы с данными, в частности документами, выразительная мощь обеих нотаций моделирования схожа. Более того, при реализации описаний простейших систем четко прослеживаются параллели между различными диаграммами в разных стандартах.

Однако при проектировании систем, в которых организационная структура играет большую роль, чем типовые процессы, предпочтение следует отдавать UML.

Скорость разработки системы с помощью рассматриваемых технологий, а также степень практической применимости выходящих продуктов в целом одинаковы и зависят от человеческого фактора.

Помимо непосредственно практического применения, данные системы рекомендуется в настоящем их состоянии использовать в учебном процессе для изучения технологии проектирования и моделирования информационных систем в качестве практического учебного материала.

Лабораторное занятие:

по теме: «Разработка графика разработки и внедрения информационной системы»

Цель: научиться анализировать методологий проектирования

По завершению практического занятия студент должен уметь: анализировать методологий проектирования

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Проектирование информационных систем: метод. указания к лабораторным работам / сост. С.В. Капустина, А.В. Паршаков; ФГОУ ВПО "СФУ". – Красноярск, 2008. – 80 с.

Задание

Принципы создания информационной системы

Многие пользователи компьютерной техники и программного обеспечения неоднократно сталкивались с ситуацией, когда программное обеспечение, хорошо работающее на одном компьютере, не работает на другом таком же устройстве. Или системные блоки одного вычислительного устройства не стыкуются с аппаратной частью другого. Или информационная система другой компании упорно не желает обрабатывать данные, которые вы подготовили в информационной системе у себя на рабочем месте. Эта проблема называется проблемой совместимости вычислительных, телекоммуникационных и информационных устройств.

Развитие систем и средств вычислительной техники, расширенное их внедрение во все сферы науки, техники, сферы обслуживания и быта привели к необходимости объединения конкретных вычислительных устройств и реализованных на их основе информационных систем в единые информационно-вычислительные системы (ИВС) и среды. При этом разработчики ИВС столкнулись с рядом проблем.

Например, разнородность технических средств вычислительной техники с точки зрения организации вычислительного процесса, архитектуры, системы команд, разрядности процессора и шины данных и т. д. потребовала создания физических интерфейсов, реализующих, как правило, взаимную совместимость устройств. При увеличении числа типов интегрируемых устройств сложность организации физического интерфейса между ними существенно возрастала. Разнородность программируемых сред, реализуемых в конкретных вычислительных устройствах и системах, с точки зрения многообразия операционных систем, различия в разрядности и прочих особенностей привела к созданию программных интерфейсов между устройствами и системами. При этом необходимо отметить, что достигнуть полной совместимости программных продуктов, разработанных для конкретной программной среды, в другой среде удавалось не всегда. Разнородность интерфейсов общения в системе "человек-компьютер" требовала постоянного согласования программно-аппаратного обеспечения и переподготовки кадров.

Принцип "открытости" информационной системы

Решение проблем совместимости привело к разработке большого числа международных стандартов и соглашений в сфере применения информационных технологий и разработки информационных систем. основополагающим понятием стало понятие открытые системы.

Термин "открытая система" сегодня можно определить как "исчерпывающий и согласованный набор международных стандартов на информационные технологии и профили функциональных стандартов, которые специфицируют интерфейсы, службы и поддерживающие их форматы, чтобы обеспечить взаимодействие и мобильность программных приложений, данных и персонала".

Это определение, сформулированное специалистами института IEEE (Institute of Electrical and Electronic Engineers), унифицирует содержание среды, которую предоставляет открытая система для широкого использования. В настоящее время общепризнанным координационным центром по разработке и согласованию стандартов открытых систем является OASIS (Organization for the Advancement of Structured Information Standards).

Общие свойства открытых информационных систем можно сформулировать следующим образом:

- расширяемость/масштабируемость: обеспечение возможности добавления новых функций ИС или изменения некоторых уже имеющихся при неизменных остальных функциональных частях ИС;
- мобильность/переносимость: обеспечение возможности переноса программ, данных при модернизации или замене аппаратных платформ ИС и возможности работы с ними специалистов, пользующихся ИТ, без их переподготовки при изменениях ИС;
- взаимодействие: способность к взаимодействию с другими ИС (технические средства, на которых реализована информационная система, объединяются сетью или сетями различного уровня: от локальной до глобальной);
- стандартизуемость: ИС проектируются и разрабатываются на основе согласованных международных стандартов и предложений, реализация открытости осуществляется на базе функциональных стандартов (профилей) в области информационных технологий;
- дружелюбность к пользователю: развитые унифицированные интерфейсы в процессах взаимодействия в системе "человек-машина", позволяющие работать пользователю, не имеющему специальной "компьютерной" подготовки.

Новый взгляд на открытые системы определяется тем, что эти черты рассматриваются в совокупности, как взаимосвязанные, и реализуются в комплексе, что вполне естественно, поскольку все указанные выше свойства дополняют друг друга. Только в совокупности возможности открытых систем позволяют решать проблемы проектирования, разработки и внедрения современных информационных систем.

Структура среды информационной системы

Обобщенная структура любой ИС может быть представлена двумя взаимодействующими частями:

- функциональной части, включающей прикладные программы, которые реализуют функции прикладной области;
 - среды или системной части, обеспечивающей исполнение прикладных программ.
- С этим разделением тесно связаны две группы вопросов стандартизации:
- стандарты интерфейсов взаимодействия прикладных программ со средой ИС, прикладной программный интерфейс (Application Program Interface — API);
 - стандарты интерфейсов взаимодействия самой ИС с внешней для нее средой (External Environment Interface — EEI).

Эти две группы интерфейсов определяют спецификации внешнего описания среды ИС — архитектуру, с точки зрения конечного пользователя, проектировщика ИС, прикладного программиста, разрабатывающего функциональные части ИС.

Спецификации внешних интерфейсов среды ИС и, как будет видно далее, спецификации интерфейсов взаимодействия между компонентами самой среды, — это точные описания всех необходимых функций, служб и форматов определенного интерфейса. Совокупность таких описаний составляет эталонную модель открытых систем (Reference Open System Model).

Эта модель используется более 20 лет и определяется системной сетевой архитектурой (SNA), предложенной IBM в 1974 году. Она основана на разбиении вычислительной среды на семь уровней, взаимодействие между которыми описывается соответствующими стандартами, и обеспечивает связь уровней вне зависимости от построения уровня в каждой конкретной реализации (рис. 4.1). Основным достоинством этой модели является детальное описание связей в среде с точки зрения технических устройств и коммуникационных взаимодействий. Вместе с тем она не принимает в расчет взаимосвязь с учетом мобильности прикладного программного обеспечения.

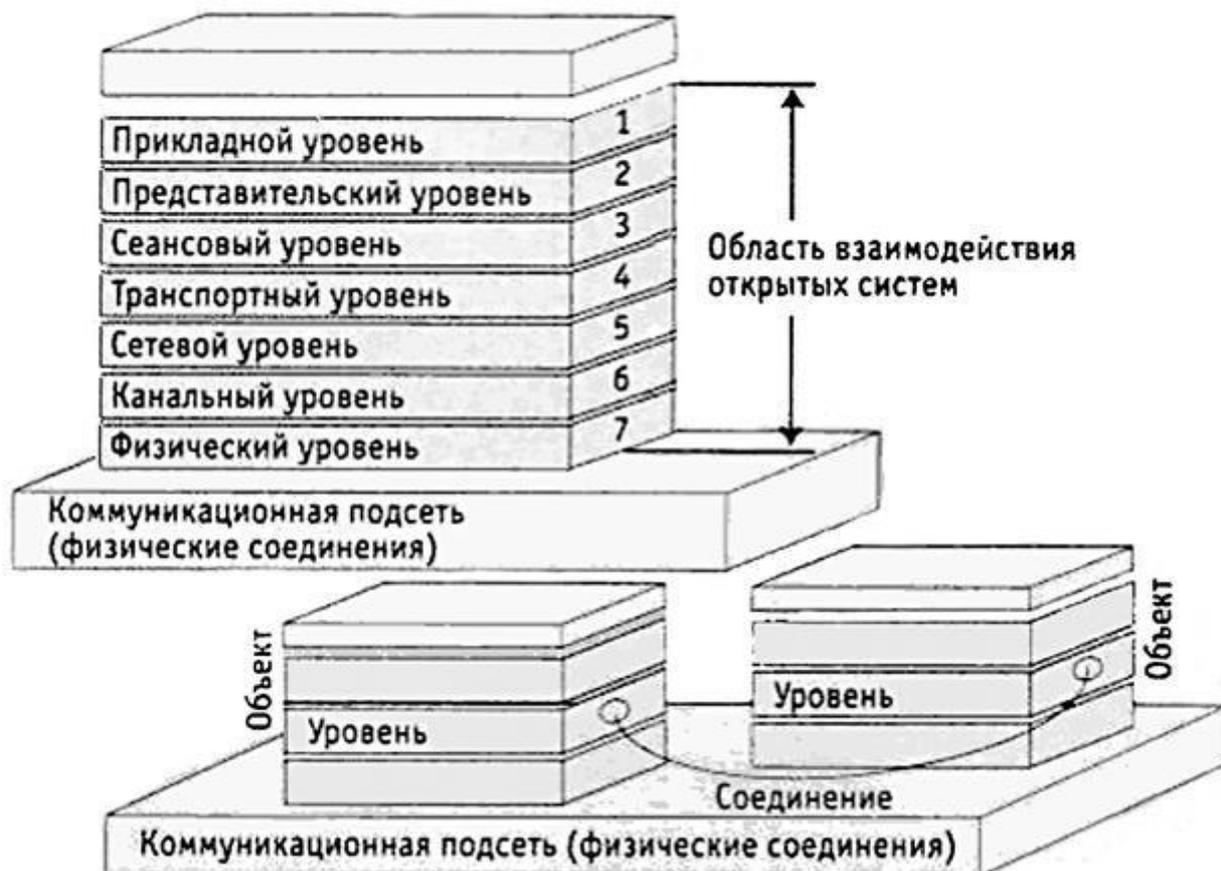


Рис. 4.1. Семиуровневая модель взаимодействия информационных систем

Эталонная модель среды открытых систем (OSI/RM) определяет разделение любой информационной системы на приложения (прикладные программы и программные комплексы) и среду, в которой эти приложения функционируют. Между приложениями и средой определяются стандартизованные интерфейсы (API), которые являются необходимой частью профилей любой открытой системы. Кроме того, в профилях ИС могут быть определены унифицированные интерфейсы взаимодействия функциональных частей друг с другом и интерфейсы взаимодействия между компонентами среды ИС.

Более подробно о применении технологии и моделей открытых систем будет рассказано в "лекции 18".

Модель создания информационной системы

Методологически важно наряду с рассмотренными моделями среды ИС предложить модель создания ИС, которая имела бы те же аспекты функциональных групп компонентов (пользователи, функции, данные, коммуникации). Такой подход обеспечит сквозной процесс проектирования и сопровождения на всех стадиях эксплуатации ИС и возможность обоснованного выбора стандартов на разработку систем и документирование проектов.

Компания является сложной онтологической (понятийной) структурой, состоящий из определенной совокупности сущностей и взаимосвязей (рис. 4.2).



Рис. 4.2. Онтологическое поле современной компании

Взаимодействия между её элементами, определяемые бизнес-логикой и закреплённые в наборе бизнес-правил, и является деятельностью компании. Информационная система "отражает" логику и правила, организуя и преобразуя информационные потоки, автоматизирует процессы работы с данными и информацией и визуализирует результаты в виде наборов отчетных форм. Поэтому для начала следует создать бизнес-модель предприятия, которая является отображением предприятия и его информационно-управляющей системы.

При создании модели формируется "язык общения" руководителей предприятия, консультантов, разработчиков и будущих пользователей, позволяющий выработать единое представление о том, ЧТО и КАК должна делать система управления предприятием (корпоративная система управления). Такая бизнес-модель — осязаемый результат, с помощью которого можно максимально конкретизировать цели внедрения ИС и определиться со следующими параметрами проекта:

- основные цели бизнеса, которые можно достичь посредством автоматизации процессов;
 - перечень участков и последовательность внедрения модулей ИС;
 - фактическая потребность в объемах закупаемого программного и аппаратного обеспечения;
 - реальные оценки сроков развертывания и запуска ИСУ;
 - ключевых пользователей ИС и уточненный список членов команды внедрения;
 - степень соответствия выбранного вами прикладного программного обеспечения специфике бизнеса вашей компании.
- В основе модели всегда лежат бизнес-цели предприятия, полностью определяющие состав всех базовых компонентов модели:
- бизнес-функции, описывающие ЧТО делает бизнес;
 - основные, вспомогательные и управленческие процессы, описывающие КАК предприятие выполняет свои бизнес-функции;
 - организационно-функциональную структуру, определяющую ГДЕ исполняются бизнес-функции и бизнес-процессы;
 - фазы, определяющие КОГДА (в какой последовательности) должны быть внедрены те или иные бизнес-функции;
 - роли, определяющие КТО исполняет бизнес-функции и КТО является "хозяйном" бизнес-процессов;
 - правила, определяющие связь и взаимодействие между всеми ЧТО, КАК, ГДЕ, КОГДА и КТО.

После построения бизнес-модели (или параллельно с этим) можно приступать к формированию модели проектирования, реализации и внедрения самой ИС (рис. 4.3).



Рис. 4.3.

Опыт создания и использования "заказных" ИС позволяет условно выделить следующие основные этапы их жизненного цикла:

- определение требований к системе и их анализ — определение того, что должна делать система;
- проектирование — определение того, как система будет делать то, что она должна делать; проектирование это, прежде всего, спецификация подсистем, функциональных компонентов и способов их взаимодействия в системе;
- разработка — создание функциональных компонентов и отдельных подсистем, соединение подсистем в единое целое;
- тестирование — проверка функционального соответствия системы показателям, определенным на этапе анализа;
- внедрение — установка и ввод системы в действие;
- функционирование — штатный процесс эксплуатации в соответствии с основными целями и задачами ИС;
- сопровождение — обеспечение штатного процесса эксплуатации системы на предприятии заказчика.

Определение требований к системе и анализ является первым этапом создания ИС, на котором требования заказчика уточняются, согласуются, формализуются и документируются. Фактически на этом этапе дается ответ на вопрос: "Для чего предназначена и что должна делать информационная система?". Именно здесь лежит ключ к успеху всего проекта.

Целью системного анализа является преобразование общих, расплывчатых знаний об исходной предметной области (требований заказчика) в точные определения и спецификации для разработчиков, а также генерация функционального описания системы. На этом этапе определяются и специфицируются:

- внешние и внутренние условия работы системы;
- функциональная структура системы;
- распределение функций между человеком и системой, интерфейсы;
- требования к техническим, информационным и программным компонентам системы;
- требования к качеству и безопасности ;
- состав технической и пользовательской документации;
- условия внедрения и эксплуатации.

Разработка перечисленных выше спецификаций при создании ИС, предназначенной для автоматизации управленческих процессов, в общем случае проходит четыре стадии.

Первая стадия анализа — структурный анализ предприятия — начинается с исследования того, как организована система управления предприятием, с обследования функциональной и информационной структуры системы управления, определения существующих и возможных потребителей информации.

По результатам обследования аналитик на первой стадии строит обобщенную логическую модель исходной предметной области, отображающую ее функциональную структуру, особенности основной деятельности и информационное пространство, в котором эта деятельность осуществляется (рис. 4.4). На этом материале аналитик строит функциональную модель "Как есть" (As Is).

Вторая стадия работы, к которой обязательно привлекаются заинтересованные представители заказчика, а при необходимости и независимые эксперты, состоит в анализе модели "Как есть", выявлении ее недостатков и узких мест, определение путей совершенствования системы управления на основе выделенных критериев качества.

Третья стадия анализа, содержащая элементы проектирования, — создание усовершенствованной обобщенной логической модели, отображающей реорганизованную предметную область или ее часть, которая подлежит автоматизации — модель "Как должно быть" (As To Be).

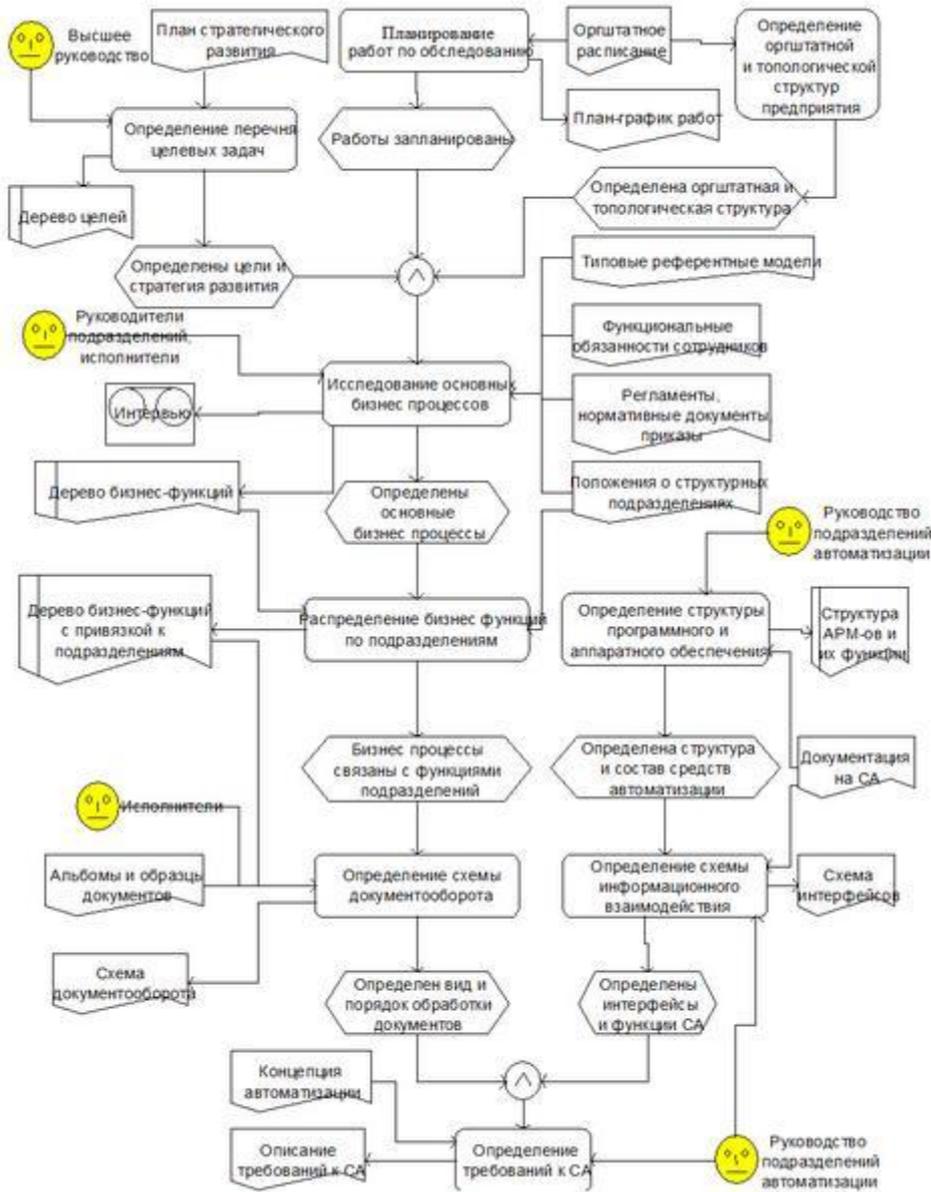


Рис. 4.4. Схема обследования предприятия

Заканчивается процесс (четвертая стадия) разработкой "Карты автоматизации", представляющей собой модель реорганизованной предметной области, на которой обязательно обозначены "границы автоматизации".

В большинстве случаев модель "Как есть" улучшается системным аналитиком за счет устранения очевидных несоответствий и узких мест, а полученный таким образом вариант модели рассматривается в дальнейшем в качестве предварительной модели "Как должно быть", которая впоследствии дополняется в соответствии со стратегией развития предприятия (рис. 4.5).



Рис. 4.5. Стадии построения модели информационной системы

На стадии анализа требований к проектируемой системе и вводятся:

- классы пользователей и соответствующие диаграммы бизнес-транзакций;
- модели (диаграммы) процессов прикладной деятельности и соответствующие перечни функциональных задач ИС;

- классы объектов предметной области и соответствующие диаграммы "сущность-связь", отражающие информационную модель этой предметной области;
- топология расположения подразделений и пользователей, обслуживаемых данной ИС;
- параметры защиты данных, информации и самой системы.

Основным документом, отражающим результаты работ первого этапа создания ИС, является техническое задание на проект (разработку), содержащее, кроме вышеперечисленных определений и спецификаций, также сведения об очередности создания системы, сведения о выделяемых ресурсах, директивных сроках проведения отдельных этапов работы, организационных процедурах и мероприятиях по приемке этапов, защите проектной информации и т. д.

Следующий этап — проектирование. В реальных условиях проектирование — это поиск, моделирование способа разработки, который удовлетворяет требованиям функциональности системы средствами имеющихся технологий с учетом заданных начальных условий и ограничений. Проектирование информационных систем всегда начинается с определения цели проекта. Основная задача любого успешного проекта заключается в том, чтобы на момент запуска системы и в течение всего времени ее эксплуатации можно было обеспечить:

- требуемую функциональность системы и степень адаптации к изменяющимся условиям ее функционирования;
- требуемую пропускную способность системы и минимальное время реакции системы на запрос;
- безотказную работу системы в требуемом режиме, готовность и доступность системы для обработки запросов пользователей;
- простоту эксплуатации и сопровождения системы;
- необходимую безопасность данных и права доступа пользователей.

Производительность и надёжность являются главными факторами, определяющими эффективность системы. Хорошее проектное решение служит основой высокопроизводительной системы.

Проектирование информационных систем охватывает три основные области:

- проектирование структур данных, которые будут реализованы в базе данных;
- проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;
- проектирование конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры, параллельной обработки, распределенной обработки данных и т. п.

На основе результатов системного анализа на стадии предварительного проекта разрабатываются:

- проект программно-аппаратной реализации, проект пользовательских интерфейсов и технологии работы пользователей в системе;
- архитектура распределенной системы и спецификации телекоммуникационной сети;
- модели (диаграммы) потоков данных;
- функциональные блок-схемы прикладного и системного программного обеспечения (последние — в соответствии с принятыми моделями среды ИС и профилями стандартов).

Стадия предварительного проекта может предусматривать прототипирование фрагментов, важных с точки зрения пользователя для проверки их соответствия требованиям на ранней фазе разработки.

На стадии детального проектирования разрабатываются:

- комплексы функциональных программ ИС и проект реализации среды ИС;
- структуры данных, средства ведения баз данных;
- сетевые адреса, протоколы телекоммуникаций и другие компоненты среды обмена информацией, включаемые в состав проектируемой ИС;
- правила разграничения доступа пользователей и средства их реализации.

Стадия реализации ИС предусматривает разработку и тестирование компонентов и комплексное тестирование системы.

Стадия эксплуатации и сопровождения предусматривает контроль функционирования ИС, внесение требуемых изменений в информационную базу в процессе текущей работы и модернизацию функций ИС силами прикладных специалистов с помощью инструментальных средств, встроенных в систему.

Этапы разработки, тестирования, внедрения, эксплуатации и сопровождения ИС объединяются термином — реализация. Реализация ИС является чрезвычайно сложным многоаспектным процессом, осуществляемым на базе совокупностей (профилей) гармонизированных международных стандартов, спецификаций и соглашений. Такая практика является залогом того, что создаваемая информационная система будет реализована как "открытая система". Иными словами такая ИС будет масштабируема, мобильна, переносима, обладать дружественными интерфейсами и т. д.

Жизненный цикл ИС формируется в соответствии с принципом нисходящего проектирования и, как правило, носит спирально-итерационный характер. Реализованные этапы, начиная с самых ранних, циклически повторяются в соответствии с изменениями требований и внешних условий, введением дополнительных ограничений и т. п. На каждом этапе жизненного цикла порождается определенный набор технических решений и документов, при этом для каждого этапа исходными являются документы и решения, принятые на предыдущем этапе. Жизненный цикл ИС заканчивается, когда прекращается её программное и техническое сопровождение.

4.2. Реинжиниринг бизнес-процессов

Внедрение информационных технологий и реализованных на их основе информационных систем в повседневную деятельность предприятия дает ему тактические и долгосрочные преимущества в бизнесе. Стремление руководства к использованию ИТ может остаться лишь благими намерениями, если оно не будет следовать сложившимся требованиям и правилам разработки, проектирования и внедрения ИТ. Выше говорилось о базовых требованиях к стандартизации объектов и функциональных задач, без которых реализуемая система не будет являться открытой системой, что приведет впоследствии к многочисленным проблемам при ее внедрении и эксплуатации.

Следование требованиям стандартов при разработке ИС автоматически приводит к тому, чтобы само предприятие — внешняя среда для ИС — также отвечало необходимым требованиям: определение и стандартизация классов пользователей и объектов, топология потоков данных и работ, архитектура наследуемых и разрабатываемых подсистем, состояние бизнес-процессов и т. д.

Бизнес-процесс представляет собой систему последовательных, целенаправленных и регламентированных видов деятельности, в которой посредством управляющего воздействия и с помощью определенных ресурсов за определенное время входы процесса преобразуются в выходы — в результаты, представляющие ценность для потребителя и приносящие прибыль изготовителю. Стандартный бизнес-процесс в масштабах предприятия реализуется в виде сети основных, вспомогательных, поддерживающих и управленческих процессов (рис. 4.6).

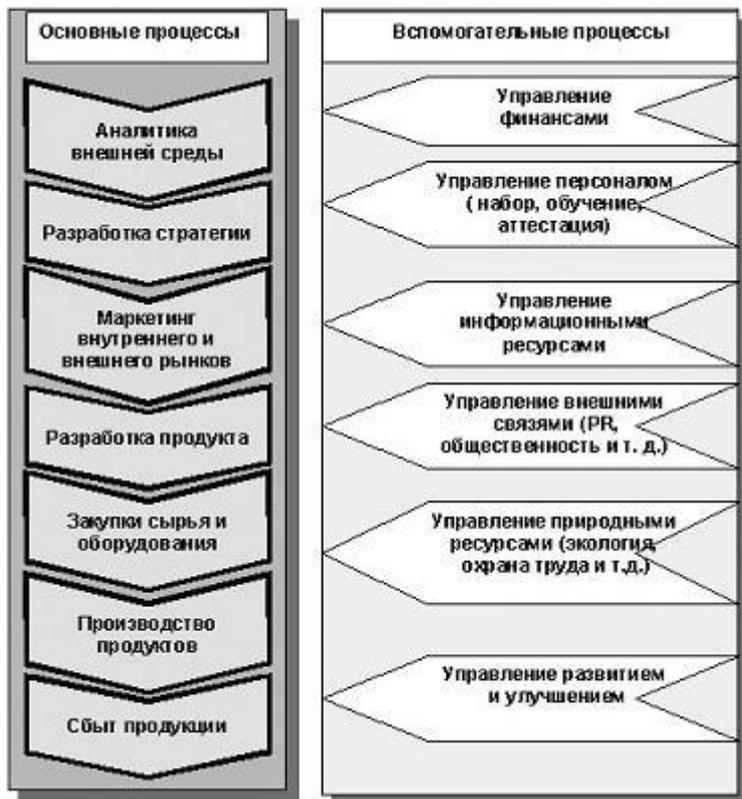


Рис. 4.6. Содержание стандартного бизнес-процесса предприятия

При этом разделение на основные и вспомогательные процессы в определяющей степени зависит от предметной области и направления деятельности предприятия: для производственной компании, например, деятельность юридического отдела является вспомогательной, а для юридической или консалтинговой фирмы — основной. Идентификация процессов является обязательным условием, без реализации которого невозможна информатизация деятельности.

Руководители предприятия, решившиеся на внедрение ИТ, должны твердо усвоить — начало работ по проектированию информационной системы чаще всего влечет за собой обязательный реинжиниринг бизнес-процессов! Реинжиниринг представляет собой множество методик и рекомендаций, среди них нужно выбрать те, которые наилучшим образом удовлетворяют поставленным целям.

Реинжиниринг бизнес-процессов — это совокупность методов и действий, служащих для перепроектирования процессов в соответствии с изменившимися условиями внешней и внутренней среды и/или целями бизнеса.

Существует несколько базовых правил, которых следует придерживаться в процессе проведения реинжиниринга:

- разработка последовательных пошаговых процедур для перепроектирования процессов;
- использование в проектировании стандартных языков и нотаций;
- наличие эвристических и прагматических показателей, позволяющих оценить или измерить степень соответствия перепроектированного процесса или функциональности заданным целям;
- подход к решению частных задач и к их совокупности должен быть системным;
- даже небольшое улучшение должно давать быстрый положительный эффект.

Реинжиниринг деловых процессов и функций начинается с пересмотра целей предприятия, его структуры, анализа потребностей внутренних пользователей и рынка, производимых продуктов и услуг (рис. 4.7).

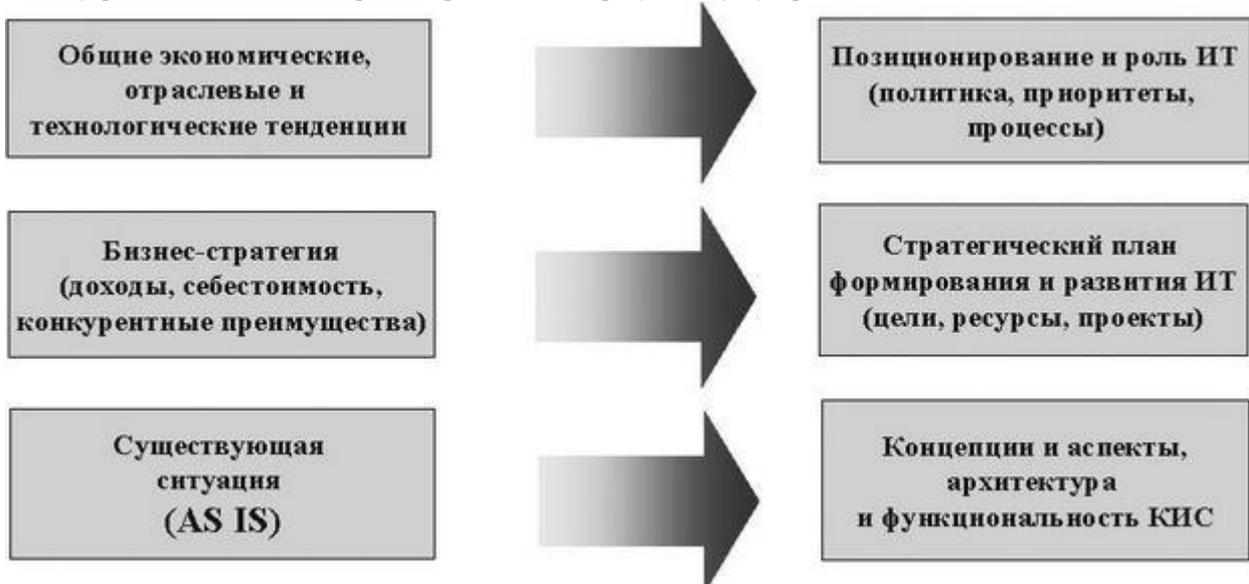


Рис. 4.7. Системный подход к реинжинирингу процессов

Перепланирование целей и задач предполагает пересмотр политики предприятия и ответа на следующие вопросы:

- Какие новые вызовы предъявляют нам изменившиеся условия бизнеса?
- Что представляет предприятие сейчас, и что мы хотим от него в будущем?
- Каких именно потребителей мы обслуживаем, насколько мы удовлетворяем их требования и ожидания, и что нужно сделать для привлечения новых?
- Какие именно показатели определяют эффективность деятельности предприятия, производительность труда и качество продукта, является ли это определение полным и адекватным?
- Какие именно информационные технологии и средства помогут нам в этом?

Для ответа на эти ключевые вопросы необходимо в первую очередь провести детальное описание бизнес-архитектуры предприятия, его бизнес-логики, построить функциональную модель взаимодействия бизнес-процессов, ресурсов и персонала и отразить её в архитектуре ИС, содержании модулей информационных подсистем и визуализации форм представления информации. Необходимо также иметь методики и инструменты реорганизации процессов, решения прикладных задач и управления проектом реинжиниринга (рис. 4.8). Описание бизнес-архитектуры предприятия позволяет:

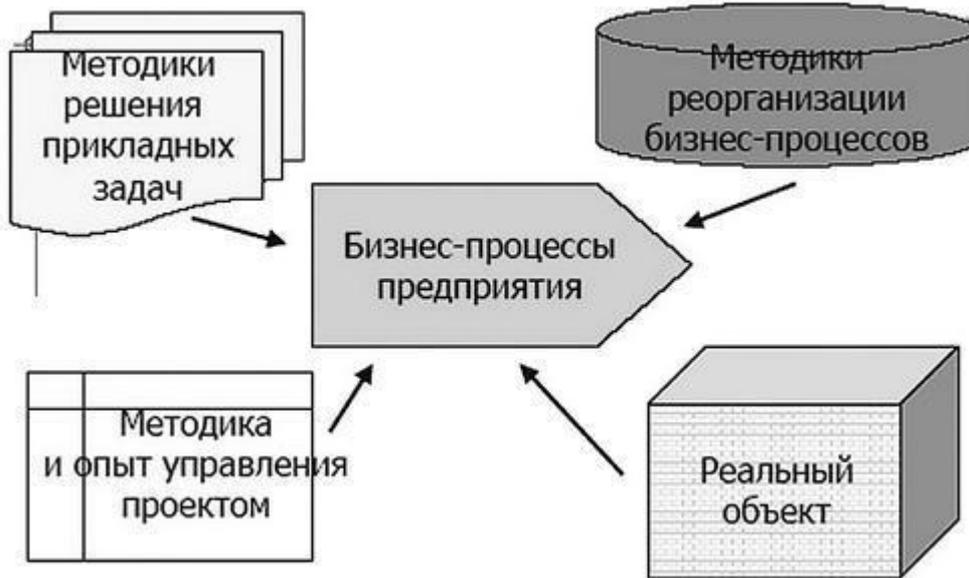


Рис. 4.8. Базовая основа улучшения процесса

- построить схему основных потоков данных, работ, движения финансов и документов;
- понять, как информация распределяется между подразделениями, и кто является конечным пользователем в том или ином бизнес-процессе;
- описать взаимодействие процессов и модулей информационной системы;
- определить критическую важность видов информации для конкретных уровней управления предприятием;
- выявить дублированные структуры и связи.

Результатом такого описания является:

- уточненная карта сети процессов;
- матрица взаимосвязей процессов и подразделений, вовлеченных в эти процессы;
- информация о том, какие системы автоматизации существуют, при выполнении каких операций используются, где и какие данные используются, какие системы автоматизации и информатизации необходимо разработать;
- функциональные схемы потоков данных (Data Flow), работ (Work Flow), финансовых потоков (Cash Flow), потоков управленческих воздействий (Control Flow) и документооборота (Doc Flow).

Функциональная модель поможет составить точные спецификации всех операций, процедур и взаимосвязей между ними. Такая модель, если она построена правильно, обеспечивает исчерпывающее описание о функционирующем процессе и обо всех имеющихся в нем потоках информации. Эта модель описывает состояние "Как есть" (As Is). По результатам анализа возможных путей улучшения от реальной модели нужно перейти к модели, характеризующей улучшения — модель "Как будет" (As To Be), вариант — "Как должно быть" (рис. 4.9).



Рис. 4.9. Схема реинжиниринга бизнес-процесса

Функциональное моделирование является достаточно серьезной проблемой, полнота и соответствие построенной модели зависят как от средств моделирования, так и от квалификации специалистов, выполняющих это моделирование.

Реинжиниринг бизнес-процессов является сложным и многоаспектным проектом, требующим тщательного планирования и проработки деталей. В таблице 4.1 показаны основные этапы реинжиниринга.

Таблица 4.1. Основные этапы реинжиниринга

Этап	Мероприятия
Планирование и начало работ	<ul style="list-style-type: none"> Выявление главных причин проведения реформы на предприятии и оценка последствий отказа от такой реформы Выявление важнейших процессов, требующих реинжиниринга Выявление единомышленников среди руководства и создание рабочей группы из представителей администрации Обеспечение поддержки проекта руководством Подготовка плана проекта: определение объема, обозначение измеримых целей, выбор методологии, составление подробного графика Согласование целей и объемов проекта с руководством Формирование группы реинжиниринга Выбор консультантов или внешних экспертов Проведение вводного совещания Доведение целей проекта до руководителей низшего звена; начальное информирование всей организации Обучение группы реинжиниринга Подготовка плана и начало работ
Исследования	<ul style="list-style-type: none"> Аналитическое исследование опыта компаний с подобными процессами Опрос клиентов и контрольных групп для выявления существующих и будущих требований Опрос служащих и руководителей для выявления вопросов; мозговой штурм Поиск в литературе и прессе данных о тенденциях в отрасли и о чужом опыте Оформление подробных документов на исходные процессы и сбор рабочих данных; выявление недоработок Обзор изменений и вариантов технологий Опрос владельцев и представителей руководства Посещение кружков и семинаров Сбор данных от внешних экспертов и консультантов
Проектирование	<ul style="list-style-type: none"> Мозговой штурм и выработка новаторских идей; упражнения по творческому мышлению, чтобы "снять шоры" Проработка сценариев "а что, если?" и применение "шаблонов успеха" других компаний Создание при помощи специалистов 3-5 моделей; разработка комплексных моделей, в которых собрано лучшее от каждой из предыдущих Создание картины идеального процесса Определение моделей нового процесса и их графическое представление Разработка организационной модели в сочетании с новым процессом Определение технологических требований; выбор платформы для новых процессов Выделение краткосрочных и долгосрочных мер
Утверждение	<ul style="list-style-type: none"> Анализ затрат и преимуществ; расчет прибыли на капитал Оценка влияния на клиентов и служащих; оценка влияния на конкурентоспособность Подготовка официального документа для высшего руководства Проведение обзорных совещаний для ознакомления и утверждения деталей проекта оргкомитетом и высшим руководством
Внедрение	<ul style="list-style-type: none"> Завершение подробной разработки процессов и организационных моделей; определение новых рабочих обязанностей Разработка систем поддержки

	Реализация предварительных вариантов и первичные испытания
	Ознакомление работников с новым вариантом; разработка и осуществление плана реформы
	Разработка поэтапного плана; внедрение как таковое
	Разработка плана обучения; обучение работников новым процессам и системам
Последующие мероприятия	Разработка мероприятий по периодической оценке; определение итогов нового процесса; внедрение программы непрерывного совершенствования нового процесса
	Предоставление окончательного отчета оргкомитету и администрации

4.3. Отображение и моделирование процессов

На сегодняшний день получили распространение три основные методологии функционального моделирования (и сопутствующий им инструментарий): IDEF (Integrated DEFinition), UML (Unified Modeling Language) и ARIS (Architecture of Integrated Information Systems). Для каждой из них существуют определенные программные продукты, которые помимо разработки позволяют проводить преобразования и операции для последующей работы с полученными моделями. Наибольшее распространение сегодня получили методологии IDEF и программный продукт BPWin, содержащий методологии IDEF0, IDEF3, DFD (Data Flow Diagrams) и ERWin (IDEF1x) от компании Computer Associates.

История методологии IDEF начинается с 70-х годов XX века с методологии SADT (Structured Analysis and Design Technique), разработанной Дугласом Россом (Softtech INC). Изначально SADT применялось Министерством Обороны США для практического моделирования процессов в рамках программы ICAM (Integrated Computer Aided Manufacturing). Принципиальным требованием при разработке рассматриваемого семейства методологий была возможность эффективного обмена информацией между всеми специалистами — участниками программы ICAM (Icam DEFinition). В последующем эта методология была трансформирована в стандарт IDEF0 (Function Modeling, FIPS № 183). Семейство IDEF включает уже упомянутые IDEF3 (Process Description Capture) и IDEF1x (Data Modeling, FIPS № 184).

После опубликования стандартов они были успешно применены в самых различных областях бизнеса, показав себя эффективным средством анализа, конструирования и отображения бизнес-процессов (к слову сказать, он активно применяется и в отечественных госструктурах, например в Государственной налоговой инспекции). Более того, собственно с широким применением IDEF (и предшествующей методологии SADT) и связано возникновение основных идей популярного ныне понятия "реинжиниринг бизнес-процессов" (Business Process Reengineering — BPR).

Информационный процесс — это устойчивый процесс (последовательность работ и действий с данными и информацией), относящийся к сопровождению производственно-хозяйственной деятельности компании и обычно ориентированный на информационное обслуживание создания новой стоимости. Бизнес-процесс включает в себя иерархию взаимосвязанных функциональных действий, реализующих одну (или несколько) бизнес-целей компании и отражающий результаты в информационной системе, например, информационное обеспечение управления и анализа выпуска продукции или ресурсное обеспечение выпуска продукции (под продукцией здесь понимают товары, услуги, решения, документы).

Работа с использованием метода IDEF начинается с постановки цели моделирования. Мировой опыт свидетельствует, что ошибки при постановке цели приводят в среднем к 50 % неудач в процессе моделирования. Формулирование цели изначально направляет работу в заданном направлении, а значит, ограничивает круг вопросов для анализа. Практическая работа начинается с определения контекста (Context, Context Diagram), то есть верхнего уровня системы, в нашем случае — предприятия. После формулировки цели необходимо очертить область моделирования (Scope), которая в последующем будет определять общие направления движения и глубину детализации (Decomposition). Собственно, сама методология IDEF определяет стандартизованные объекты для работы и отображения. Например, к таковым относятся функция (Activity), интерфейсная дуга (Arrow), заметка (Note) а также способ их расположения и трактования (Semantics).

В последнее время на российском рынке появился программный продукт Business Studio, который специально создан для работы с методами IDEF и обладает интуитивным и дружественным интерфейсом (User-friendly Interface).

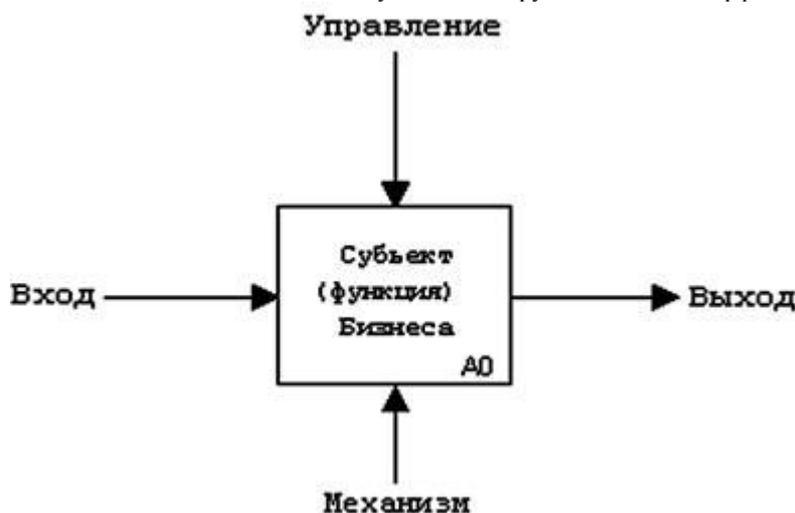


Рис. 4.10. Базовый блок методологии IDEF0

В основе нотации и методологии IDEF0 лежит понятие "блока", то есть прямоугольника, который выражает некоторую функцию бизнеса (рис. 4.10). В соответствии со стандартом функция должна быть выражена глагольным оборотом В IDEF0 роли сторон прямоугольника (функциональные значения) различны: верхняя сторона имеет значение "управление", левая — "вход", правая — "выход", нижняя — "механизм исполнения".

Вторым элементом методологии и нотации является "поток", называемый в стандарте "интерфейсная дуга". Это элемент, описывающий данные, неформальное управление, или что-либо другое, оказывающее влияние на функцию, изображенную блоком. Потоки обозначаются оборотом существительного.

В зависимости от того, к какой стороне блока направлен поток, он, соответственно, носит название "входной", "выходной", "управляющий". Изобразительным элементом, представляющим поток, является стрелка. Поток можно интерпретировать как представление объекта, под которым понимается как информационный объект, так и реальный физический объект.

Важным фактором является то, что "источником" и "приемником" потоков (то есть, началом и концом стрелки) могут быть, как правило, только блоки. При этом источником может являться только выходная сторона блока, приемником — любая из трех оставшихся. Если же необходимо подчеркнуть внешний характер потока, то может быть применен метод "туннелирования" — скрытие или появление интерфейсной дуги из "туннеля".

И, наконец, "третьим китом" методологии IDEF0 является принцип функциональной декомпозиции блоков, который представляет собой модельную интерпретацию той практической ситуации, что любое действие (тем более такое сложное, как бизнес-процесс) может быть разбито (декомпозировано) на более простые операции (действия, бизнес-функции). Или, другими словами, действие может быть представлено как совокупность элементарных функций.

Пример функциональной модели процесса отгрузки и доставки продукции показан на рис. 4.11.

Степень формализации описания бизнес-процессов может быть различной в зависимости от решаемых при этом задач. Для описания информационных процессов разработан специализированный язык BPEL (Business Process Execution Language). BPEL создан на основе XML для формального описания бизнес-процессов и протоколов их взаимодействия между собой. BPEL расширяет модель взаимодействия Web-служб и включает в эту модель поддержку транзакций.



увеличить изображение

Рис. 4.11. Пример функциональной модели процесса отгрузки и доставки

В настоящее время активно развивается методология BPMS (Business Process Management System) — класс программного обеспечения для управления бизнес-процессами и административными регламентами. (Употребляются также термины BPM-система и просто BPM). Использование BPMS позволяет организовать эффективное взаимодействие между управленцами и ИТ-специалистами, лучше использовать существующие подсистемы и ускорить разработку новых.

Основные функции BPMS — моделирование, исполнение и мониторинг бизнес-процессов. Основываясь на данных мониторинга, предприятия выявляют узкие места и совершенствуют свои бизнес-процессы. Цикл управления замыкается, когда при помощи BPMS измененные бизнес-процессы оперативно внедряются в эксплуатацию.

Современные методы разработки и развития программного обеспечения ИС в полной мере стараются ориентироваться на возможности автоматизированного оперативного внесения изменений. Наиболее сложным оказался процесс стандартизации языка BPEL для унификации использования одних и тех же конструкций программным обеспечением разных производителей. Фирмы IBM и Microsoft определили два довольно-таки схожих языка: WSFL (Web Services Flow Language) и Xlang, соответственно. Рост популярности BPML и открытое движение BPMS к пользователям привело корпорации Intalio Inc., IBM и Microsoft к решению объединить эти языки в новый язык BPEL4WS. В апреле 2003 года корпорации BEA Systems, IBM, Microsoft, SAP и Siebel Systems передали BPEL4WS версии 1.1 в OASIS для стандартизации в Web Services BPEL Technical Committee. Хотя BPEL4WS появился в версиях 1.0 и 1.1, технический комитет WS-BPEL OASIS проголосовал 14 сентября 2004 за то, чтобы назвать спецификацию WS-BPEL 2.0. Это изменение было сделано, чтобы согласовать BPEL с другими стандартами Web-сервисов, которые на основании "Соглашения об именовании" начинаются сочетаниями букв "WS-".

Корпорации Active Endpoints, Adobe, BEA, IBM, Oracle и SAP опубликовали согласованные спецификации BPEL4 People и WS-HumanTask, в которых описывалось, как может быть реализовано в системе и нотациях BPEL взаимодействие процессов с людьми. Предполагается добавление в BPEL семантики в форме WS-HumanTask и других разнообразных дополнений.

4.4. Обеспечение процесса анализа и проектирования ИС возможностями CASE-технологий

Термин CASE (Computer Aided Software/System Engineering) используется в настоящее время в весьма широком смысле. Первоначальное значение термина CASE, ограниченное вопросами автоматизации разработки только лишь программного обеспечения (ПО), в настоящее время приобрело новый смысл, охватывающий процесс разработки сложных ИС в целом.

Теперь под термином CASE-средства понимаются программные средства, поддерживающие процессы создания и сопровождения ИС, включая анализ и формулировку требований, проектирование прикладного программного обеспечения (приложений) и баз данных, генерацию кода, тестирование, документирование, обеспечение качества, конфигурационное управление и управление проектом, а также другие процессы. Таким образом, современные CASE-средства вместе с системным программным обеспечением и техническими средствами поддержки образуют полную среду разработки ИС.

Появлению CASE-технологии и CASE-средств предшествовали исследования в области методологии программирования. Программирование обрело черты системного подхода с разработкой и внедрением языков высокого уровня, методов структурного и модульного программирования, средств визуального моделирования и проектирования на базе языка UML (Unified Modeling Language), средств их поддержки, формальных и неформальных языков описаний системных требований и спецификаций и т. д. Кроме того, появлению CASE-технологии способствовали и такие факторы, как:

- подготовка аналитиков и программистов, восприимчивых к концепциям модульного и структурного программирования;

- широкое внедрение и постоянный рост производительности компьютеров, позволившие использовать эффективные графические средства и автоматизировать большинство этапов проектирования;
- внедрение сетевой технологии, предоставившей возможность объединения усилий отдельных исполнителей в единый процесс проектирования путем использования разделяемой базы данных, содержащей необходимую информацию о проекте. 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 [Вендров А. М., www.citforum.ru/database/case/index.shtml].

CASE-средства позволяют создавать не только продукт, практически готовый к использованию, 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.

Большинство CASE-средств основано на парадигме "методология/метод/нотация/структура/средство".

Методология задает руководящие указания для оценки и выбора проекта разработки ПО, этапы и последовательность работ, правила применения тех или иных методов.

Метод — систематическая процедура или технология генерации описаний компонент ПО (например, описание потоков и структур данных).

Нотации предназначены для описания системы в целом, ее элементов: графы, диаграммы, таблица, блок-схемы, алгоритмы, формальные языки и языки программирования.

Структуры являются средством для реализации структурного анализа и построения структуры конкретной системы.

Средства — технологические и программные инструменты для поддержки и усиления методов.

CASE-технологии обладают следующими основными достоинствами, которые позволяют широко использовать их при разработке информационных систем:

- ускоряют процесс коллективного проектирования и разработки;
- позволяют за короткий срок создать прототип заказанной системы с заданными свойствами;
- освобождают разработчика от рутинной работы, оставляя время для творчества;
- обеспечивают эффективность и качество разрабатываемого ПО за счет автоматизации контроля всего процесса разработки;
- поддерживают сопровождение и развитие системы на высоком уровне.

Следует отметить, что, несмотря на все потенциальные возможности CASE-средств, существует достаточно много примеров их неудачного внедрения, в результате которых CASE-средства становятся "полочным" ПО (Shelfware).

В связи с этим необходимо учитывать следующее:

- CASE-средства не обязательно дают немедленный эффект, он может быть получен только спустя какое-то время;
- реальные затраты на внедрение CASE-средств обычно намного превышают затраты на их приобретение;
- CASE-средства обеспечивают возможности для получения существенной выгоды только после успешного завершения процесса их внедрения, эффективного обучения пользователей и регулярного применения.

Можно также перечислить следующие факторы, усложняющие определение возможного эффекта от использования CASE-средств:

- широкое разнообразие качества и возможностей CASE-средств;
- относительно небольшое время использования CASE-средств в различных организациях и недостаток опыта их применения;
- широкое разнообразие в практике внедрения различных организаций;
- отсутствие детальных метрик и данных для уже выполненных и текущих проектов;
- широкий диапазон предметных областей проектов;
- различная степень интеграции CASE-средств в различных проектах.

Некоторые аналитики считают, что реальная выгода от использования некоторых типов CASE-средств может быть получена только после одно- или двухлетнего опыта. Другие полагают, что воздействие может реально проявиться в фазе эксплуатации жизненного цикла ИС, когда технологические улучшения могут привести к снижению эксплуатационных затрат.

Ниже перечислены основные виды и последовательность работ, рекомендуемые при построении логических моделей предметной области в рамках CASE-технологии анализа системы управления предприятием.

1. Проведение функционального и информационного обследования системы управления (административно-управленческой деятельности) предприятия (рис. 4.2):
 - определение организационно-штатной структуры предприятия;
 - определение функциональной структуры предприятия;
 - определение перечня целевых функций структурных элементов (подразделений и должностных лиц);
 - определение круга и очередности обследования структурных элементов системы управления согласно сформулированным целевым функциям;
 - обследование деятельности выделенных структурных элементов;
 - построение FD-диаграммы системы управления с указанием структурных элементов и функций, реализация которых будет моделироваться на DFD-уровне.
2. Разработка моделей деятельности структурных элементов и системы управления в целом:
 - выделение множества внешних объектов, оказывающих существенное влияние на деятельность структурного элемента;
 - спецификация входных и выходных информационных потоков;
 - выявление основных процессов, определяющих деятельность структурного элемента и обеспечивающих реализацию его целевых функций;
 - спецификация информационных потоков между основными процессами деятельности, уточнение связей между процессами и внешними объектами;

- оценка объемов, интенсивности и других необходимых характеристик информационных потоков;
 - разработка иерархии диаграмм потоков данных, образующих функциональную модель деятельности структурного элемента;
 - объединение DFD-моделей структурных элементов в единую модель системы управления предприятием.
3. Разработка информационных моделей структурных элементов и модели информационного пространства системы управления:
- определение сущностей модели и их атрибутов;
 - проведение атрибутного анализа и оптимизация сущностей;
 - идентификация отношений между сущностями и определение типов отношений;
 - анализ и оптимизация информационной модели;
 - объединение информационных моделей в единую модель информационного пространства.
4. Разработка предложений по автоматизации системы управления предприятия:
- определение границ автоматизации — составление перечня автоматизируемых структурных элементов, разбиение процессов основной деятельности на автоматические, автоматизированные и ручные;
 - составление перечня подсистем и логических АРМов (автоматизированных рабочих мест), определение способов их взаимодействия;
 - разработка предложений по очередности проектирования и реализации подсистем и отдельных логических АРМов, входящих в состав ИС;
 - разработка требований к средствам базового технического обеспечения ИС;
 - разработка требований к средствам базового программного обеспечения ИС.

Логическая модель, отображающая деятельность системы управления предприятия и информационное пространство, в котором эта деятельность протекает, представляют собой "снимок" положения дел (функциональная структура, роли должностных лиц, взаимодействие подразделений, принятые технологии обработки управленческой информации, автоматизированные и неавтоматизированные процессы и т. д.) на момент обследования. Эта модель позволяет понять, что делает и как функционирует предприятие с позиций системного анализа, сформулировать предложения по улучшению ситуации.

Развитие логической модели предметной области, ее последовательное превращение в модель целевой ИС, позволит интегрировать перспективные предложения руководства и ведущих сотрудников предприятия, экспертов и системных аналитиков, сформировать видение новой, реорганизованной и автоматизированной деятельности предприятия (рис. 4.12).



Рис. 4.12. Модель системы в технологическом CASE-решении

Построенная модель является законченным результатом по следующим причинам.

1. Она включает в себя модель существующей неавтоматизированной технологии, принятой на предприятии. Формальный анализ этой модели позволяет выявить узкие места в управлении предприятием и сформулировать рекомендации по его улучшению (независимо от того, предполагается ли дальнейшая разработка автоматизированной системы или нет).
2. Она независима и отделяема от конкретных разработчиков, не требует сопровождения и может быть безболезненно передана другим лицам. Более того, если по каким-либо причинам предприятие не готово к реализации проекта в данный момент времени, модель может быть "положена на полку" до тех пор, пока в ней не возникнет необходимость.

3. Она позволяет осуществлять эффективное обучение новых работников конкретным направлениям деятельности предприятия, так как соответствующие технологии содержатся в модели.
4. С ее помощью можно осуществлять предварительное моделирование перспективных направлений деятельности предприятия с целью выявления новых потоков данных, взаимодействующих процессов и структурных элементов.
5. Она обеспечивает распространение накопленного опыта на других предприятиях, дает возможность унифицировать административно-управленческую и финансовую деятельность этих предприятий.

Модель является не просто реализацией начальных этапов работы и основанием для формирования технического задания на ее последующие этапы. Она представляет собой самостоятельный результат, имеющий большое практическое значение, так как позволяет дальнейшее применение CASE-технологий для реального проектирования и разработки ИС.

Современные CASE-пакеты имеют широкие возможности инструментального расширения за счёт использования стандартных программных средств, что делает их чрезвычайно удобными при разработке программных и информационных систем (рис. 4.13 и 4.14).

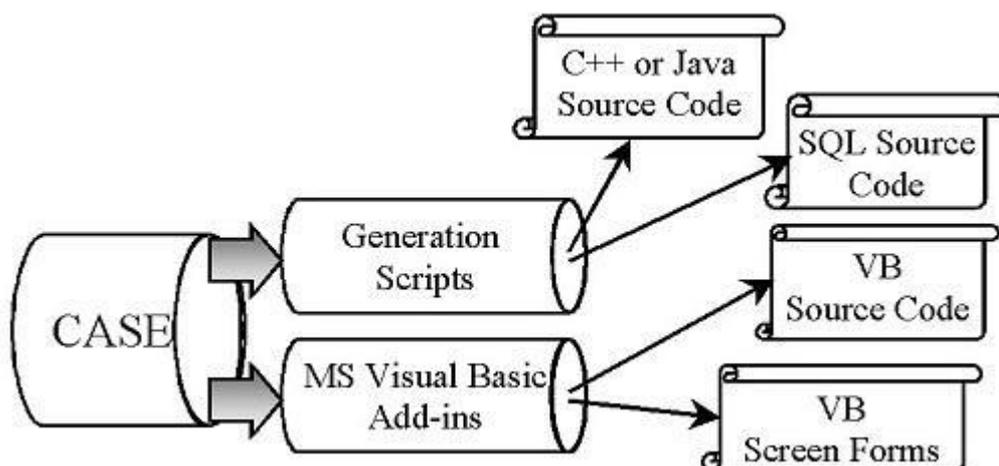


Рис. 4.13.

Для успешного внедрения CASE-средств организация должна обладать следующими качествами:

- Культура. Готовность к внедрению новых процессов и взаимоотношений между разработчиками и пользователями, ИТ/ИС-управленцами и пользователями.
- Управление. Четкое руководство и организованность по отношению к наиболее важным этапам и процессам внедрения.
- Технология. Понимание ограниченности существующих возможностей и способность принять новую технологию.

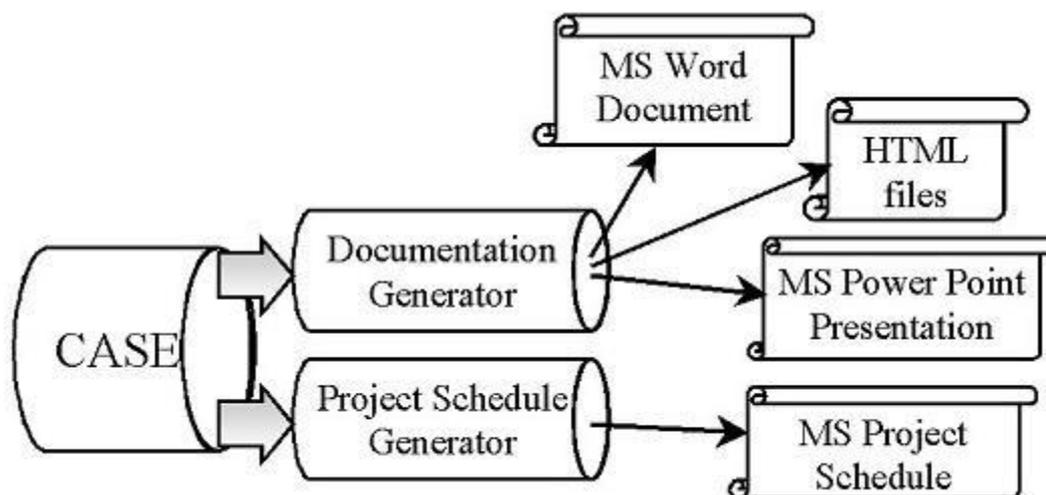


Рис. 4.14.

Если организация не обладает хотя бы одним из перечисленных качеств, то внедрение CASE-средств может закончиться неудачей независимо от степени тщательности следования различным рекомендациям по внедрению.

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

- VPwin — моделирование бизнес-процессов;
- ERwin — моделирование баз данных и хранилищ данных;
- ERwin Examiner — проверка структуры СУБД и моделей, созданных в Erwin;
- ModelMart — среда для командной работы проектировщиков;
- Paradigm Plus — моделирование приложений и генерация объектного кода;
- Rational Rose — моделирование бизнес-процессов и компонентов приложений
- Rational Suite AnalystStudio — пакет для аналитиков данных;
- Oracle Designer (входит в Oracle9i Developer Suite) — высоко функциональное средство проектирования программных систем и баз данных, реализующее технологию CASE и собственную методологию Oracle — CDM. Позволяет команде разра-

ботчиков полностью провести проект, начиная от анализа бизнес-процессов через моделирование к генерации кода и получению прототипа, а в дальнейшем и окончательного продукта. Сложное CASE-средство, имеет смысл использовать при ориентации на линейку продуктов Oracle.

Самым мощным из указанных программных пакетов является пакет Rational Rose (RR) компании IBM-Rational, с помощью которого можно спроектировать и сопровождать весь жизненный цикл разработки программного продукта (рис. 4.15).

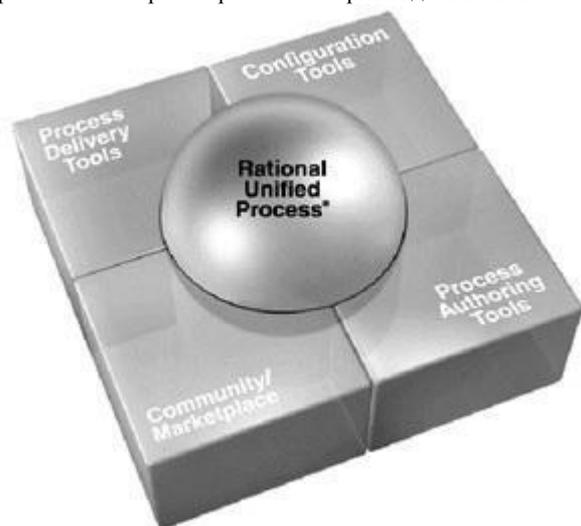
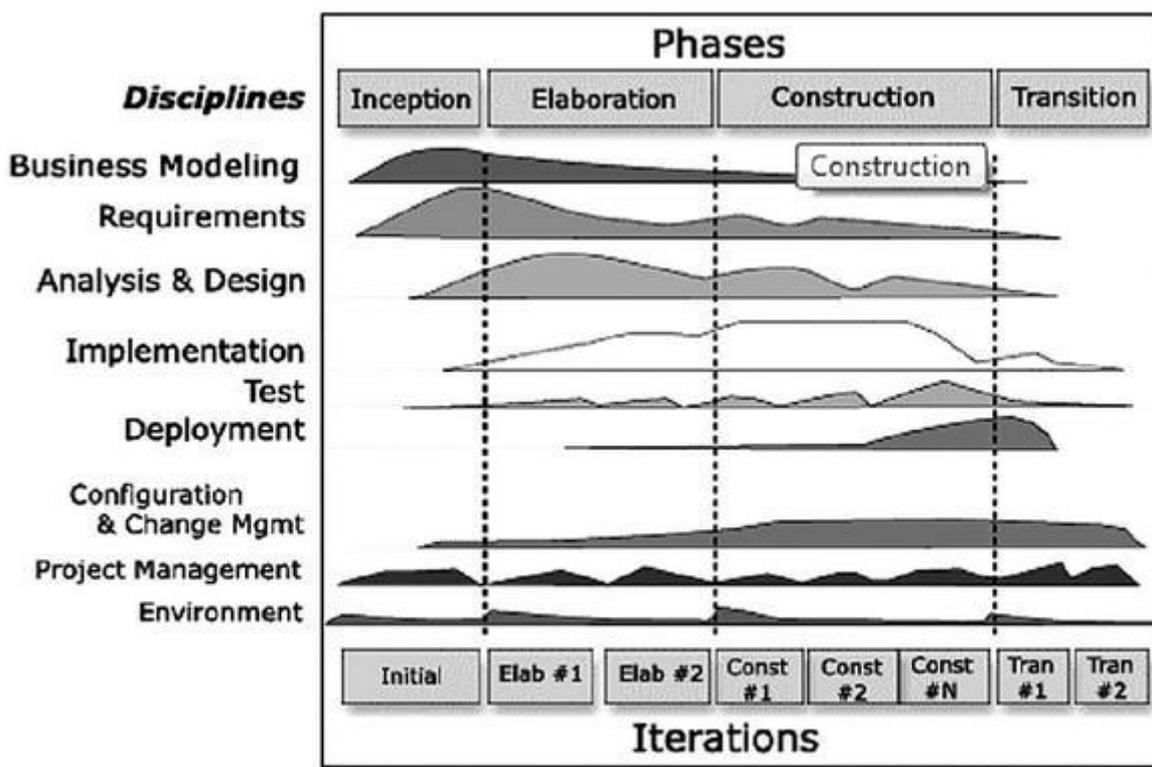


Рис. 4.15. Состав CASE-средства IBM-Rational

Пакет включает набор средств моделирования объектно-ориентированных информационных систем, базирующихся на языке моделирования UML. RR способен решать практически любые задачи в проектировании информационных систем: от анализа бизнес процессов до кодогенерации на определенном языке программирования, позволяет разрабатывать как высокоуровневые, так и низкоуровневые модели, осуществляя тем самым абстрактное либо логическое проектирование (рис. 4.16).



Практическое занятие:

по теме: «Анализ бизнес-процессов подразделения»

Цель: научиться анализировать бизнес-процессы подразделения

По завершению практического занятия студент должен уметь: анализировать бизнес-процессы подразделения

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Проектирование информационных систем: метод. указания к лабораторным работам / сост. С.В. Капустина, А.В. Паршаков; ФГОУ ВПО «СФУ». – Красноярск, 2008. – 80 с.

Задание

Организационная модель предприятия – это принципы формирования подразделений, делегирования полномочий и наделения ответственностью. По сути, организационная модель показывает, как сформировать подразделение.

Структуры управления на многих современных предприятиях были построены в соответствии с принципами управления, сформулированными еще в начале XX века. Наиболее полную формулировку этих принципов дал немецкий социолог Макс Вебер (концепция рациональной бюрократии):

- принцип иерархичности уровней управления, при котором каждый нижестоящий уровень контролируется вышестоящим и подчиняется ему;
- принцип соответствия полномочий и ответственности работников управления месту в иерархии;
- принцип разделения труда на отдельные функции и специализации работников по выполняемым функциям;
- принцип формализации и стандартизации деятельности, обеспечивающий однородность выполнения работниками своих обязанностей и скоординированность различных задач;
- принцип обезличенности выполнения работниками своих функций;
- принцип квалификационного отбора, в соответствии с которым приём и увольнение с работы производится в строгом соответствии с квалификационными требованиями.

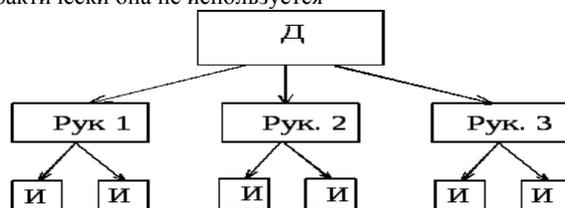
Организационная структура, построенная в соответствии с этими принципами, получила название иерархической или бюрократической структуры.

На практике применяют следующие модели:

- линейная модель: каждый руководитель обеспечивает руководство нижестоящими подразделениями по всем видам деятельности;
- **функциональная модель**: «одно подразделение = одна функция»;
- **линейно-функциональная модель**: ступенчатая иерархическая;
- **процессная модель**: «одно подразделение = один процесс»;
- **матричная модель**: «один процесс или один проект = группа сотрудников из разных функциональных подразделений»;
- **дивизиональная**;
- **множественная (смешанная)**;
- **модель, ориентированная на контрагента**: «одно подразделение = один контрагент (клиент или клиентская группа, поставщик, подрядчик и прочее), модель применяется в случае, если рынок контрагента ограничен. Например, в случае, если число потребителей сильно ограничено, целесообразно применить модель, ориентированную на клиента или клиентскую группу: «одно подразделение = один клиент».

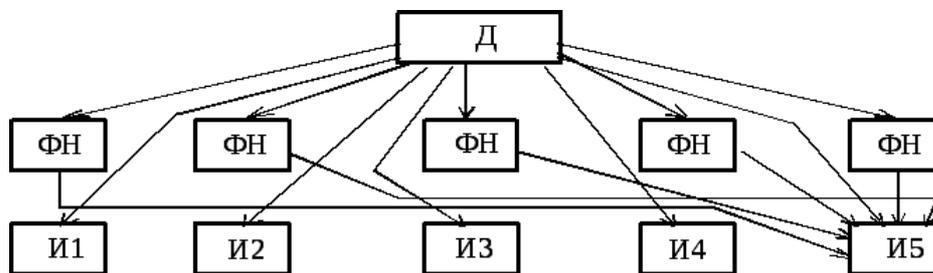
В **линейной** структуре управления каждый руководитель обеспечивает руководство нижестоящими подразделениями по всем видам деятельности. Достоинство - простота, экономичность, предельное единоначалие. Основной недостаток - высокие требования к квалификации руководителей. Сейчас практически не используется.

Функциональная организационная структура - связь административного управления с осуществлением функционального управления (рис.2). На рис.2 административные связи функциональных начальников с исполнителями (И1 - И4) такие же, как и для исполнителя И5 (они не показаны в целях обеспечения ясности рисунка). В этой структуре нарушен принцип единоначалия и затруднена координация. Практически она не используется



Д - директор; Рук. - руководители подразделений; И - исполнители

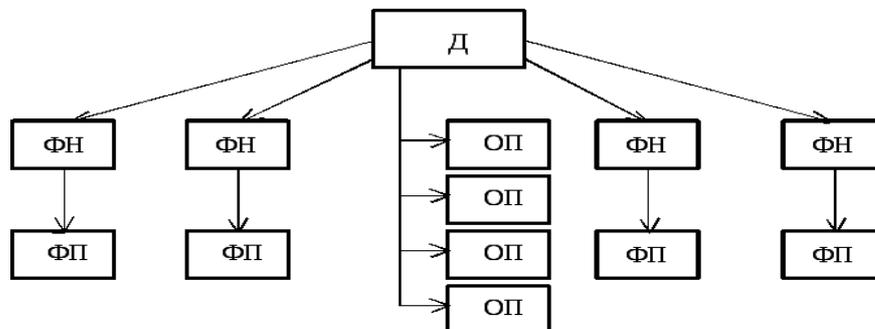
Рис.1 Линейная структура управления



Д- директор; ФН - функциональные начальники; И - исполнители

Рис.2. Функциональная структура управления

Линейно-функциональная структура - линейные руководители являются единоначальниками, а им оказывают помощь функциональные органы. Линейные руководители низших ступеней административно не подчинены функциональным руководителям высших ступеней управления. Она применялась наиболее широко (рис.3).



Д- директор; ФН - функциональные начальники; ФП - функциональные подразделения; ОП - подразделения основного производства

Рис.3. Линейно-функциональная структура управления

Основы линейно-функциональных структур составляет так называемый «шахтный» принцип («принцип колодца») построения и специализация управленческого процесса по функциональным подсистемам организации (маркетинг, производство, исследования и разработки, финансы, персонал и т. д.). По каждой подсистеме формируется иерархия служб («шахта» или «колодец»), пронизывающая всю организацию сверху донизу (Рис. 4). Результаты работы каждой службы оцениваются показателями, характеризующими выполнение ими своих целей и задач. Соответственно строится и система стимулирования и поощрения работников. При этом конечный результат – эффективность и качество работы организации в целом – становится как бы второстепенным, так как считается, что все службы в той или иной мере работают на его получение.

Подобные структуры базируются с одной стороны на линейных полномочиях.

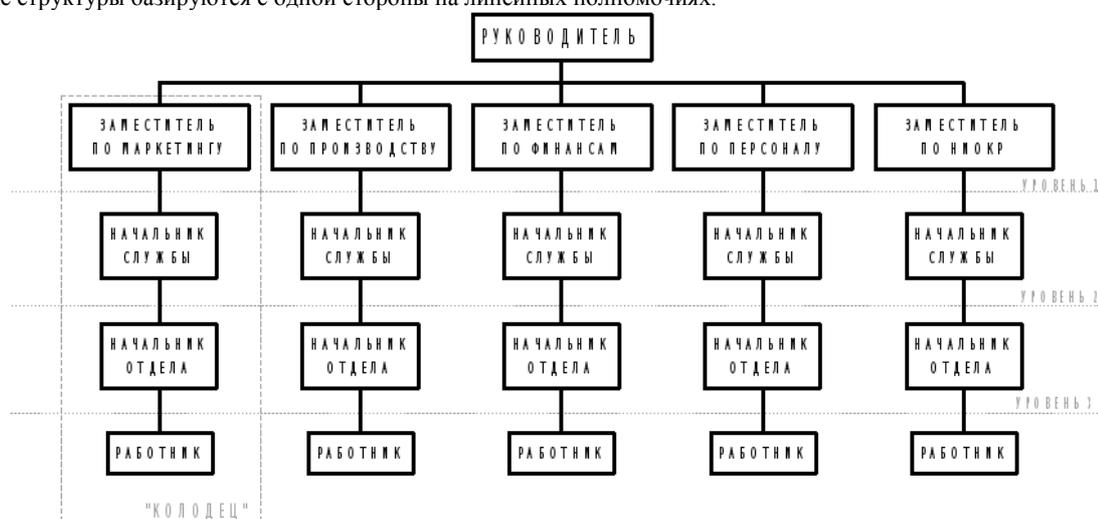


Рис.4. Линейно-функциональная структура управления

Линейные полномочия — это полномочия, которые передаются непосредственно от начальника к подчиненному и далее к другим подчиненным (иерархия уровней управления). Важная особенность такой структуры заключается в единоначалии и Цепи команд. Схема линейной структуры показана на рисунке. Кроме того, в основе подобных структур управления лежит принцип **функциональной департаментализации** (процесс деления организации на отдельные элементы, каждый из которых имеет свою четко определенную, конкретную задачу и обязанности). Конкретные характеристики и черты деятельности того или иного подразделения соответствуют наиболее важным направлениям деятельности всей организации. Совокупность линейности полномочий и функциональной департаментализации в линейно — функциональной структуре обеспечивает преимущества и недостатки такого типа структур.

Преимущества:

- четкая система взаимных связей внутри функций и в соответствующих им подразделениях;
- четкая система единоначалия – один руководитель сосредотачивает в своих руках руководство всей совокупностью функций, составляющих деятельность;
- ясно выраженная ответственность;
- быстрая реакция исполнительных функциональных подразделений на прямые указания вышестоящих.

Недостатки:

- в работе руководителей практически всех уровней оперативные проблемы («текучка») доминируют над стратегическими;
- слабые горизонтальные связи между функциональными подразделениями порождают волокиту и перекладывание ответственности при решении проблем, требующих участия нескольких подразделений;
- малая гибкость и приспособляемость к изменению ситуации;
- критерии эффективности и качества работы подразделений и организации в целом разные и часто взаимоисключающие;
- большое число «этажей» или уровней управления между работниками, выпускающими продукцию, и лицом, принимающим решение;
- перегрузка управленцев верхнего уровня;
- повышенная зависимость результатов работы организации от квалификации, личных и деловых качеств высших управленцев.

Таким образом, можно заключить, что в современных условиях недостатки структуры перевешивают ее достоинства.

Процессная модель. Истоки концепции управления процессами ведут к теориям управления, разработанным еще в XIX веке. В 80-х годах XIX-го века Фредерик Тейлор предложил менеджерам использовать методы процессного управления для наилучшей организации деятельности. В начале 1900-х годов Анри Файоль разработал концепцию реинжиниринга – осуществление

деятельности в соответствии с поставленными задачами путем получения оптимального преимущества из всех доступных ресурсов (Рис.5).

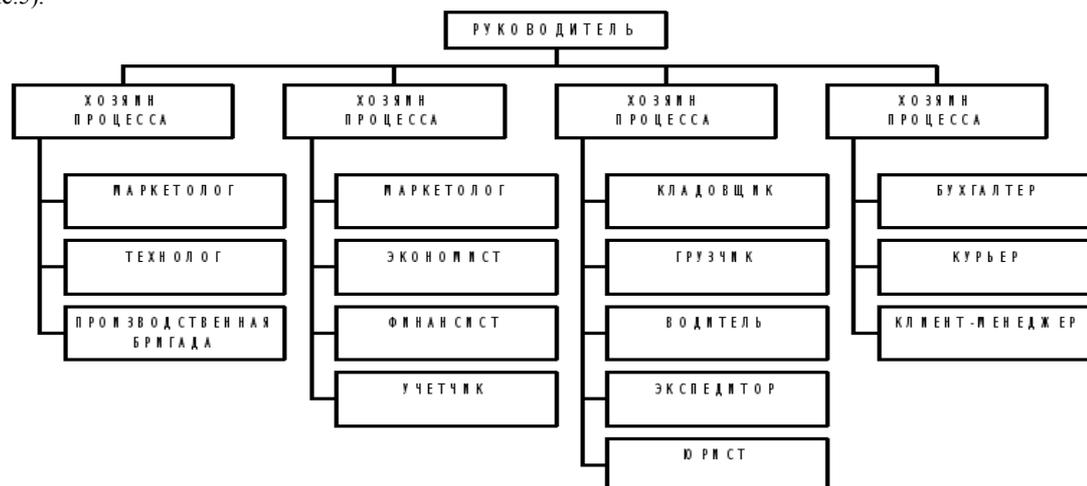


Рис.5. Процессная организационная структура

Процессные системы строятся на базе нескольких базовых принципов:

- *принцип объединения процедур*: выполнявшиеся различными сотрудниками операции, интегрируются в одну, то есть происходит горизонтальное сжатие процесса. Если не удастся привести все шаги процесса к одной работе, то создается команда, отвечающая за данный процесс;
- *принцип неразрывной последовательности*: шаги процесса выполняются в естественном порядке, работа выполняется в том месте, где это целесообразно, смешанными группами, состоящими из работников различной предметной (функциональной) принадлежности или специализации;
- *принцип владельца процесса*: уполномоченный менеджер обеспечивает единую точку контакта, он играет роль буфера между сложным процессом и заказчиком, и ведет себя с заказчиком так, как если бы был ответственным за весь процесс;
- *принцип самостоятельности выбора*: исполнители принимают самостоятельные решения и несут ответственность за получение заданного результата деятельности;
- *принцип горизонтального контроля*: качество результата проверяется его потребителем – следующим элементом процессной цепочки;
- *принцип системности (целостности) управления*: управление затратами происходит по месту их возникновения, система управления издержками строится совместно с организационной структурой, без отрыва от деятельности, «один процесс – одно подразделение – один бюджет».

Преимущества процессных структур:

- четкая система взаимных связей внутри процессов и в соответствующих им подразделениях;
- четкая система единоначалия – один руководитель сосредотачивает в своих руках руководство всей совокупностью операций и действий, направленных на достижение поставленной цели и получение заданного результата;
- наделение сотрудников большими полномочиями и увеличение роли каждого из них в работе компании приводит к значительному повышению их отдачи;
- быстрая реакция исполнительных процессных подразделений на изменение внешних условий;
- в работе руководителей стратегические проблемы доминируют над оперативными;
- критерии эффективности и качества работы подразделений и организации в целом согласованы и сонаправлены.

Недостатки процессной структуры:

- повышенная зависимость результатов работы организации от квалификации, личных и деловых качеств рядовых работников и исполнителей.
- управление смешанными в функциональном смысле рабочими командами – более сложная задача, нежели управление функциональными подразделениями;
- наличие в команде нескольких человек различной функциональной квалификации неизбежно приводит к некоторым задержкам и ошибкам, возникающим при передаче работы между членами команды. Однако потери здесь значительно меньше, чем при традиционной организации работ, когда исполнители подчиняются различным подразделениям компании.

Матричная модель. Матричные структуры совмещают принципы построения функциональных и процессных систем. В этих структурах существуют жестко регламентированные процессы, находящиеся под управлением менеджера процесса. При этом деятельность осуществляется работниками, находящимися в оперативном подчинении менеджера процесса и в административном подчинении руководителя в функциональном «колодце» (Рис.66, 7):

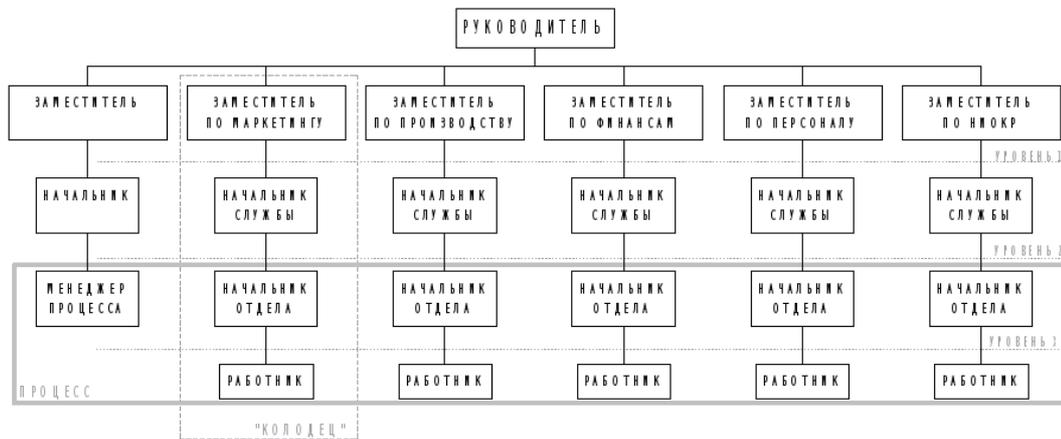


Рис.6. Матричная структура

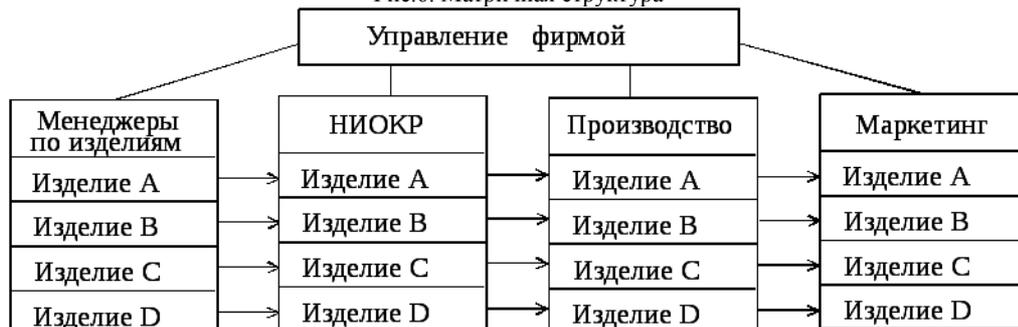


Рис.7. Матричная структура управления, ориентированная на продукт

По существу, роль менеджера процесса состоит в координации действий внутри процесса.

Подобное решение, с одной стороны, не полностью реализует преимущества процессного подхода, а с другой стороны, не полностью устраняет недостатки функциональной системы. На практике матричные структуры хорошо применимы для организации управления проектной деятельностью и мало подходят для регулярного менеджмента, так как содержат в своей природе некоторое двоевластие – процессов и функций.

Проектные структуры - это структурные образования, которые призваны в условиях ограничений по затратам, срокам и качеству работ решить поставленную задачу (проект). Строго говоря, специальные проектные структуры управления применимы к крупномасштабным проектам (рис.8). Специальные проектные структуры управления (проектный офис) создаются на время реализации проектов. На верхнем уровне управления осуществляются централизованные функции (например, учёт, маркетинг, стратегическое планирование и т.д.), все задачи, прямо связанные с выполнением проекта решаются на уровне проектного офиса. В состав временных групп включают необходимых специалистов: инженеров, бухгалтеров, руководителей производства, исследователей, а также специалистов по управлению

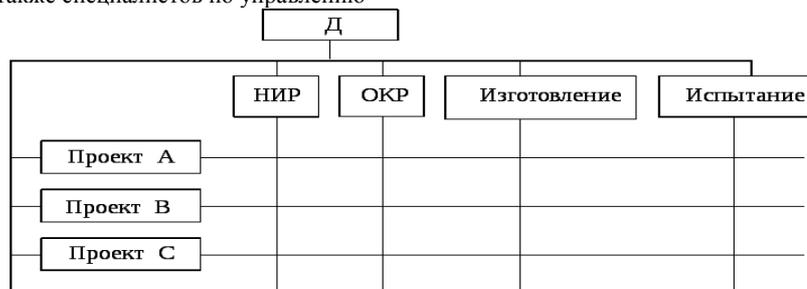


Рис.8. Матричная структура управления по проектам

Достаточно часто деятельность предприятия рассматривается как совокупность выполняемых проектов, каждый из которых имеет фиксированное начало и окончание. Под каждый проект выделяются трудовые, финансовые, промышленные и т.д. ресурсы, которыми распоряжается руководитель проекта. Каждый проект имеет свою структуру, и управление проектом включает определение его целей, формирование структуры, планирование и организацию работ, координацию действий исполнителей. После выполнения проекта структура проекта распадается, ее компоненты, включая сотрудников, переходят в новый проект или увольняются (рис.9).



Рис.9. Проектная структура

Преимущества

1. Комплексный подход к реализации проекта, решению проблемы;
2. Концентрация усилий на решении одной задачи, на выполнении одного конкретного проекта;
3. Большая гибкость структуры;
4. Активизация деятельности руководителей проектов и исполнителей в результате формирования проектных групп;
5. Усиление личной ответственности конкретного руководителя как за проект в целом, так и за его элементы.

Недостатки

1. При наличии нескольких организационных проектов или программ проектные структуры приводят к дроблению ресурсов и заметно усложняют поддержание и развитие производственного и научно-технического потенциала компании как единого целого;
2. От руководителя проекта требуется не только управление всеми стадиями жизненного цикла проекта, но и учет места проекта в сети проектов данной компании;
3. Формирование проектных групп, не являющихся устойчивыми образованиями, лишает работников осознания своего места в компании;
4. При использовании проектной структуры возникают трудности с перспективным использованием специалистов в данной компании;
5. Наблюдается частичное дублирование функций.

Обратите внимание на сходство с дивизиональной структурой. Главное отличие — временный характер структурного образования. Как итог, проекты реализовываются в каждой организации, но далеко не каждая обладает проектной организационной структурой.

Дивизиональная (филиальная) структура изображена на рис. 10. Дивизионы (филиалы) выделяются или по области деятельности, или географически. Дивизиональные структуры управления ориентируются на изделия, рынки сбыта, регионы. При этом обеспечивается:

- относительно большая самостоятельность руководителей дивизионов,
- организация директивных связей по линейному принципу,

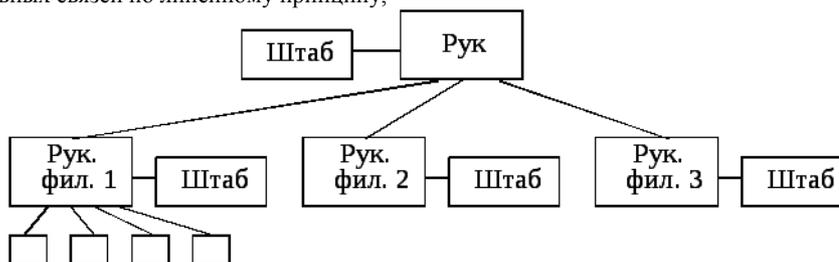


Рис. 10. Дивизиональная структура управления

- относительно мощное использование инструмента координации с технической поддержкой,
- быстрая реакция на изменения рынка,
- освобождение высших руководителей фирмы от оперативных и рутинных решений,
- снижение конфликтных ситуаций вследствие гомогенности целей в дивизионе.

К числу недостатков этой структуры относят:

- относительно высокие затраты на координацию ввиду децентрализации вплоть до отдельного финансирования из бюджета и системы расчетных цен,
- при децентрализации теряются преимущества кооперации, что часто требует централизации выполнения отдельных функций (НИОКР, снабжение и т.д.).

Смешанные структуры. Если применять различные модели организации деятельности в пределах отдельных бизнес-процессов, то можно использовать преимущества той или иной организационной модели. При этом для организации в целом будет применяться процессная организация основных структурных блоков, а в рамках отдельных блоков могут применяться различные модели. Например,

- для организации структурного блока, реализующего бизнес-процесс разработки новых и совершенствования существующих продуктов, целесообразно использовать матричную структуру;
- при определенных условиях для организации процессов воспроизводства ресурсов (зависимость от монополистов-поставщиков), воспроизводства средств производства (использование подрядчиков для выполнения работ), продвижения и продаж (работа с ограниченными клиентскими группами) целесообразно использовать модели, ориентированные на контрагента;
- структура финансовых служб будет выглядеть привычнее при функциональной организации.

Выбор тех или иных субмоделей зависит от специфики и особенности бизнеса.

Практическое занятие:

по теме: «Разработка и оформление предложений по расширению функциональности информационной системы»

Цель: научиться разрабатывать и оформлять предложения по расширению функциональности информационной системы

По завершению практического занятия студент должен уметь: разрабатывать и оформлять предложения по расширению функциональности информационной системы

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Проектирование информационных систем: метод. указания к лабораторным работам / сост. С.В. Капустина, А.В. Паршаков; ФГОУ ВПО «СФУ». – Красноярск, 2008. – 80 с.

Задание

Разработка требований к функциональности информационной системы

На основе анализа информации, полученной на этапе обследования организации, и моделей бизнес-процессов продуктовым ИТ-консультантом разрабатываются требования к функциональности информационной системы. При выполнении этих работ моделирование бизнес-процессов проводится одновременно с фиксированием слабых мест и документированием соответствующих им требований. Для каждого процесса и функции определяются и фиксируются требования, которым должна отвечать информационная система. При этом учитывается множество различных факторов таких, как сложность бизнес-процессов, технологические характеристики, возможности взаимодействия с другими приложениями и ориентация на создание единого информационного пространства организации.

Разрабатываемые требования к функциональности делятся на две большие группы: общие и требования к функциям.

Общие требования включают требования к составу необходимых основных функциональных подсистем и их основным характеристикам и функциям; требования к перечню инструментов для разработки дополнительной функциональности; требования к набору специализированных средств для формирования отчетов произвольной формы на основании данных, хранящихся в системе; требования к перечню средств для загрузки/выгрузки данных; требования к режимам функционирования системы. Требования к функциям содержат детальные функциональные требования к информационной системе по процедурам/функциям бизнес-процессов. Для окончательной формализации и подтверждения требований относительно тех или иных бизнес-процессов продуктовые ИТ-консультанты проводят дополнительные собеседования и собрания, выполняют работы по формализации, документированию и согласованию требований. Результаты работы оформляются в виде раздела отчета, в котором по каждому бизнес-процессу представлена следующая информация:

1. Функциональная модель бизнес-процесса.
2. Описание бизнес-процесса и функциональные требования к процессу в целом (содержательная часть, в т. ч. цель, задачи, требования к исполнению и контролю, распределение ответственности, входная и выходная информация, требования к безопасности).
3. Перечень и описание функций/процедур бизнес-процесса
4. Описание требований к функциональности ИС по всем функциям/процедурам бизнес-процессов.
5. Список входных и выходных документов.

На основе выявленных требований в дальнейшем разрабатывается техническое задание.

Задача формирования требований является наиболее трудной частью работ, выполняемых продуктовым ИТ- консультантом. Это связано с возникающими в процессе выполнения работ такими проблемами, как сложность получения полной и исчерпывающей информации; наличие различных источников происхождения информации; противоречивый характер требований, поступающих от различных специалистов; потеря управляемости требованиями из-за их большого количества.

6.7.3. Работы при выборе и обосновании продуктового решения

Существуют различные подходы к построению информационной системы организации. При выборе подхода решается вопрос о стратегии автоматизации - использовании существующих на рынке типовых тиражируемых программных продуктов или необходимости создания заказного решения, ориентированного только на задачи конкретной организации, рассматриваются возможные варианты реализации выбранного подхода.

Заказные решения обычно используются при уникальности автоматизируемых процессов или отсутствии на рынке программных продуктов требуемой функциональности. Каждая организация имеет свои особенности, не бывает типовых тиражируемых программных продуктов, которые на 100% отвечают всем требованиям. Наиболее полно всю специфику организации и ее уникальные процессы учитывают именно заказные решения. Кроме того, при разработке заказного решения могут быть учтены интеграционные требования, в то время как структура типового программного решения может не позволить решить вопросы интеграции с другими эксплуатируемыми в организации программными продуктами.

Основными недостатками заказной разработки являются следующие положения:

- Работоспособность типового тиражируемого продукта можно проверить до его приобретения (на основе сведений о выполненных проектах по его внедрению на других предприятиях), поэтому его использование менее рискованно, чем заказная разработка.
- Тиражируемое решение внедряется поэтапно и частично может быть доступно в рабочем режиме гораздо быстрее, чем заказная разработка.
- Заказные разработки характеризуются низкой расширяемостью, могут не учитывать возможности расширения бизнес-операций предприятия и при изменениях потребуются существенная модификации программного продукта.
- Временные затраты на разработку и внедрение заказного решения гораздо выше, чем при использовании типового программного решения, т.к. последние складываются из временных затрат на выбор решения и его внедрение.

Выбор и обоснование наиболее подходящего для организации подхода к автоматизации и конкретного программного решения - ключевой момент создания информационной системы предприятия, важная и сложнейшая задача в условиях высокой динамики бизнеса.

Для создания информационной системы организации применяются различные классы типовых тиражируемых программных продуктов:

- системы управления ресурсами предприятия (Enterprise Resource Planning, ERP - планирование ресурсов предприятия / Manufacturing Requirement Planning, *MRP II* - планирование производственных ресурсов/ Material Requirements Planning, *MRP* -- планирование материальных ресурсов);
- системы *управления активами* и фондами (Enterprise Asset Management, *EAM*);
- системы управления взаимоотношениями с клиентами (Customer Relationship Management, *CRM*);
- системы управления цепочками поставок (*Supply Chain Management, SCM*);
- системы управления персоналом (*Human Resources Management, HRM*);
- системы документационного обеспечения управленческой деятельности;
- системы управления эффективностью бизнеса (*Business Performance Management, BPM*);
- системы интеллектуального бизнес-анализа (*Business Intelligence BI*);
- системы управления данными об изделии (*Product Data Management, PDM*).

Назначение основных классов программных продуктов, их функциональные возможности, и особенности рассмотрены в п.6.8. Типовые тиражируемые решения представлены на рынке программных средств как отечественными, так и зарубежными разработчиками. Возможность использования зарубежной или отечественной разработки оценивается на основе анализа достоинств и недостатков в условиях конкретного проекта.

Зарубежные программные продукты ориентированы на хорошо структурированную систему бизнес-процессов организации, как правило, опираются на наборы стандартов, которым процессы должны удовлетворять, но имеют более высокую стоимость по сравнению с российскими решениями.

Российские программные продукты более полно учитывают национальные особенности, российскую учетную специфику.

Выбор типовых тиражируемых программных продуктов, разработчиков уникальных программных систем может проводиться как на конкурсной, так и внеконкурсной основе.

Если организация не планирует проведение полномасштабного конкурса, то для выбора и оценки программных продуктов может использоваться процедура запроса предложений.

В этом случае организацией дается объявление о проведении открытого запроса предложений, либо проводится рассылка информационного письма "Запрос информации" потенциальным поставщикам.

В письме "Запрос информации" кратко дается общая информация о проекте и условиях участия, запрашивается краткая информация о поставщиках, программных продуктах.

Поставщикам, ответившим на указанные письма, либо откликнувшимся на приглашение к участию в процедуре запроса предложений, передается документ "Запрос предложения". Этот документ содержит основные положения о планируемом проекте и все необходимые организации требования к программному продукту, в т.ч. предполагаемую функциональность.

В состав документа "Запрос предложения", как правило, включают следующую информацию:

1. Общие сведения об организации.
2. Цели организации, задачи, стратегический план (необходимые выдержки).
3. ИТ-стратегию или тактический план (необходимые выдержки).
4. Ожидаемые результаты проекта.
5. Требования к программному продукту, включая требования к функциональности.
6. Требования к поставщику решения.
7. Требования по оформлению и документарному составу предложения. Стандартные формы.
8. Критерии и методику оценки предложений.
9. Модель ценообразования.
10. Основные договорные требования / проект договора.
11. Контакты.
12. Сроки и место представления предложений.

Анализ поступивших предложений и выбор наилучшего решения проводится по заранее разработанным критериям оценки предложения и выбранной методике *сравнительной оценки*.

Предварительно разрабатывают следующие группы критериев выбора, используемых при сравнительном анализе:

1. Критерии, позволяющие оценить соответствие тех или иных программных решений заданным требованиям.
2. Квалификационные и другие критерии, предъявляемые к поставщику решения.

Вопросами разработки требований и критериев оценки программных продуктов занимаются как сами организации - клиенты, так и разработчики программных средств, исследовательские компании, продуктовые ИТ- консультанты.

По результатам исследования, проведенного компаниями SAP и Market-Visio Consulting/ Gartner, организации при выборе поставщика ИТ-решений, в первую очередь, обращают внимание на качество предлагаемых продуктов и услуг. Вторым важным фактором является наличие истории успешных внедрений в организациях данной или схожей отрасли. Третий фактор - квалификация сотрудников ИТ- *интеграторов*.

При выборе поставщика комплексных программных решений ведущая аналитическая компания Gartner рекомендует использовать следующие критерии оценки:

1. Функциональные возможности.
2. Архитектура: техническая инфраструктура, необходимая для поддержки решения.
3. Устойчивость продукта: оценка в трех измерениях - финансы, структура и рынок.
4. Цена: средняя стоимость приобретения, инсталляции, обновления версий и технической поддержки программного продукта.
5. Сервис и поддержка: уровень технической поддержки, который обеспечивают поставщик и его партнеры.
6. Концепция и видение: прогнозы поставщика относительно тенденций развития отрасли; действия поставщика в плане функциональности и стратегии развития продукта в свете этих прогнозов.

Поставщик оценивается в двух аспектах - стабильность и полнота концепции.

В настоящее время компанией Gartner Inc. запатентована методика "Magic Quadrant" (Магический Квадрант), которая заключается в подготовке отчета с графическим представлением определенного рынка продукции за некоторый период времени. Этот отчет сравнивает компании по набору критериев, разработанных для данного рынка, и отражает мнение исследовательской компании. В "Магическом квадранте" компании оцениваются по полноте видения рынка (completeness of vision) и по способности к практической реализации этого видения (ability to execute). В соответствии с этими критериями системы различных производителей располагаются в четырех квадрантах: "лидеры" (*leaders*), "провидцы" (*visionaries*), "бросающие вызов" (*challengers*) и "нишевые игроки" (*Niche Players*). К сектору лидеров компания Gartner Inc. относит поставщиков, которые успешно ведут свою деятельность, имеют четкую концепцию работы на рынке и активно совершенствуют свои возможности для реализации этой концепции и удержания своих лидирующих позиций.

Исследования компании Gartner Inc. предоставляют один из источников информации для проведения сравнительного анализа и не являются окончательной рекомендацией выбирать только того производителя, который оценен как "Лидер".

Обобщая различные подходы, можно выделить следующие типовые критерии, применяемые при сравнительной оценке программных продуктов:

- функциональная полнота и возможность поддержки информационной модели организации;
- отраслевая специфика;
- наличие инструментов разработки, позволяющих дополнить отсутствующие функции;
- масштабируемость;
- гибкость;
- стандартизация и открытость;
- сложность сопровождения и администрирования;
- архитектура и техническая платформа;
- стоимость;
- перспективы развития;
- информационная безопасность;
- профессиональные знания и квалификация поставщика;
- опыт и репутация поставщика;
- надежность поставщика.

Следует отметить, что разработка состава критериев оценки программных продуктов по функциональности зависит от класса, к которому принадлежат рассматриваемые программные решения. Например, один из подходов к сопоставлению ERP-систем по функциональности разработан аналитической компанией Arlington Software Corporation в рамках проекта ERP Evaluation Center.

ERP Evaluation Center является ресурсом компании TEC Group, целью которого является анализ и сравнение, представленных на рынке ERP-систем. Согласно разработанному подходу для оценки функциональности используется дерево критериев, со-

держат более 3600 частных критериев. Критерии нижнего уровня входят в критерии более высокого уровня со своими весовыми коэффициентами. Вершина дерева представляет собой комплексную численную оценку функциональности системы. В рамках проекта разработана таблица весов критериев и программные средства для решения многокритериальных задач. Помимо критериев функциональности для ERP-систем разработаны иерархии частных критериев для отдельных систем: CRM (более 1100 критериев), PLM (Product Lifecycle Management - управление жизненным циклом продукции, более 1300 критериев), SCM (более 2200 критериев), BI (более 1300 критериев).

Следует отметить, что только функциональное сравнение программных решений не может обеспечить полного видения картины при принятии решения о выборе программного продукта.

Итогом результатом проведенного сравнительного анализа программных продуктов по всему комплексу выбранных критериев является заключение о выборе наилучшего решения, а также отчет, позволяющий проанализировать обоснование сделанных рекомендаций. На основе представленных результатов руководство организации или специальная комиссия принимает окончательное решение о приобретении и внедрении программного продукта.

Для крупных проектов выбор программных продуктов из альтернативных вариантов целесообразно проводить на конкурсной основе.

Стандартное аппаратное и программное обеспечение, информационные системы имеют свои особенности как предмет конкурса.

Выделяют следующие особенности стандартного аппаратного и программного обеспечения как предмета конкурса:

- спецификации стандартного программного и аппаратного обеспечения должны отражать текущее состояние рынка данной продукции в условиях высокой динамики его развития, в то время как подготовка и проведение конкурса может занимать достаточно длительный период;
 - диапазон оборудования, которое соответствует понятию стандартного аппаратного обеспечения, очень широк;
 - подготовка спецификаций на оборудование отличается высокой трудоемкостью, поскольку может включать подготовку чертежей, например, в случае закупки сопутствующих услуг по монтажу вычислительных систем;
 - необходимо учитывать особенности лицензионной политики разработчиков стандартного программного обеспечения (корпоративные лицензии, скидки для отдельных категорий пользователей);
 - в конкурсную документацию должны быть внесены требования:
 - обеспечивающие возможность модернизации аппаратного и программного обеспечения;
 - по обеспечению расходными материалами;
 - необходимо учитывать общую стоимость владения;
 - требования должны быть сформулированы в соответствии с положениями нормативных документов.
- Особенности информационной системы как предмета конкурса определяются следующими положениями:
- необходимостью подготовки специальных требований:
 - к аппаратному и программному обеспечению;
 - по проведению тестирования, приемо-сдаточных испытаний, пуско-наладочных работ;
 - по интеграции с уже имеющимися информационными системами;
 - необходимостью определения организационных мероприятий, связанных с проведением пусконаладочных и других работ;
 - необходимостью учета в условиях контракта вопросов, связанных с причинением ущерба в ходе выполнения различных работ.

Эти особенности обуславливают необходимость и целесообразность привлечения продуктовых ИТ-консультантов для участия в работах при подготовке и проведении конкурса.

Проведению открытого конкурса по закупкам стандартного программного и аппаратного обеспечения, информационных систем должно предшествовать проведение предварительное обследование организации, на основе документированных результатов которого разрабатываются требования к информационной системе, программному и аппаратному обеспечению.

Формально предварительное обследование не относится к процедуре открытого конкурса, однако его качественное проведение оказывает существенное влияние на проведение конкурса.

В работах по проведению предварительного обследования предприятия обычно принимают участие продуктовые ИТ-консультанты. На основе полученных результатов принимается решение о проведении определенного вида конкурса (одноэтапного, двухэтапного, с предварительным отбором, без предварительного отбора), а выявленные требования к функциональности программного продукта и информационной системе в целом включаются в конкурсную документацию и техническое задание. В состав конкурсной документации включают также требования к правомочности и квалификации поставщика, критерии и методы оценки программных продуктов и информационных систем.

Практическое занятие:

по теме: «Разработка перечня обучающей документации на информационную систему»

Цель: научиться разрабатывать перечень обучающей документации на информационную систему

По завершению практического занятия студент должен уметь: разрабатывать перечень обучающей документации на информационную систему

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Задание

Основу отечественной нормативной базы в области документирования ПС составляет комплекс стандартов *Единой системы программной документации (ЕСПД)*.

Основная и большая часть комплекса ЕСПД была разработана в 70-е и 80-е годы 20 века. Сейчас этот комплекс представляет собой систему межгосударственных стандартов стран СНГ (ГОСТ), действующих на территории Российской Федерации на основе межгосударственного соглашения по стандартизации.

Единая система программной документации - это комплекс государственных стандартов, устанавливающих взаимосвязанные правила разработки, оформления и обращения программ и программной документации.

Стандарты ЕСПД в основном охватывают ту часть документации, которая создается в процессе разработки программных средств, и связаны, по большей части, с документированием функциональных характеристик программных средств. Следует отметить, что стандарты ЕСПД (ГОСТ 19) носят рекомендательный характер. Впрочем, это относится и ко всем другим стандартам в области ИС (ГОСТ 34, международному стандарту ISO/IEC и др.). Дело в том, что в соответствии с Законом РФ «О стандартизации» эти стандарты становятся обязательными на контрактной основе, т.е. при ссылке на них в договоре на разработку (поставку) программного средства.

Говоря о состоянии ЕСПД в целом, можно констатировать, что большая часть стандартов ЕСПД морально устарела. Тем не менее до пересмотра всего комплекса многие стандарты могут с пользой применяться в практике документирования программных средств.

К числу **программных** ЕСПД относят документы, содержащие сведения, необходимые для разработки, изготовления, сопровождения и эксплуатации программ.

Как известно, грамотно составленный пакет программной документации позволяет избежать при проектировании многих неприятностей. В частности, избавиться от назойливых вопросов и необоснованных претензий заказчика можно, просто отослав пользователя к документации. Это касается прежде всего важнейшего документа — **Технического задания**.

Техническое задание (ТЗ) содержит совокупность требований к программному средству и может использоваться как критерий проверки и приемки разработанной программы. Поэтому достаточно полно составленное (с учетом возможности внесения дополнительных разделов) и принятое заказчиком и разработчиком ТЗ является одним из основополагающих документов проекта программного средства.

ГОСТ 19.201-78, входящий в ЕСПД, устанавливает порядок построения и оформления технического задания на разработку программы или программного изделия для вычислительных машин, комплексов и систем независимо от их назначения и области применения.

2. Задание: разработать техническое задание на проектирование информационной системы, предназначенной для решения задач автоматизации деятельности организации.

Исходными данными для проектирования информационной системы являются описание предметной области и виды запросов в информационной системе (приложение 1).

Алгоритм выполнения работы

1) В соответствии с назначенным преподавателем вариантом определить наименование информационной системы (табл. 1), подлежащей проектированию в ходе лабораторного практикума, для удовлетворения основных требований к ней с применением системы управления базами данных

№ варианта	Наименование информационной системы
1	Информационная система медицинских организаций города
2	Информационная система автопредприятия города
3	Информационная система проектной организации
4	Информационная система ГИБДД
5	Информационная система строительной организации
6	Информационная система библиотечного фонда города
7	Информационная система спортивных организаций города
8	Информационная система аэропорта
9	Информационная система гостиничного комплекса

Microsoft Access 2007 и/или инструментального средства Borland Turbo Delphi.

2) Изучить описание предметной области информационной системы (приложение 1).

3. На основании анализа описания предметной области и запросов к будущей информационной системе (приложение 1)

№ варианта	Наименование информационной системы
10	Информационная система торговой организации
11	Информационная система ВУЗа
12	Информационная система железнодорожной пассажирской станции
13	Информационная система зоопарка
14	Информационная система театра
15	Информационная система фотоцентра

сформулировать основные требования к ее функциям.

4. Выполнить поиск прототипа проектируемой информационной системы с применением Интернет.

5. Используя сформулированные требования к информационной системе, а также документацию пользователя на прототип найденного программного средства, разработать техническое задание в соответствии с ГОСТ 19.201-78

Практическое занятие:

по теме: «Разработка руководства оператора»

Цель: научиться разрабатывать руководство для оператора

По завершению практического занятия студент должен уметь: разрабатывать руководство для оператора

Продолжительность: 3 аудиторных часа (135 минут)

Необходимые принадлежности

Задание

Оператор ЭВМ должен знать:

- приказы, указания, распоряжения, инструкции и другие нормативно-распорядительные документы, регламентирующие работу оператора ЭВМ;
- правила эксплуатации ЭВМ и обслуживания ксерокса;
- правила оформления документов, в том числе деловой документации с использованием типовых форм;
- правила орфографии и пунктуации;
- правила ведения делопроизводства;
- программное обеспечение (правила работы с Windows, Microsoft Office и т.д.)
- средства вычислительной техники, коммуникаций и связи;
- культуру труда и этику делового общения;
- основы законодательства о труде и охране труда Российской Федерации;
- устав предприятия, его штатное расписание, правила внутреннего трудового распорядка;
- правила и нормы охраны труда, техники безопасности, производственной санитарии и противопожарной защиты.

Оператор ЭВМ персонала подчиняется

- 3.1. Знакомиться с проектами решений руководства компании, касающимися его деятельности.
- 3.2. Вносить на рассмотрение руководства предложения по совершенствованию работы, связанной с обязанностями, предусмотренными настоящей должностной инструкцией.
- 3.3. Осуществлять взаимодействие с сотрудниками всех структурных подразделений.
- 3.4. Запрашивать лично или по поручению непосредственного руководителя от сотрудников других подразделений компании информацию и документы, необходимые для выполнения своих должностных обязанностей.
- 3.5. Требовать от руководства предприятия оказания содействия в исполнении своих должностных обязанностей и прав.

«Единый тарифно-квалификационный справочник работ и профессий рабочих», назначение и содержание ЕТКС
Единый тарифно-квалификационный справочник работ и профессий рабочих состоит из тарифно-квалификационных характеристик, содержащих характеристики основных видов работ по профессиям рабочих в зависимости от их сложности, и соответствующих им тарифных разрядов, а также требования, предъявляемые к профессиональным знаниям и навыкам рабочих.

Характеристика профессии «Оператор электронно-вычислительных машин»: согласно ЕТКС.

Ведение процесса обработки информации на электронно - вычислительных машинах по рабочим инструкциям с пульта управления. Ввод информации в электронно - вычислительные машины (ЭВМ) с технических носителей информации и каналов связи и вывод ее из машины. Передача по каналам связи полученных на машинах расчетных данных на последующие операции. Подготовка технических носителей информации на устройствах подготовки данных.

Запись, считывание и перезапись информации с одного вида носителей на другой. Контроль технических носителей информации. Обеспечение проведения вычислительного процесса в соответствии с рабочими программами. Наблюдение за работой ЭВМ. Установление причин сбоев в работе в процессе обработки информации. Запись об использовании машинного времени и замеченных дефектах работы машин в журнал по учету машинного времени. Оформление результатов выполненных работ.

Должен знать: устройство пульта управления ЭВМ и правила ее технической эксплуатации; основы программирования в объеме среднего специального или общего образования и курсовой подготовки; рабочие инструкции и другие руководящие материалы по обработке информации; технические носители информации, коды, применяемые на ЭВМ, структуру выходных таблиц для обнаружения сбоев во время работы ЭВМ.

Обязанности оператора ЭВМ, согласно «Общероссийскому классификатору занятий», приведите примеры профессий, входящих в данную базовую группу.

Техники и операторы по обслуживанию ЭВМ осуществляют стандартное программное обеспечение ЭВМ, техническое обслуживание и контроль за работой вычислительной техники, средств приема и передачи информации, участвуют в совершенствовании систем обработки данных.

Их обязанности включают:

- эксплуатацию и контроль за работой электронно-вычислительной техники;
- организацию и эффективное выполнение вычислительных операций;
- подготовку средств вычислительной техники к работе, составление схем обработки информации, программ и алгоритмов решения задач;
- проведение технического обслуживания, тестовых проверок, профилактических осмотров, регулировки, наладки и текущего ремонта оборудования;
- ведение учета объемов выполненных работ, использования машинного времени, замеченных дефектов работы машин;
- выполнение родственных по содержанию обязанностей;
- руководство другими работниками.

Примеры профессий, входящих в данную базовую группу:

Техник (по обслуживанию ЭВМ) Оператор ЭВМ

Профессия Оператор вычислительных и электронно-вычислительных машин, характеристика работ и обязанности, согласно постановлению Минтруда РФ «Об утверждении тарифно-квалификационных характеристик по общетраслевым профессиям рабочих».

Ведение процесса обработки информации на электронно-вычислительных машинах по рабочим инструкциям с пульта управления. Ввод информации в электронно-вычислительные машины (ЭВМ) с технических носителей информации и каналов связи и вывод ее из машин. Передача по каналам связи полученных на машинах расчетных данных на после-

дующие операции. Обработка первичных документов на вычислительных машинах различного типа путем суммирования показателей сводок с подгибкой и подкладкой таблиц, вычислений по инженерно-конструкторским расчетам. Выписка счетов-фактур и составление ведомостей, таблиц, сводок, отчетов механизированным способом, с выводом информации на перфоленту. Контроль вычислений, выверка расхождений по первичному документу. Подготовка машины к работе, установка шины управления или блок-схемы на данную работу. Введение перфорации, верификации, дублирования, репродукции и табуляции перфокарт. Считывание и пробивка отверстий закодированной информации, содержащейся в перфокартах, на основании графических отметок.

Проверка правильности переноса информации с первичных документов на перфокарты "на свет" и счетным контролем и правильности перебивки неверно отперфорированных перфокарт с исправлением соответствующих показателей и итогов в табуляграмме.

Контроль табуляграмм, составленных механизированным способом, сличением их итоговых данных с контрольными числами; проведение выборочной балансировки с отметкой на полях табуляграмм; запись выверенных итогов табуляграмм в журнал контрольных чисел; оформление и выпуск проверенных табуляграмм. Настройка машины по простым схемам коммутации и самостоятельное осуществление несложной перекоммутации. Установка пропускной линейки, упоров и табуляционных пластин для осуществления многократных пропусков перфокарт. Работа с математическими справочниками, таблицами. Оформление сопроводительного документа и рабочего наряда на выполненные работы.

Должен знать:

- технико-эксплуатационные характеристики вычислительных машин;
- устройство пульта управления и правила технической эксплуатации ЭВМ;
- руководящие материалы, определяющие последовательность и содержание выполняемых операций технологического процесса;
- действующие шифры и коды;
- методы проведения расчетов и вычислительных работ, контроля технических носителей информации;
- основы коммутации и простые блок-схемы настройки машин;
- формы исходных и выпускаемых документов;
- основы программирования в объеме среднего специального или общего образования и курсовой подготовки.

Влияние ПК на здоровье человека;

- Головная боль и боль в глазах;
- Утомление, головокружение;
- Нарушение ночного сна;
- Сонливость в течении дня;
- Изменение настроения;
- Повышенная раздражительность;
- Депрессия;
- Снижение интеллектуальных способностей, ухудшение памяти;
- Натяжение кожи лба и головы;
- Выпадение волос;
- Боль в мышцах;
- Боль в области сердца, неровное сердцебиение, одышка;
- Снижение половой активности.

Инструкция по технике безопасности, основные положения, назначение;

1.1. Настоящая инструкция распространяется на следующих работников гимназии:

- программистов, занятых на ПК разработкой, проверкой и отладкой программ;
- пользователей ПК (работников, совмещающих работу оператора ПК с основной работой).

Перечисленные выше работники называются далее операторами.

1.2. Во время работы с ПК на оператора возможно воздействие следующих опасных и вредных факторов:

а) физических:

- низкочастотные электрические и магнитные поля;
- статическое электричество;
- лазерное и ультрафиолетовое излучение;
- повышенная температура;
- ионизация воздуха;
- опасное напряжение в электрической сети;

б) химических:

- пыль;
- вредные химические вещества, выделяемые при работе принтеров и копировальной техники;

в) психофизиологических:

- напряжение зрения и внимания;
- интеллектуальные и эмоциональные нагрузки;
- длительные статические нагрузки и монотонность труда.

1.3. К работам с ПК допускаются лица:

- не моложе 18 лет, прошедшие обязательный предварительные при приеме на работу и ежегодные медицинские осмотры в порядке и сроки, установленные Минздравмедпромом России и Госкомсанэпиднадзором России, и не имеющие медицинских противопоказаний для работы с ПК и ВДТ;
 - прошедшие курс обучения принципам работы с вычислительной техникой и специальное обучение работе на ПЭВМ с использованием конкретного программного обеспечения;
 - прошедшие вводный инструктаж по электробезопасности с присвоением 1-й квалификационной группы;
 - ознакомленные с инструкциями по эксплуатации на используемые на рабочем месте средства оргтехники (собственно ПК, принтеры, сканеры, источники бесперебойного питания и т.п.).
- 1.4. К работе с ПК не допускаются женщины со времени установления беременности и в период кормления ребенка грудью.

Область распространения и порядок применения инструкции по технике безопасности;

Настоящая инструкция распространяется на следующих работников гимназии:

- программистов, занятых на ПК разработкой, проверкой и отладкой программ;

- пользователей ПК (работников, совмещающих работу оператора ПК с основной работой).

Перечисленные выше работники называются далее операторами.

Изучите инструкцию, находящуюся в Вашем компьютерном классе. Какие пункты можно было бы добавить в инструкцию, какие пункты Вы считаете лишними;

Я считаю, что в нашей инструкции нет лишних пунктов и добавлять не нужно.

Требования к персоналу, эксплуатирующему средства вычислительной техники и периферийное оборудование;
К самостоятельной эксплуатации электроаппаратуры допускается только специально обученный персонал не моложе 18 лет, пригодный по состоянию здоровья и квалификации к выполнению указанных работ.

Требование по электрической безопасности;

1. Необходимо постоянно следить на своем рабочем месте за исправным состоянием электропроводки, выключателей, штепсельных розеток, при помощи которых оборудование включается в сеть. При обнаружении неисправности немедленно обесточить электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности.

2. Во избежание повреждения изоляции проводов и возникновения коротких замыканий не разрешается:

а) вешать что-либо на провода;

б) окрашивать и белить шнуры и провода;

в) закладывать провода и шнуры за батареи отопительной системы;

г) выдергивать штепсельную вилку из розетки за шнур, усилие должно быть приложено к корпусу вилки.

3. Для исключения поражения электрическим током запрещается:

а) часто включать и выключать компьютер без необходимости;

б) прикасаться к экрану и к тыльной стороне блоков компьютера;

в) работать с оборудованием мокрыми руками;

г) работать с оборудованием при нарушении целостности корпуса, изоляции проводов, неисправную индикацию включения питания, с признаками электрического напряжения на корпусе.

д) класть на оборудование посторонние предметы.

4. Запрещается под напряжением очищать от пыли и загрязнения электрооборудование.

5. Во избежание поражения электрическим током, при пользовании электроприборами нельзя касаться одновременно каких-либо трубопроводов, батарей отопления, металлических конструкций, соединенных с землей.

6. При обнаружении оборвавшегося провода необходимо немедленно сообщить об этом администрации, принять меры по исключению контакта с ним людей. Прикосновение к проводу опасно для жизни.

7. Спасение пострадавшего при поражении электрическим током главным образом зависит от быстроты освобождения его от действия тока.

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации, одновременно оповестить о пожаре администрацию.

Особенности электропитания системного блока;

Все компоненты системного блока получают электроэнергию от блока питания. Блок питания ПК — это автономный узел, находящийся в верхней части системного блока. Правила техники безопасности не запрещают вскрывать системный блок, например при установке дополнительных внутренних устройств или их модернизации, но это не относится к блоку питания. Блок питания компьютера — источник повышенной пожаро-опасности, поэтому вскрытию и ремонту он подлежит только в специализированных мастерских.

Блок питания имеет встроенный вентилятор и вентиляционные отверстия. В связи с этим в нем неминуемо накапливается пыль, которая может вызвать короткое замыкание. Рекомендуется периодически (один - два раза в год) с помощью пылесоса удалять пыль из блока питания через вентиляционные отверстия без вскрытия системного блока. Особенно важно производить эту операцию перед каждой транспортировкой или наклоном системного блока.

Требования к видеосистеме;

В прошлом монитор рассматривали в основном как источник вредных излучений, воздействующих прежде всего на глаза. Сегодня такой подход считается недостаточным. Кроме вредных электромагнитных излучений (которые на современных мониторах понижены до сравнительно безопасного уровня) должны учитываться параметры качества изображения, а они определяются не только монитором, но и видеоадаптером, то есть всей видеосистемой в целом.

Требования к рабочему месту;

В требования к рабочему месту входят требования к рабочему столу, посадочному месту (стулу, креслу), Подставкам для рук и ног. Несмотря на кажущуюся простоту, обеспечить правильное размещение элементов компьютерной системы и правильную посадку пользователя чрезвычайно трудно. Полное решение проблемы требует дополнительных затрат, сопоставимых по величине со стоимостью отдельных узлов компьютерной системы, поэтому и битву и на производстве этими требованиями часто пренебрегают.

Несмотря на то, что школьники проводят в компьютерном классе сравнительно немного времени, обучить их правильной гигиене труда на достойном примере очень важно, чтобы полезные навыки закрепились на всю жизнь.

Это не просто требование гигиены, а требование методики:

1. Монитор должен быть установлен прямо перед пользователем и не требовать поворота головы или корпуса тела.

2. Рабочий стол и посадочное место должны иметь такую высоту, чтобы уровень глаз пользователя находился чуть выше центра монитора. На экран монитора следует смотреть сверху вниз, а не наоборот. Даже кратковременная работа с монитором, установленным слишком высоко, приводит к утомлению шейных отделов позвоночника.

3. Если при правильной установке монитора относительно уровня глаз выясняется, что ноги пользователя не могут свободно покоиться на полу, следует установить подставку для ног, желательно наклонную. Если ноги не имеют надежной опоры, это непременно ведет к нарушению осанки и утомлению позвоночника. Удобно, когда компьютерная мебель (стол и рабочее кресло) имеют средства для регулировки по высоте. В этом случае проще добиться оптимального положения.

4. Клавиатура должна быть расположена на такой высоте, чтобы пальцы рук располагались на ней свободно, без напряжения, а угол между плечом и предплечьем составлял 100° — 110°. При использовании обычных школьно-письменных столов добиться одновременно правильного " положения и монитора, и клавиатуры практически невозможно. Для работы рекомендуется использовать специальные компьютерные столы, имеющие выдвижные полочки для клавиатуры. Если такой полочки нет и клавиатура располагается на том же столе, что и монитор, использование подставки для ног становится практически неизбежным, особенно когда с компьютером работают дети.

5. При длительной работе с клавиатурой возможно утомление сухожилий кистевого сустава. Известно тяжелое профессиональное заболевание — кистевой туннельный синдром, связанное с неправильным положением рук на клавиатуре.

Во избежание чрезмерных нагрузок на кисть желательнее предоставить рабочее кресло с подлокотниками, уровень высоты которых, замеренный от пола, совпадает с уровнем высоты расположения клавиатуры.

6. При работе с мышью рука не должна находиться на весу. Локоть руки или хотя бы запястье должны иметь твердую опору. Если предусмотреть необходимое расположение рабочего стола и кресла затруднительно, рекомендуется применить коврик для мыши, имеющий специальный опорный валик. Нередки случаи, когда в поисках опоры для руки (обычно правой) располагают монитор сбоку от пользователя (соответственно, слева), чтобы он работал вполборота, опирая локоть или запястье правой руки о стол. Этот прием недопустим. Монитор должен обязательно находиться прямо перед пользователем.

Требование по обеспечению пожарной безопасности;

На рабочем месте запрещается иметь огнеопасные вещества.

В помещениях запрещается:

- а) зажигать огонь;
- б) включать электрооборудование, если в помещении пахнет газом;
- в) курить;
- г) сушить что-либо на отопительных приборах;
- д) закрывать вентиляционные отверстия в электроаппаратуре

Источниками воспламенения являются:

- а) искра при разряде статического электричества
- б) искры от электрооборудования
- в) искры от удара и трения
- г) открытое пламя

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации, одновременно оповестить о пожаре администрацию.

Помещения с электрооборудованием должны быть оснащены огнетушителями типа ОУ-2 или ОУБ-3.

Требования охраны труда при работе на ПК;

- Максимальное время работы за компьютером не должно превышать 6 часов за смену.
- Необходимо делать перерыв в работе за компьютером продолжительностью 10 минут через каждые 45 мин.
- Продолжительность непрерывной работы за компьютером без регламентированного перерыва не должно превышать 1 час.

Практическое занятие:

по теме: «Разработка моделей интерфейсов пользователей»

Цель: научиться разрабатывать модели интерфейсов пользователей

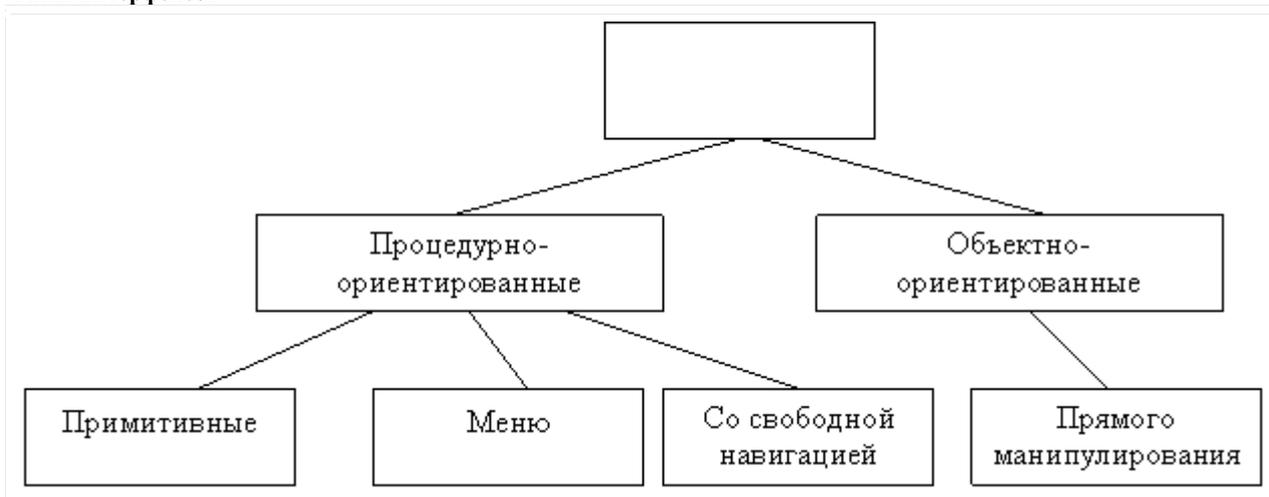
По завершению практического занятия студент должен уметь: разрабатывать модели интерфейсов пользователей

Продолжительность: 3 аудиторных часа (135 минут)

Необходимые принадлежности

Задание

Типы интерфейсов



Интерфейсы Пользователя Рисунок 1 – Типы интерфейсов

Процедурно-ориентированные интерфейсы используют модель взаимодействия с пользователем, основанную на понятиях «процедура» и «операция». Программное обеспечение предоставляет пользователю возможность выполнения некоторых *действий*, для которых пользователь определяет соответствующие данные и следствием выполнения которых является получение желаемых результатов.

Объектно-ориентированные интерфейсы используют модель взаимодействия с пользователем, ориентированную на манипулирование объектами предметной области. Пользователю предоставляется возможность напрямую взаимодействовать с каждым объектом и инициировать выполнение операций, в процессе которых взаимодействуют несколько объектов. Задача пользователя формулируется как целенаправленное изменение некоторого объекта, имеющего внутреннюю структуру, определенное содержание и внешнее символическое или графическое представление. Например, модель реальной системы или процесса, база данных, текст и т. д. Элементы интерфейсов данного типа включены в пользовательский интерфейс Windows. Например, пользователь может «взять» файл и «переместить» его в другую папку. Таким образом, он инициирует выполнение операции перемещения файла.

Примитивным называют интерфейс, который организует взаимодействие с пользователем в консольном режиме. Такой интерфейс реализует конкретный сценарий работы программного обеспечения, например: ввод данных – решение задачи – вывод результата. Обычно используются при обучении программированию или когда программа реализует одну функцию.

Интерфейс-меню позволяет выбирать необходимые операции из специального списка, выводимого ему программой. Эти интерфейсы предполагают реализацию множества сценариев работы, последовательность действий в которых определяется пользователем. Различают одноуровневые и иерархические меню.

Интерфейсы со свободной навигацией называют *графическими пользовательскими интерфейсами*. Интерфейсы этого типа ориентированы на использование экрана в графическом режиме с высокой разрешающей способностью.

Графические интерфейсы поддерживают концепцию интерактивного взаимодействия с программным обеспечением, осуществляя визуальную обратную связь с пользователем и возможность прямого манипулирования объектами и информацией на экране.

В отличие от интерфейса-меню интерфейс со свободной навигацией обеспечивает возможность осуществления любых допустимых в конкретном состоянии операций, доступ к которым возможен через различные интерфейсные компоненты. Например, окна программ, реализующих интерфейс Windows, обычно содержат:

- меню различных типов: ниспадающее, кнопочное, контекстное;
- разного рода компоненты ввода данных.

Причем выбор следующей операции в меню осуществляется как мышью, так и с помощью клавиатуры.

Особенность интерфейса со свободной навигацией - способность изменяться в процессе взаимодействия с пользователем, предлагая выбор только тех операций, которые имеют смысл в конкретной ситуации. Реализуют интерфейсы со свободной навигацией, используя событийное программирование и объектно-ориентированные библиотеки, что предполагает применение визуальных сред разработки программного обеспечения.

Интерфейс прямого манипулирования предполагает, что взаимодействие пользователя с программным обеспечением осуществляется посредством выбора и перемещения *пиктограмм*, соответствующих объектам предметной области. Для реализации таких интерфейсов также используют событийное программирование и объектно-ориентированные библиотеки.

Этапы разработки пользовательского интерфейса. Разработка пользовательского интерфейса включает те же основные этапы, что и разработка программного обеспечения:

- постановка задачи – определение типа интерфейса и общих требований к нему;
- анализ требований и определение спецификаций – определение сценариев использования и пользовательской модели интерфейса;
- проектирование – проектирование диалогов и их реализация в виде процессов ввода-вывода;
- реализация – программирование и тестирование интерфейсных процессов.

2. Психофизические особенности человека, связанные с восприятием, запоминанием и обработкой

При проектировании пользовательских интерфейсов необходимо учитывать психофизические особенности человека, связанные с восприятием, запоминанием и обработкой информации.

Исследованием принципов работы мозга человека занимается *когнитивная психология*.

Информация о внешнем мире поступает в наш мозг в огромных количествах. Часть мозга, которую условно можно назвать «процессором восприятия», постоянно без участия сознания перерабатывает ее, сравнивая с прошлым опытом, и помещает в хранилище уже в виде зрительных, звуковых и прочих образов. Любые внезапные или просто значимые для нас изменения в окружении привлекают наше внимание, и тогда интересующая нас информация поступает в кратковременную память. Если же наше внимание не было привлечено, то информация в хранилище пропадает, замещаясь следующими порциями.

В каждый момент времени фокус внимания может фиксироваться в одной точке. Поэтому, если возникает необходимость «одновременно» отслеживать несколько ситуаций, то обычно фокус перемещается с одного отслеживаемого элемента на другой. При этом внимание «рассредоточивается», и какие-то детали могут быть упущены.

Обработка процессором восприятия требует некоторого времени и, если сигнал выдается в течение времени, меньшем времени обработки, то наш мозг его не воспринимает.

Восприятие во многом основано на мотивации. Например, если человек голоден, то он в первую очередь будет замечать все съедобное, а если устал – то, войдя в комнату, он в первую очередь увидит диван или кровать.

В процессе переработки информации мозг сравнивает поступающие данные с предыдущими.

При смене кадра мозг на некоторое время блокируется: он «осваивает» новую картинку, выделяя наиболее существенные детали. А значит, если необходима быстрая реакция пользователя, то резко менять картину не стоит.

Краткосрочная память – самое «узкое» место «системы обработки информации» человека. Ее емкость приблизительно равна 7±2 несвязанных объектов. Не востребованная информация хранится в краткосрочной памяти не более 30 с.

При проектировании интерфейсов следует иметь в виду, что подавляющему большинству людей сложно, например, запомнить и ввести на другом экране число, содержащее более 5 цифр (7-2), или некоторое сочетание букв.

Долговременная память человека – хранилище информации с неограниченной емкостью и временем хранения. Однако доступ к этой информации непрост: по всей вероятности, механизмы извлечения информации из памяти имеют ассоциативный характер. Специальная методика запоминания информации (*мнемоника*) использует именно это свойство памяти: для запоминания информации ее «привязывают» к тем данным, которые память уже хранит и позволяет легко получить.

Поскольку доступ к долговременной памяти затруднен, целесообразно рассчитывать не на то, что пользователь вспомнит нужную информацию, а на то, что он ее узнает. Поэтому интерфейс типа меню так широко используется.

Особенности восприятия цвета. Цвет в сознании человека ассоциируется с эмоциональным фоном. Теплые цвета: красный, оранжевый, желтый человека возбуждают, а холодные: синий, фиолетовый, серый – успокаивают. Причем цвет является очень сильным раздражителем, поэтому применять цвета в интерфейсе необходимо крайне осторожно.

Обилие оттенков привлекает внимание, но быстро утомляет. Поэтому не стоит ярко раскрашивать окна, с которыми пользователь будет долго работать. Необходимо учитывать и индивидуальные особенности восприятия цветов человеком, например, каждый десятый человек плохо различает какие-то цвета, поэтому в ответственных случаях необходимо предоставить пользователю возможность настройки цветов.

Особенности восприятия звука. В интерфейсах звук обычно используют с разными целями: для привлечения внимания, как фон, обеспечивающий некоторое состояние пользователя, как источник дополнительной информации. Следует учитывать, что большинство людей очень чувствительны к звуковым сигналам, особенно, если последние указывают на наличие ошибки. Поэтому при создании звукового сопровождения целесообразно предусматривать возможность его отключения.

Субъективное восприятие времени. Человеку свойственно субъективное восприятие времени. Занятый человек обычно времени не замечает. Зато в состоянии ожидания время тянется бесконечно, что связано с тем, что в это время мозг оказывается в состоянии информационного вакуума. К аналогичному состоянию приводит усталость: информация поступает, но больше обрабатывается, а потому и ход времени замедляется.

При ожидании более 1-2 с пользователь может отвлечься, «потерять мысль», что неблагоприятно сказывается на результатах работы и увеличивает усталость, так как каждый раз после ожидания много сил тратится на включение в работу. Сократить время ожидания можно, заняв пользователя, но не отвлекая его от работы. Например, можно предоставить ему какую-либо информацию для обдумывания. По возможности целесообразно выводить пользователю промежуточные результаты: во-первых, он будет занят их обдумыванием, во-вторых, по ним он сможет оценить будущие результаты, и отменит операцию, если они его не удовлетворяют.

Для «развлечения» пользователя используется анимация. Например, в Windows при копировании файлов демонстрируется «ролик» с летающими листочками. Следует иметь в виду, что, когда какую-либо анимацию смотришь первый раз, то это интересно, а когда в течение получаса наблюдаешь, как «летают» листочки при получении информации из Интернета, то это начинает раздражать.

Чтобы уменьшить раздражение, возникающее при ожидании, необходимо соблюдать основное правило: информировать пользователя, что заказанные им операции потребуют некоторого времени выполнения. Для этого используют индикаторы оставшегося времени, анимированные объекты, как в Интернете, и изменение формы курсора мыши на песочные часы. Очень важно точно обозначить момент, когда система готова продолжать работу. Обычно для этого используют значительные изменения внешнего вида экрана.

Взаимодействие пользователя с интерфейсом будет определяться не только физическими возможностями и особенностями человека, но и пользовательской моделью интерфейса.

3. Пользовательская и программная модели интерфейса

Модели пользовательского интерфейса:

- модель программиста;
- модель пользователя;
- программная модель.

Программист, разрабатывая пользовательский интерфейс, исходит из того, управление какими операциями ему необходимо реализовать, как это осуществить, не затрачивая ни существенных ресурсов компьютера, ни своих сил и времени. Его интересуют функциональность, эффективность, технологичность, внутренняя стройность и другие, не связанные с удобством пользователя характеристики программного обеспечения. Поэтому большинство интерфейсов существующих программ вызывают серьезные нарекания пользователей.

Пользовательская модель интерфейса – это совокупность обобщенных представлений конкретного пользователя или некоторой группы пользователей о процессах, происходящих во время работы программы или программной системы. Эта модель базируется на особенностях опыта конкретных пользователей, который характеризуется:

- уровнем подготовки в предметной области разрабатываемого программного обеспечения;
- интуитивными моделями выполнения операций в этой предметной области;
- уровнем подготовки в области владения компьютером;
- устоявшимися стереотипами работы с компьютером.

Для построения пользовательской модели необходимо изучить перечисленные выше особенности опыта предполагаемых пользователей программного обеспечения. С этой целью используют опросы, тесты и даже фиксируют последовательность действий, осуществляемых в процессе выполнения некоторых операций, на пленку.

Приведение в соответствие моделей пользователя и программиста, а также построение на их базе программной модели интерфейса задача непростая.



Рисунок 2 – Процесс проектирования пользовательского интерфейса. Интуитивные модели выполнения операций в предметной области должны стать основой для разработки интерфейса, а потому в большинстве случаев их необходимо не менять, а уточнять и

совершенствовать.

Критерии оценки интерфейса пользователем. Основные критерии интерфейсов пользователя:

- простота освоения и запоминания операций системы – конкретно оценивают время освоения и продолжительность сохранения информации в памяти;
- скорость достижения результатов при использовании системы – определяется количеством вводимых или выбираемых мышью команд и настроек;
- субъективная удовлетворенность при эксплуатации системы (удобство работы, утомляемость и т. д.).

Наилучшими характеристиками для пользователей-профессионалов обладают интерфейсы со свободной навигацией, а для пользователей-непрофессионалов – интерфейсы прямого манипулирования. Замечено, что при выполнении операции копирования файлов большинство профессионалов используют оболочки типа Far, а непрофессионалы – «перетаскивание объектов» Windows.

4. Классификация диалогов и общие принципы их разработки

Типы диалога. Тип диалога определяет, кто из «собеседников» управляет процессом обмена информацией.

Различают два типа диалога:

- управляемые программой;
- управляемые пользователем.

Диалог, *управляемый программой*, предусматривает наличие жесткого, линейного или древовидного, т. е. включающего возможные альтернативные варианты, сценария диалога, заложенного в программное обеспечение. Такой диалог обычно сопровождают большим количеством подсказок, которые уточняют, какую информацию необходимо вводить на каждом шаге.

Диалог, *управляемый пользователем*, подразумевает, что сценарий диалога зависит от пользователя, который применяет систему для выполнения необходимых ему операций. При этом система обеспечивает возможность реализации различных пользовательских сценариев.

Форма диалога. Никакой диалог невозможен, если не существует языка, понятного «собеседникам». Описание языка, на котором ведется диалог, включает определение его *синтаксиса* – правил, определяющих допустимые конструкции (слова, предложения) языка или его форму, и *семантики* – правил, определяющих смысл синтаксически корректных конструкций языка или его содержание.

Различают три формы диалога:

- фразовую,
- директивную,
- табличную.

Фразовая форма предполагает «общение» с пользователем на естественном языке или его подмножестве. Содержание диалога составляют повелительные, повествовательные и вопросительные предложения и ответы на вопросы.

Чаще всего используют диалоги, предполагающие односложные ответы.

Например:

Программа: Введите свой возраст (полных лет):

Пользователь

При обработке фраз оперируют понятием словоформа.

Словоформа – отрезок текста между двумя соседними пробелами или знаками препинания.

Морфологический анализ - обработка словоформ вне связи с контекстом.

Два метода морфологического анализа:

- декларативный – предполагает, что в словаре находятся все возможные словоформы каждого слова, тогда анализ сводится к поиску словоформы в словаре. Данный метод обеспечивает возможность обработки сообщений, состоящих из строчных и прописных букв в произвольной комбинации, при чем как латинского, так и русского или других алфавитов;
- процедурный – предполагает выделение в текущей словоформе основы, которую затем идентифицируют.

После распознавания словоформ осуществляют синтаксический анализ сообщения, по результатам которого определяют его синтаксическую структуру, т. е. выполняют разбор предложения.

Далее выполняют *семантический анализ*, т. е. определяют смысловые отношения между словоформами. При этом выделяют главные предикаты, определяющие смысл предложения.

Интерфейс, реализующий фразовую форму диалога, должен: преобразовывать сообщения из естественно-языковой формы в форму внутреннего представления и обратно, выполнять анализ и синтез сообщений пользователя и системы, отслеживать и запоминать пройденную часть диалога.

Недостатки фразовой формы:

- большие затраты ресурсов;
- отсутствие гарантии однозначной интерпретации формулировок;
- необходимость ввода длинных грамматически правильных фраз.

Достоинства фразовой формы – свободное общение с системой.

Директивная форма - использование команд (директив) *специально разработанного формального языка*.

Команда – предложение этого языка, описывающее комбинированные данные, которые включают идентификатор иницируемого процесса и, при необходимости, данные для него.

Команду можно вводить:

- в виде строки текста, специально разработанного формата (команды MS DOS в командной строке);
- нажатием некоторой комбинации клавиш (комбинации «быстрого доступа» Windows-приложений);
- посредством манипулирования мышью («перетаскиванием» пиктограмм);
- комбинацией второго и третьего способов.

Достоинства директивной формы:

- небольшой объем вводимой информации;
- гибкость – возможности выбора операции, ограничивается набором допустимых команд;
- ориентация на диалог, управляемый пользователем;
- использование минимальной области экрана или не использование ее вообще;
- возможность совмещения с другими формами.

Недостатки директивной формы:

- практическое отсутствие подсказок на экране, что требует запоминания вводимых команд и их синтаксиса;
- почти полное отсутствие обратной связи о состоянии иницируемых процессов;
- необходимость навыков ввода текстовой информации или манипуляций мышью;
- отсутствие возможности настройки пользователем.

Директивная форма удобна для пользователя-профессионала, который обычно быстро запоминает синтаксис часто используемых команд или комбинации клавиш. Достоинства формы (гибкость и хорошие временные характеристики) проявляются в этом случае особенно ярко.

Табличная форма – пользователь выбирает ответ из предложенных программой. Язык диалога имеет простейший синтаксис и однозначную семантику, что достаточно легко реализовать. Форма удобна для пользователя, так как выбрать всегда проще, что существенно для пользователя-непрофессионала. Эту форму можно использовать, если множество возможных ответов на конкретный вопрос конечно. Если количество возможных ответов велико (более 20), то применение табличной формы может оказаться нецелесообразным.

Достоинства табличной формы:

- наличие подсказки;
- сокращение количества ошибок ввода: пользователь не вводит информацию, а указывает на нее;
- сокращение времени обучения пользователя;
- возможность совмещения с другими формами;
- в некоторых случаях возможность настройки пользователем.

Недостатки табличной формы:

- необходимость наличия навыков навигации по экрану;
- использование сравнительно большой площади экрана для изображения визуальных компонентов;
- интенсивное использование ресурсов компьютера, связанное с необходимостью постоянного обновления информации на экране.

Типы и формы диалога выбирают независимо друг от друга: любая форма применима для обоих типов диалогов.

Синхронные - диалоги, происходящие в процессе нормальной работы программного обеспечения.

Асинхронные – диалоги, возникающие по инициативе системы или пользователя при нарушении сценария нормального процесса. Их используют для выдачи экстренных сообщений от системы или пользователя.

Разработка диалогов. Стадии проектирования и реализации диалогов:

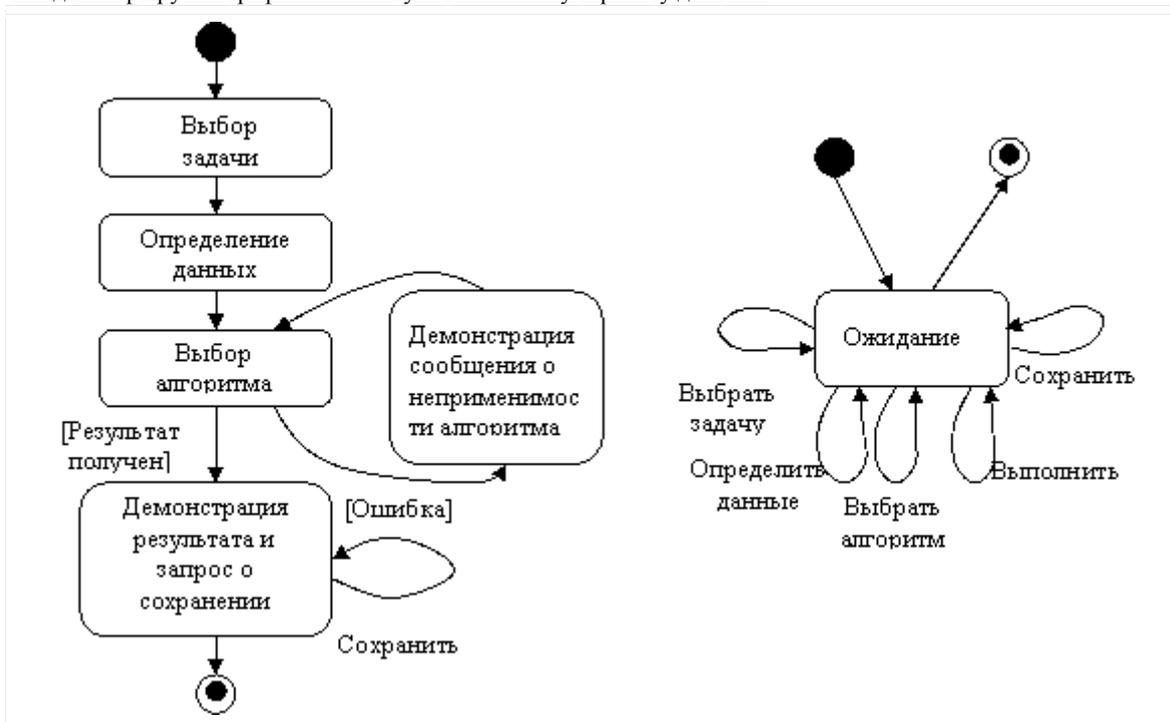
- определение множества необходимых диалогов, их основных сообщений и возможных сценариев – проектирование *абстрактных диалогов*;
- определение типа и формы каждого диалога, а также синтаксиса и семантики используемых языков – проектирование *конкретных диалогов*;
- выбор основных и дополнительных устройств и проектирование процессов ввода-вывода для каждого диалога, а также уточнение передаваемых сообщений – проектирование *технических диалогов*.

Основа абстрактных диалогов – идеология технологического процесса, для автоматизации которого предназначается программный продукт.

Кроме сценариев используют *диаграммы состояний интерфейса* или *графы диалога*.

Граф диалога – ориентированный взвешенный граф, каждой вершине которого сопоставлена конкретная картина на экране (*кадр*) или определенное состояние диалога, характеризующееся набором доступных пользователю действий. Дуги, исходящие из вершин, показывают возможные изменения состояний при выполнении пользователем указанных действий. Дуги, исходящие из вершин, показывают возможные изменения состояний при выполнении пользователем указанных действий. В качестве весов дуг указывают условия переходов из состояния в состояние и операции, выполняемые во время перехода.

Каждый маршрут на графе соответствует возможному варианту диалога.



а б
Рисунок 3 –
Графы абстрактного диалога:
а – диалог, управляемый системой;
б – диалог, управляемый пользователем
5. Основные компоненты графических пользовательских интерфейсов
Графические пользовательские интерфейсы поддерживаются операционными системами

Windows, Apple Macintosh, OS/2 и т. д. Для таких интерфейсов разработаны наборы стандартных компонентов взаимодействия с пользователем для каждой операционной системы.

Интерфейсы строятся по технологии WIMP: W – Windows (окна), I – Icons (пиктограммы), M – Mouse (мышь), P – Pop-up (всплывающие или выпадающие меню). Основные элементы графических интерфейсов: окна, пиктограммы, компоненты ввода-вывода и мышь, которую используют в качестве указывающего устройства и устройства прямого манипулирования объектами на экране.

Окна. *Окно* – прямоугольная, ограниченная рамкой область физического экрана. Окно может менять размеры и местоположение в пределах экрана.

5 категорий окон:

- основные окна (окна приложений);
- дочерние или подчиненные окна;
- окна диалога;
- информационные окна;
- окна меню.

Окно приложения Windows содержит: рамку, ограничивающую рабочую область окна, строку заголовка с кнопкой системного меню и кнопками выбора представления окна и выхода, строку меню, пиктографическое меню (панель инструментов), горизонтальные и вертикальные полосы прокрутки и строку состояния.

Дочернее окно Windows используют в многодокументных программных интерфейсах (MDI). Это окно не содержит меню. В строке заголовка – специальное имя, идентифицирующее связанный с ним документ или файл. Пиктограммы всех дочерних окон одинаковы.

Диалоговое окно Windows используют для просмотра и задания различных режимов работы, необходимых параметров или другой информации.

Оно может содержать:

- строку заголовка с кнопкой системного меню;
- компоненты, обеспечивающие пользователю возможность ввода или выбора ответа;
- вспомогательные компоненты, обеспечивающие подсказку (поле просмотра или кнопка справки).

Размер окна не изменяем, но по экрану его можно перемещать.

Информационные окна двух типов:

- окна сообщений;
- окна помощи.

Окна сообщений содержат: заголовок с кнопкой системного меню, текст сообщения, одна или несколько кнопок реакции пользователя (Yes, No, Cancel).

Окно помощи содержит: меню, полосы прокрутки, информационная область, аналогично окну приложения, но имеет узкоспециальное назначение.

Окна меню Windows используют как открывающиеся панели иерархического меню или как контекстные меню.

Каждой строке окна меню может соответствовать:

- команда;
- меню следующего уровня, что обеспечивается стрелкой;
- окно диалога, что обозначается тремя точками.

Добавляется указание клавиш быстрого вызова.

Пиктограммы. Пиктограмма – небольшое окно с графическим изображением, отражающим содержимое буфера, с которым она связана.

Виды пиктограмм:

- программные, связанные с соответствующей программой;
- пиктограммы дочерних окон, обеспечивающие доступ к различным документам;
- пиктограммы панели инструментов, дублируют доступ к соответствующим функциям через меню, обеспечивая их быстрый доступ;
- пиктограммы объектов, для прямого манипулирования объектами.

Прямое манипулирование изображением. *Прямое манипулирование изображением* – это возможность замены команды воздействия на некоторый объект физическим действием в интерфейсе, осуществляемым с помощью мыши. При этом любая область экрана рассматривается как адресат, который может быть активизирован при подведении курсора и нажатии клавиши мыши.

По реакции на воздействие различают типы адресатов:

- указание и выбор (развертывание пиктограмм, определение активного окна);
- буксировка и «резиновая нить» (перенос объекта или его границ);
- экранные кнопки и «скользящие» барьеры (выполнение или циклически повторяемых действий (выполнение некоторых операций или рисование, подразумеваемых при активизации определенной области экрана - кнопки)).

Динамический визуальный сигнал - изменение изображения на экране (курсор мыши при выполнении конкретных операций, изменения изображения кнопки).

Компоненты ввода-вывода. Интерфейсы включают несколько меню: основное или «ниспадающее» иерархическое меню, пиктографическое меню (панели инструментов) и контекстные меню для разных ситуаций. Любое из указанных меню представляет собой компонент ввода-вывода, реализующий диалог с пользователем, используя табличную форму.

Иерархическое меню используют, чтобы организовать выполняемые программным обеспечением операции, если их число превышает в соответствии с рекомендациями фирмы IBM), и обеспечить пользователю их обзор. Панели инструментов и контекстное меню применяют для обеспечения быстрого доступа к часто используемым командам, обеспечивая пользователю возможность относительно свободной навигации.

Другие формы ввода-вывода:

- фразовая,
- табличная,
- смешанная.

6. Реализация диалогов в графическом пользовательском интерфейсе

Диалоги обоих типов:

- управляемые пользователем,
- управляемые системой.

Реализация диалогов, управляемых пользователем. Для реализации применяют меню различных видов:

- основное,
- панели инструментов,
- контекстные и кнопочные.

Как альтернативу меню целесообразно использовать директивную форму диалога, поставив в соответствие основным командам определенные комбинации клавиш. Целесообразно предусмотреть возможность управления меню клавиатурой, если большую часть времени работы с системой пользователь вводит текст или данные, т. е. взаимодействует с клавиатурой.

Меню. Меню проектируют на основе графов диалогов разрабатываемого программного обеспечения. Если число операций не превышает 5, то обычно используют кнопки. Если число операций не более 9-10, то – одноуровневое меню. Если число операций более 10, то используют «ниспадающее» двухуровневое иерархическое меню.

Ниспадающее меню. Первый уровень иерархического меню должен содержать имена основных групп операций.

Традиционно (обычно в текстовых и графических редакторах):

1. пункт Файл,
 2. пункт Правка,
 3. пункт Вид,
- последний пункт Справка.

Количество уровней иерархического меню не должно превышать 2-3 (сложно искать). Число операций в окне не должно превышать 7-8 операций.

Если число операций превышает 70-80. Разработчики Microsoft Word предложили *адаптивное* иерархическое меню, где содержимое окна меню второго уровня постоянно меняется, отображая только те операции, которые использует пользователь.

Если пользователь не находит нужной операции, то через несколько секунд или при нажатии специальной кнопки Word демонстрирует окно меню полностью.

7 Пользовательские интерфейсы прямого манипулирования и их проектирование

Возможность прямого манипулирования, предусмотренная в WIMP интерфейсах, позволяет разрабатывать для приложений объектно-ориентированные интерфейсы прямого манипулирования.

Интерфейсы используют директивную форму диалога: ввод команды осуществляется при выполнении определенных действий с пиктограммой объекта мышью. Основными элементами этих интерфейсов являются: метафоры, объекты, представления объектов и технологии Drag and Drop («перетащил и бросил»).

Метафоры. *Метафоры* – мысленный перенос свойств или признаков одного объекта на другой, чем-то аналогичный первому. Использование метафор в интерфейсах предполагает активизацию имеющегося у пользователя опыта.

Интерфейс прямого манипулирования должен обеспечить пользователю среду, содержащую знакомые элементы, с которыми пользователь не раз встречался в профессиональной деятельности или в быту, и предоставлять ему возможность манипулирования отдельными объектами. (Метафора «Выбрасывание мусора» – для удаления файлов).

Похожие элементы должны вести себя похожим образом, элементы, выделенные одним цветом, должны находиться в определенной связи друг с другом.

Целесообразно не делать изображения слишком реалистичными, чтобы не обмануть ожидания пользователя.

Метафоры и анимация. При реализации метафор все большая роль уделяется средствам мультимедиа, в основном анимации. Используя мультипликацию, можно не только развлекать пользователя, но и «готовить» его к смене кадров, сокращая время, необходимое для адаптации к изменившейся ситуации.

Программа, реализующая анимационные интерфейсы, никогда не простаивает, так как во время ожидания ввода команды пользователя она продолжает отображать соответствующие кадры. В основе таких программ лежит *временное программирование*. В отличие от событийного программирования, которое позволяет связывать изображение на экране с внешними и внутренними событиями в системе, временное программирование обеспечивает изменение проецируемой *последовательности кадров* в зависимости от состояния моделируемых процессов и действий пользователя.

Объекты интерфейса прямого манипулирования и их представления.

Три основные типа объектов интерфейсов прямого манипулирования:

- объекты-данные,
- объекты контейнеры,
- объекты устройства.

Объекты-данные снабжают пользователя информацией (тексты, изображения, электронные таблицы, музыка, видео). В рамках операционной системы таким объектам соответствуют приложения, которые запускаются при раскрытии объекта.

Объекты-контейнеры могут манипулировать своими внутренними объектами, в том числе и другими контейнерами (копировать их или сортировать в любом порядке). К типичным контейнерам относятся папки, корзины. При раскрытии контейнера демонстрируются сохраняемые им компоненты, и появляется возможность ими манипулировать. Компоненты могут обозначаться пиктограммами или представляться в виде таблицы.

Объекты-устройства представляют устройства, существующие в реальном мире: телефоны, факсы, принтеры и т. д. их используют для обозначения этих устройств в абстрактном мире интерфейса. При раскрытии такого объекта можно увидеть его настройки.

Каждому объекту соответствует одно окно. В исходном состоянии это окно представлено пиктограммой, но при необходимости его можно раскрыть и выполнить требуемые операции, например настройки объекта. Окно объекта в раскрытом состоянии может содержать меню и панели инструментов. Пиктограмме же должно соответствовать контекстное меню, содержащее перечень операций над объектом.

Имя пиктограммы формируют по своему для каждого типа объектов. Пиктограммам объектов-данных присваивают имена, соответствующие именам хранимых данных, а тип данных кодируется самой пиктограммой. Имя пиктограммы-контейнера или пиктограммы устройства обозначает сам объект, а потому не зависит от содержимого.

Различие между типами объектов является условным, так как один и тот же объект в разных ситуациях может вести себя то, как объект-данные, то, как объект-устройство, то, как объект-контейнер (принтер – объект-устройство, может обладать свойством объекта-контейнера, может содержать объекты-данные в очереди на печать; представление в виде пиктограммы, окна очереди на печать, окна настроек; имя представления целесообразно указывать в заголовке окна объекта).

Технология Drag and Drop. Основные принципы прямого манипулирования, описанные в руководстве по разработке пользовательских интерфейсов фирмы IBM:

- результат перемещения объекта должен соответствовать ожиданиям пользователя;
- пользователи не должны неожиданно терять информацию;
- пользователь должен иметь возможность отменить неправильное действие.

Основные принципы визуализации операции прямого манипулирования:

- исходное выделение – используется в качестве обратной связи пользователю, чтобы сообщить ему, что объект захвачен, в Windows с этой целью используется выделение цветом;
- визуализация перемещения – используется для идентификации выполняемого действия;
- целевое выделение – используется для идентификации пункта назначения, показывая, таким образом, куда «упадет» объект, если его отпустить в текущий момент времени;

- визуализация действия – используется для обозначения времени ожидания завершения операции, обычно с этой целью применяют анимацию или изменение формы курсора на «песочные часы».

Существует два вида пунктов назначения: один принимает объект, а другой его копию (Пользователь «бросает» документ в «корзину» – уничтожается сам документ, а если на принтер, то передается копия документа).

Проектирование интерфейсов прямого манипулирования. Проектирование выполняется на основе графов диалога, разработанных для конкретного программного обеспечения, и включает следующие процедуры:

- формирование *множества объектов предметной области*, которое должно быть представлено на экране, причем в качестве основы в этом случае используют не варианты использования, а концептуальную модель предметной области;
- анализ *объектов*, определение их *типов и представлений*, а также перечня операций с этими объектами;
- уточнение *взаимодействия объектов* и построение матрицы *прямого манипулирования*;
- определение *визуальных представлений* объектов;
- разработка *меню окон объектов и контекстных меню*;
- создание *прототипа* интерфейса;
- тестирование на *удобство использования*.

8 Интеллектуальные элементы пользовательских интерфейсов

Элементы пользовательских интерфейсов: Мастер, Советчик, Агент. Сделано множество попыток создания социализированного пользовательского интерфейса. В основе такого интерфейс лежит идея создания персонализированного, т. е. «имеющего личность», интерфейса. Развлекающие программы, такие как Cats(кошки) и Dogs(собаки), реализующие сложное поведение домашних животных в разных ситуациях, показывают, что технически это вполне решаемая задача.

Советчики. Представляют собой форму подсказки. Их можно вызвать с помощью меню справки, командной строки окна или из всплывающего меню. Советчики помогают пользователям в выполнении конкретных задач.

Мастера. Программу-мастер используют для выполнения общераспространенных, но редко выполняемых отдельным пользователем задач (установка программ или оборудования). Выполнение подобных действий требует от пользователя принятия сложных взаимосвязанных решений, последовательность которых диктует программа-мастер. Интеллектуальные Мастера способны на каждом шаге демонстрировать в окне просмотра результаты ответов пользователя на предыдущие вопросы, помогая последнему сориентироваться в ситуации.

Мастер реализует последовательный или древовидный сценарий диалога. Его целесообразно использовать для решения хорошо структурированных, последовательных задач.

При этом необходимо:

- предоставить пользователю возможность возврата на предыдущий шаг;
- предусмотреть возможность отмены работы Мастера;
- нумеровать шаги и сообщать пользователю количество шагов Мастера, особенно, если таких шагов больше трех;
- пояснить пользователю каждый шаг;
- по возможности демонстрировать результат уже выполненных операций на каждом шаге.

Программные агенты. Используются для выполнения рутинной работы. Основными функциями Агентов-Помощников являются: наблюдение, поиски управление. Различают:

программы-агенты, настраиваемые на выполнение указанных задач;

программы-агенты, способные обучаться (фиксируя действия пользователя (по типу магнитофона)).

Практическое занятие:

по теме: «Настройка доступа к сетевым устройствам»

Цель: научиться настраивать доступ к сетевым устройствам

По завершению практического занятия студент должен уметь: настраивать доступ к сетевым устройствам

Продолжительность: 3 аудиторных часа (135 минут)

Необходимые принадлежности

Компьютеры.

Ход выполнения работы

1. Если монитор вычислительной системы имеет питание, отдельное от системного блока, включите монитор.
2. Включите компьютерную систему выключателем системного блока.
3. При появлении запроса о пароле нажмите на клавиатуре клавишу Esc.

Установка контроллера удаленного доступа

4. Нажмите кнопку **Пуск** на панели задач. Выберите пункт **Настройка -> Панель Управления**.
5. Откройте объект **Установка и удаление программ**. В появившемся окне:
 - 5.1. на вкладке **Установка Windows** в окне **Компоненты** выберите пункт **Связь** и нажмите кнопку **Состав...**
 - 5.2. В появившемся окне выберите пункт (установите флажок) **Удаленный доступ к сети** и нажмите кнопку **ОК**.
6. Подождите, пока система устанавливает программное обеспечение. По завершении перезагрузите компьютер.

Установка модема

7. Нажмите кнопку **Пуск** на панели задач. Выберите пункт **Настройка -> Панель Управления**.
8. Откройте объект **Модемы** (появится диалоговое окно **Установка нового модема**).
 1. Установите флажок **Не определять тип модема** (выбор из списка). Нажмите кнопку **Далее >**.
 2. Прочитайте и законспектируйте сообщение. Выберите соответствующие пункты в окнах **Изготовители: (Standard Modem Types)** и **Модели: Standard 28800 bps Modem**. Нажмите кнопку **Далее >**.
 3. В окне **Укажите порт, к которому он присоединен: укажите Последовательный порт (COM2)**. Нажмите кнопку **Далее >**.
 4. Подождите, пока идет установка модема. По завершении нажмите кнопку **Готово**.

Создание удаленного соединения

9. Откройте объект **Мой компьютер**.
10. Откройте объект **Удаленный доступ к сети**.
11. Откройте объект **Новое соединение**. В появившемся окне:
 - 11.1. введите название соединения **Лаб9**; выберите в выпадающем списке установленный модем. Нажмите кнопку **Далее...**
 - 11.2. введите Код города: **222**; Телефон: **22222**, Код страны: **Россия (7)**. Нажмите кнопку **Далее...**
 - 11.3. нажмите кнопку **Готово**

Настройка удаленного соединения

12. В окне **Удаленный доступ к сети** выберите объект **Лаб 9**. Выберите в меню **Файл** пункт Свойства. В открывшемся окне:
 - 12.1. на вкладке **Общие** проверьте код города, код страны, телефон.
 - 12.2. на вкладке **Тип сервера** отметьте тип удаленного сервера; установите **Допустимые сетевые протоколы**: TCP/IP.
 - 12.3. нажмите кнопку **ОК**.

Установка удаленного соединения

13. В окне **Удаленный доступ к сети** откройте объект **Лаб 9**. В открывшемся окне:
 - 13.1. введите Имя пользователя: dial-up
 - 13.2. введите Пароль: 12345
 - 13.3. нажмите кнопку **Установить связь**

Фазы установления соединения:

14. набор номера
15. согласование параметров связи
16. проверка имени пользователя и пароля
17. вход в сеть
18. установка соединения

Завершение работы

19. Нажмите кнопку Пуск на панели задач. Выберите пункт **Настройка** -> **Панель Управления**.
20. В окне **Удаленный доступ к сети** выберите объект **Лаб 9**. Выберите в меню **Файл** пункт Удалить.
21. Откройте объект **Модемы**. Выберите Standard 28800 bps Modem. Нажмите кнопку Удалить Нажмите кнопку Закрывать
22. Откройте объект **Сеть**. Выберите Контроллер удаленного доступа. Нажмите кнопку Удалить Нажмите кнопку ОК
23. Уточните у преподавателя порядок завершения работы с компьютером. Приведите компьютер в исходное состояние.

Контрольные вопросы

1. Порядок настройки удаленного доступа в сеть.
2. Что такое:
ISP, DCE, DTE, канал передачи данных, модем?
3. Модемы: назначение, типы, выполняемые функции, протоколы.
4. Протоколы канального уровня: UUCP, SLIP, PPP.
5. Фазы установления удаленного соединения.

Практическое занятие:

по теме: «Настройка политики безопасности»

Цель: научиться настраивать доступ к сетевым устройствам

По завершению практического занятия студент должен уметь: настраивать доступ к сетевым устройствам

Продолжительность: 3 аудиторных часа (125 минут)

Необходимые принадлежности

Компьютеры.

Основные сведения

При физическом соединении двух или более компьютеров образуется компьютерная сеть. Компьютерная сеть представляет собой комплекс технических, коммуникационных и программных средств, обеспечивающих эффективное распределение вычислительных ресурсов.

Уже сейчас есть сферы человеческой деятельности, которые принципиально не могут существовать без сетей (например, работа банков, крупных библиотек и т. д.) Сети используются при управлении крупными автоматизированными производствами, газопроводами, электростанциями и т.п.

В общем случае, для создания компьютерных сетей необходимо специальное аппаратное обеспечение - сетевое оборудование и специальное программное обеспечение - сетевые программные средства. Назначение всех видов компьютерных сетей определяется двумя функциями:

- ✓ обеспечение совместного использования аппаратных и программных ресурсов сети;
- ✓ обеспечение совместного доступа к ресурсам данных.

Например, все участники локальной сети могут совместно использовать одно общее устройство печати - сетевой принтер или, например, ресурсы жестких дисков одного выделенного компьютера - файлового сервера. Аналогично можно совместно использовать и программное обеспечение. Если в сети имеется специальный компьютер, выделенный для совместного использования участниками сети, он называется файловым сервером. Основными компонентами сети являются рабочие станции, серверы, передающие среды (кабели) и сетевое оборудование.

Рабочими станциями называются компьютеры сети, на которых пользователями сети реализуются прикладные задачи.

Серверы сети - это аппаратно-программные системы, выполняющие функции управления распределением сетевых ресурсов общего доступа. Сервером может быть это любой подключенный к сети компьютер, на котором находятся ресурсы, используемые другими устройствами локальной сети. В качестве аппаратной части сервера используется достаточно мощные компьютеры.

Аппаратура локальной сети обычно состоит из кабеля, разъемов, T-коннекторов (рис. 1), терминаторов и сетевых адаптеров. Кабель, очевидно, используется для передачи данных между рабочими станциями. Для подключения кабеля используются разъемы. Эти разъемы через T-коннекторы подключаются к сетевым адаптерам - специальным платам, вставленным в слоты расширения материнской платы рабочей станции. Терминаторы подключаются к открытым концам сети.



Рис. 1. Т-коннектор

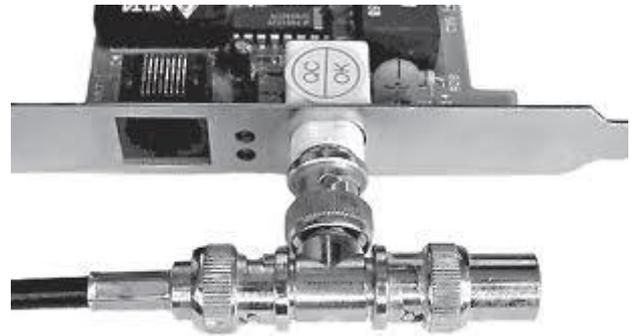


Рис. 2. Т-коннектор, присоединенный к сетевой карте

Для Ethernet (Ethernet — пакетная технология передачи данных преимущественно локальных компьютерных сетей) могут быть использованы кабели разных типов: тонкий коаксиальный кабель, толстый коаксиальный кабель и неэкранированная витая пара. Для каждого типа кабеля используются свои разъемы и свой способ подключения к сетевому адаптеру.

Сети можно создавать с любым из типов кабеля.

1. Витая пара (TP - Twisted Pair) — это кабель, выполненный в виде скрученной пары проводов (рис. 3). Он может быть экранированным и неэкранированным. Экранированный кабель более устойчив к электромагнитным помехам. Витая пара наилучшим образом подходит для малых учреждений. Недостатками данного кабеля является высокий коэффициент затухания сигнала и высокая чувствительность к электромагнитным помехам, поэтому максимальное расстояние между активными устройствами в ЛВС при использовании витой пары должно быть не более 100 метров.



Рис. 3 Кабель на основе витой пары

2. Коаксиальный кабель (рис. 4) состоит из одного цельного или витого центрального проводника, который окружен слоем диэлектрика. Проводящий слой алюминиевой фольги, металлической оплетки или их комбинации окружает диэлектрик и служит одновременно как экран против наводок. Общий изолирующий слой образует внешнюю оболочку кабеля.

Коаксиальный кабель может использоваться в двух различных системах передачи данных: без модуляции сигнала и с модуляцией. В первом случае цифровой сигнал используется в таком виде, в каком он поступает из ПК и сразу же передается по кабелю на приемную станцию. Он имеет один канал передачи со скоростью до 10 Мбит/сек и максимальный радиус действия 4000 м. Во втором случае цифровой сигнал превращают в аналоговый и направляют его на приемную станцию, где он снова превращается в цифровой. Операция превращения сигнала выполняется модемом; каждая станция должна иметь свой модем. Этот способ передачи является многоканальным (обеспечивает передачу по десяткам каналов, используя для этого всего лишь один кабель). Таким способом можно передавать звуки, видео сигналы и другие данные. Длина кабеля может достигать до 50 км.



Рис. 4. Устройство коаксиального кабеля
1 — внутренний проводник (медная проволока),
2 — изоляция (сплошной полиэтилен),
3 — внешний проводник (оплётка из меди),
4 — оболочка (светостабилизированный полиэтилен).

3. Оптоволоконный кабель (рис. 5) является более новой технологией, используемой в сетях.носителем информации является световой луч, который модулируется сетью и принимает форму сигнала.

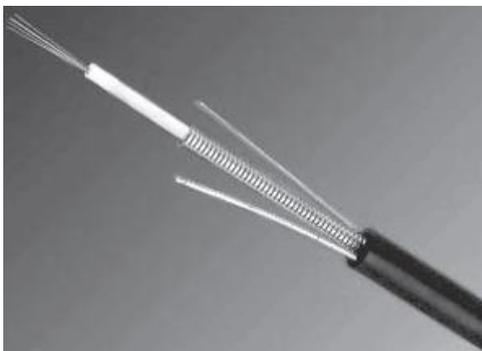


Рис. 5. Оптоволоконный кабель

Такая система устойчива к внешним электрическим помехам и таким образом возможна очень быстрая, секретная и безошибочная передача данных со скоростью до 2 Гбит/с. Количество каналов в таких кабелях огромно. Передача данных выполняется только в симплексном режиме, поэтому для организации обмена данными устройства необходимо соединять двумя оптическими волокнами (на практике оптоволоконный кабель всегда имеет четное, парное кол-во волокон). К недостаткам оптоволоконного кабеля можно отнести большую стоимость, а также сложность подсоединения.

4. Радиоволны в микроволновом диапазоне используются в качестве передающей среды в беспроводных локальных сетях, либо между мостами или шлюзами для связи между локальными сетями. В первом случае максимальное расстояние между станциями составляет 200 - 300 м, во втором - это расстояние прямой видимости. Скорость передачи данных - до 2 Мбит/с.

Выделяют следующие виды сетевого оборудования.

1. Сетевые карты — это контроллеры, подключаемые в слоты расширения материнской платы компьютера, предназначенные для передачи сигналов в сеть и приема сигналов из сети (рис. 6).
2. Терминаторы - это резисторы номиналом 50 Ом, которые производят затухание сигнала на концах сегмента сети.
3. Концентраторы (Hub) — это центральные устройства кабельной системы или сети физической топологии "звезда", которые при получении пакета на один из своих портов пересылает его на все остальные (рис. 7). В результате получается сеть с логи-

ческой структурой общей шины. Различают концентраторы активные и пассивные. Активные концентраторы усиливают полученные сигналы и передают их. Пассивные концентраторы пропускают через себя сигнал, не усиливая и не восстанавливая его.



Рис. 6. Сетевая карта в виде платы расширения, устанавливаемой в PCI-слот



Рис. 7. Концентратор с фиксированным количеством портов

4. Повторители (Repeater)- устройства сети, усиливают и заново формирует форму входящего аналогового сигнала сети на расстоянии другого сегмента (рис. 8). Повторитель действует на электрическом уровне для соединения двух сегментов. Повторители ничего не распознают сетевые адреса и поэтому не могут использоваться для уменьшения трафика.

Повторители (repeater) представляют собой сетевые устройства, функционирующие на первом (физическом) уровне эталонной модели OSI. Для того чтобы понять работу повторителя, необходимо знать, что по мере того, как данные покидают устройство отправителя и выходят в сеть, они преобразуются в электрические или световые импульсы, которые после этого передаются по сетевой передающей среде. Такие импульсы называются сигналами (signals). Когда сигналы покидают передающую станцию, они являются четкими и легко распознаваемыми. Однако чем больше длина кабеля, тем более слабым и менее различимым становится сигнал по мере прохождения по сетевой передающей среде.

Целью использования повторителя является регенерация и ресинхронизация сетевых сигналов на битовом уровне, что позволяет передавать их по среде на большее расстояние. Термин повторитель (repeater) первоначально означал отдельный порт «на входе» некоторого устройства и отдельный порт на его «выходе». В настоящее время используются также повторители с несколькими портами. В эталонной модели OSI повторители классифицируются как устройства первого уровня, поскольку они функционируют только на битовом уровне и не просматривают другую содержащуюся в пакете информацию.



Рис. 8. Повторители (Repeater)

5. Коммутаторы (Switch) - управляемые программным обеспечением центральные устройства кабельной системы, сокращающие сетевой трафик за счет того, что пришедший пакет анализируется для выяснения адреса его получателя и соответственно передается только ему (рис.9).

Использование коммутаторов является более дорогим, но и более производительным решением. Коммутатор обычно значительно более сложное устройство и может обслуживать одновременно несколько запросов. Если по какой-то причине нужный порт в данный момент времени занят, то пакет помещается в буферную память коммутатора, где и дожидается своей очереди. Построенные с помощью коммутаторов сети могут охватывать несколько сотен машин и иметь протяженность в несколько километров.



Рис. 9. Коммутатор

6. Маршрутизаторы (Router) - стандартные устройства сети, работающие на сетевом уровне и позволяющие переадресовывать и маршрутизировать пакеты из одной сети в другую, а также фильтровать широковещательные сообщения (рис. 10).

7. Мосты (Bridge)- устройства сети, которое соединяют два отдельных сегмента, ограниченных своей физической длиной, и передают трафик между ними (рис.11). Мосты также усиливают и конвертируют сигналы для кабеля другого типа. Это позволяет расширить максимальный размер сети, одновременно не нарушая ограничений на максимальную длину кабеля, количество подключенных устройств или количество повторителей на сетевой сегмент.



Рис. 10. Беспроводной маршрутизатор

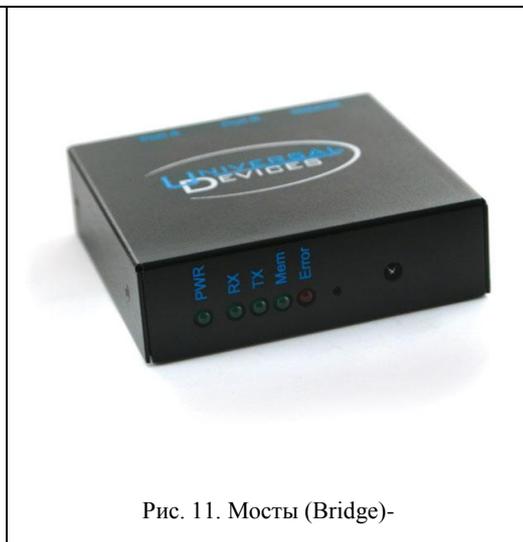


Рис. 11. Мосты (Bridge)-

8. Шлюзы (Gateway) - программно-аппаратные комплексы, соединяющие разнородные сети или сетевые устройства. Шлюзы позволяет решать проблемы различия протоколов или систем адресации. Они действует на сеансовом, представительском и прикладном уровнях модели OSI.

9. Мультиплексоры – это устройства центрального офиса, которое поддерживают несколько сотен цифровых абонентских линий. Мультиплексоры посылают и получают абонентские данные по телефонным линиям, концентрируя весь трафик в одном высокоскоростном канале для передачи в Internet или в сеть компании.

10. Межсетевые экраны (firewall, брандмауэры) - это сетевые устройства, реализующие контроль за поступающей в локальную сеть и выходящей из нее информацией и обеспечивающие защиту локальной сети посредством фильтрации информации. Большинство межсетевых экранов построено на классических моделях разграничения доступа, согласно которым субъекту (пользователю, программе, процессу или сетевому пакету) разрешается или запрещается доступ к какому-либо объекту (файлу или узлу сети) при предъявлении некоторого уникального, присущего только этому субъекту, элемента. В большинстве случаев этим элементом является пароль. В других случаях таким уникальным элементом является микропроцессорные карточки, биометрические характеристики пользователя и т. п. Для сетевого пакета таким элементом является адреса или флаги, находящиеся в заголовке пакета, а также некоторые другие параметры. Таким образом, межсетевой экран - это программный и/или аппаратный барьер между двумя сетями, позволяющий устанавливать только авторизованные межсетевые соединения. Обычно межсетевые экраны защищают соединяемую с Internet корпоративную сеть от проникновения извне и исключает возможность доступа к конфиденциальной информации.

Беспроводные локальные сети считаются перспективным направлением развития ЛС. Их преимущество - простота и мобильность. Также исчезают проблемы, связанные с прокладкой и монтажом кабельных соединений - достаточно установить интерфейсные платы на рабочие станции, и сеть готова к работе.

Сердцем любой беспроводной сети является точка доступа (рис. 12), через которую конечные устройства по радио связываются с корпоративной сетью. Она определяет не только радиус действия и скорость передачи данных, но и решает элементарные задачи управления и обеспечения безопасности.

Хорошие точки доступа оснащаются двумя антеннами, причем в каждый момент времени работает антенна с лучшим качеством приема. Переключение антенн уже на удалении в несколько метров дает повышение качества и, соответственно, скорости передачи по сравнению с «однорукими» точками доступа. Обычно используемые ненаправленные антенны жестко крепятся к корпусу.

Радиохарактеристики точки доступа во многом определяются тем, какие антенны используются. Так, одну и ту же точку доступа с разными антеннами можно использовать для решения разных задач. Если, к примеру, точка доступа применяется в качестве радиомоста между зданиями, удаленными на 2 км или более (до 25 км), то предпочтительнее установить направленную антенну.



Рис. 12. Точка доступа

Программное обеспечение локальных сетей.

После подключения компьютеров к сети необходимо установить на них специальное сетевое программное обеспечение. Существует два подхода к организации сетевого программного обеспечения:

- ✓ сети с централизованным управлением;
- ✓ одно-ранговые сети. Сети с централизованным управлением.

В сети с централизованным управлением выделяются одна или несколько машин, управляющих обменом данными по сети. Диски выделенных машин, которые называются файл-серверами, доступны всем остальным компьютерам сети. На файл-серверах должна работать специальная сетевая операционная система. Обычно это мультизадачная OS, использующая защищенный режим работы процессора.

Остальные компьютеры называются рабочими станциями. Рабочие станции имеют доступ к дискам файл-сервера и совместно используемым принтерам, но и только. С одной рабочей станции нельзя работать с дисками других рабочих станций. С одной стороны, это хорошо, так как пользователи изолированы друг от друга и не могут случайно повредить чужие данные. С другой стороны, для обмена данными пользователи вынуждены использовать диски файл-сервера, создавая для него дополнительную нагрузку.

Есть, однако, специальные программы, работающие в сети с централизованным управлением и позволяющие передавать данные непосредственно от одной рабочей станции к другой минуя файл-сервер. Пример такой программы - программа NetLink. После ее запуска на двух рабочих станциях можно передавать файлы с диска одной станции на диск другой, аналогично тому, как копируются файлы из одного каталога в другой при помощи программы Norton Commander.

На рабочих станциях должно быть установлено специальное программное обеспечение, часто называемое сетевой оболочкой. Это обеспечение работает в среде той OS, которая используется на данной рабочей станции, - DOS, OS/2 и т.д.

Файл-серверы могут быть выделенными или невыделенными. В первом случае файл-сервер не может использоваться как рабочая станция и выполняет только задачи управления сетью. Во втором случае параллельно с задачей управления сетью файл-сервер выполняет обычные пользовательские программы в среде MS-DOS. Однако при этом снижается производительность файл-сервера и надежность работы всей сети в целом, так как ошибка в пользовательской программе, запущенной на файл-сервере, может привести к остановке работы всей сети. Поэтому не рекомендуется использовать невыделенные файл-серверы, особенно в ответственных случаях.

Существуют различные сетевые OS, ориентированные на сети с централизованным управлением. Самые известные из них - Novell NetWare, Microsoft Lan Manager (на базе OS/2), а также выполненная на базе UNIX System V сетевая OS VINES.

Задание

1. Изучить назначение и основные функции аппаратного обеспечения компьютерных сетей
2. Изучить программное обеспечение компьютерных сетей
3. Выполнить настройку общего доступа к принтеру локальной сети
4. Ответить в тетради для практических и лабораторных работ на контрольные вопросы.
5. Создать презентацию на тему «Локальные вычислительные сети. Программное обеспечение сетей».

Презентация должна соответствовать следующим требованиям:

- а) на первом слайде указывается тема презентации, курс и группа учащегося, Ф. И. О. учащегося и Ф.И. О. руководителя;
- б) шрифт текста: Times Roman, 14, интервал 1,15. Заголовки Times Roman 18(можно использовать жирный);
- в) цветовая гамма спокойных тонов. Текст должен быть читаемым;
- г) картинки и фотографии должны быть понятными, если необходимо подписаны или прокомментированы.
- д) на последнем слайде обязательно благодарим за внимание.

Контрольные вопросы

1. Каковы преимущества беспроводных локальных сетей?
2. Каково назначение точки доступа?
3. Чем отличаются сети с выделенным сервером от одноранговых сетей?
4. Что такое технология клиент-сервер?
5. Приведите примеры сетевых операционных систем.
6. Что представляет собой проводник витая пара?
7. Каково устройство коаксиального кабеля?
8. Почему оптоволоконный кабель является приоритетным для проводных сетей? В чем его недостатки?
9. Что такое шлюзы? Какими могут быть шлюзы?
10. Зачем нужны повторители?
11. В чем состоят преимущества использования коммутаторов?
12. Для чего служит межсетевой экран (брандмауэр)?
13. Что такое концентратор?

Лабораторное занятие:

по теме: «Выполнение задач тестирования в процессе внедрения»

Цель: изучить классификацию видов тестирования, практически закрепить эти знания путем генерации тестов различных видов, научиться планировать тестовые активности в зависимости от специфики поставляемой на тестирование функциональности.

Продолжительность: 2 аудиторных часа (90 минут)

Необходимые принадлежности

Компьютеры.

Задание

Порядок выполнения работы

1. Получить задание у преподавателя.
2. Выполнить генерацию тестов различных видов для конкретного объекта реального мира (пример приведен на рисунке 1).
3. Спланировать тестовые активности для следующих задач:
 - 3.1. Поставлен на тестирование модуль 1, модуль 2, модуль
 - 3.3.2. Проведены исправления (fix) для заведенных дефектов, доставлена новая функциональность –модуль
 - 3.3.3. Заказчик решил расширять рынки сбыта и просит осуществить поддержку для Великобритании (кроме уже существующей Беларуси).
- 3.4. Заказчик хочет убедиться, что ПО держит нагрузку в 2000 пользователей.
4. Оформить отчет и защитить лабораторную работу

МДК. 06.02 Инженерно-техническая поддержка сопровождения информационных систем

Текущий контроль

бсеместр

Лабораторное занятие:

по теме: «Разработка плана резервного копирования»

Цель: научиться разрабатывать план резервного копирования

Продолжительность: 6 аудиторных часов (270 минут)

Необходимые принадлежности

Компьютеры.

Задание

Порядок выполнения работы

Прежде чем приступить к созданию самого плана кратко определим конфигурацию копируемой системы и параметры плана резервного копирования.

Структура дисков копируемого сервера

До начала создания плана резервного копирования рассмотрим структуру дисков копируемого сервера или компьютера. Как правило это системный диск и диск с данными. При этом это могут быть два логических диска как на одном физическом, так и на двух разных, также это могут быть и отдельные тома входящие в состав RAID (с т.ч. RAID 0 для системного диска).

В любом случае необходимо копировать и системный диск и диск с данными. Для обоих дисков лучше всего создать резервную копию тома на дисковом а не на файловом уровне. Резервное копирование путем создания образа всего диска более быстрый процесс, более надежно и позволяет полностью сохранить структуру файлов/папок на томе. Это важно не только для системного диска, где определенные файлы должны находиться в определенных папках, но также и для дисков с данными, так как многие базы данных имеют сложные структуры файлов/папок.

В нашем примере мы будем копировать два логических диска на одном физическом: системный диск (C:) и диск с данными (D:).

План резервного копирования

Зададим чтобы резервная копия создавалась в автономном режиме после окончания рабочего дня и в течение выходных. В этом случае данный процесс не будет пересекаться с основной работой, когда требуется доступ к данным.

План резервного копирования **системного диска** включает следующее:

- Создавать **полный образ** первое воскресенье каждого месяца в 14:00.
- Создавать **образ в дифференциальном режиме** каждое воскресенье в 17:00.
- Резервные копии хранятся в течение трех месяцев.

Данный план позволит произвести "откат" системы на состояние любой недели в течение последних трех месяцев. Например, если **критический сбой в работе системы** произошел во вторник, то необходимо будет переделать только то что было сделано момента создания последней резервной копии, т.е. с 17:00 предыдущего воскресенья. Это означает что будут утрачены только около 2 дней работы. Самое большее, что вы можете утратить, это изменения конфигурации системы за последние 7 дней - например, если сбой в работе системы случился в воскресенье до 17:00. Это достаточно приемлемый уровень риска учитывая что какие-либо изменения системы выполняются не часто.

План резервного копирования **диска с данными** включает следующее:

- Создавать **полный образ** первую субботу каждого месяца в 17:00.
- Создавать **образ в дифференциальном режиме** каждое воскресенье в 17:00.
- Создавать **образ в инкрементальном режиме** каждый рабочий день в 23:00.
- Резервные копии хранятся в течение трех месяцев.

В соответствии с данным планом состояние данных сохраняется один раз в день, при этом возможно произвести "откат" данных на состояние любого дня в течение последних трех месяцев. Резервная копия создается по субботам а не по воскресеньям и поэтому данное действие не пересекается с копированием системного диска. В этом случае максимум что может быть утрачено это изменения сделанные в течение одного дня - например, если вы сохранили файл во вторник утром и после обеда произошел **сбой в работе системы**, то можно будет восстановить состояние файлов на вечер понедельника. Если файл был сохранен во вторник вечером до 23:00, а сбой в работе системы произошел на следующее утро, то ничего не будет утрачено.

Режимы Создания Образа: Преимущества и Недостатки

Полный образ	Файл содержащий полный образ всего диска. Преимущества: Для восстановления всего диска необходим только этот единственный файл. Недостатки: Большой размер.
Образ в дифференциальном режиме	Файл содержащий изменения, сделанные на диске от момента создания полного образа до текущего момента. Преимущества: Размер меньше размера полного образа. Недостатки: Размер как правило больше размера инкрементального образа. Для восстановления диска необходим один образ в дифференциальном режиме и полный образ.
Образ в инкрементальном режиме	Файл содержащий изменения, сделанные на диске от момента создания последнего образа (полного, дифференциального или инкрементального) до текущего момента. Преимущества: Меньший размер по сравнению с дифференциальным или полным образом. Недостатки: Для восстановления данных могут потребоваться другие инкрементальные/дифференциальные образы помимо полного образа. Если какой-либо из инкрементальных образов поврежден, то не удастся также восстановить и данные из более поздних инкрементальных копий.

Место хранения резервных копий

Наиболее безопасным местом для хранения резервных копий будет удаленный сервер, физически расположенный где-то в другом месте. Это убережет сервер с резервными копиями и ваши данные от, например, пожара или каких-либо других форс-мажорных обстоятельств, которые могут случиться в вашем здании. Другой вариант это хранить резервные копии на каком-либо другом оборудовании в том же здании. Это защитит вас от утраты данных при сбое в работе аппаратной части (т.к. если резервные копии находятся на используемом в работе физическом диске, то при сбое могут быть утрачены как сами данные, так и их резервные копии).



Рис.1. Рекомендуемая схема сети при резервном копировании небольшого корпоративного сервера

Кликните по изображению для его увеличения

В качестве удаленного сервера не обязательно использовать какое-либо дорогостоящее оборудование. Здесь вполне может подойти экономичное и надежное сетевое хранилище данных (устройство NAS). Его возможно приобрести практически в любом магазине торгующем компьютерами и IT оборудованием.

Обратите внимание, что многие устройства NAS выпускаются с поддержкой внутренних томов RAID. Для хранения резервных копий используйте отказоустойчивые уровни RAID - например, RAID 1, RAID 4, RAID 5 или RAID 6. Не используйте чередующиеся тома (RAID 0) - если один из его дисков выйдет из строя, то все данные будут утрачены.

R-Drive Image совместим с любой серверной платформой поддерживающей сетевой протокол SMB. Среди них Windows, MAC OS и Linux. В большинстве устройствах NAS используются разновидности ОС Linux.

В нашем примере мы будем хранить резервную копию на удаленном сервере в сети (*BCK-UBUNTU*).

Email уведомления

После завершения создания резервной копии отправляется сообщение электронной почты о результатах операции. Его пример можно посмотреть в разделе *Примеры E-mail Уведомления* в конце данной статьи.

Копирование Системного Диска

Ежемесячный полный образ системного диска

Первая часть в плане резервного копирования системного диска это ежемесячный полный образ создаваемый в первое воскресенье каждого месяца в 14:00.

1. На этапе *Выбор Действия (Action Selection)* нажмите **Планировщик задач / Создать Скрипт (Scheduler / Create a Script)**.

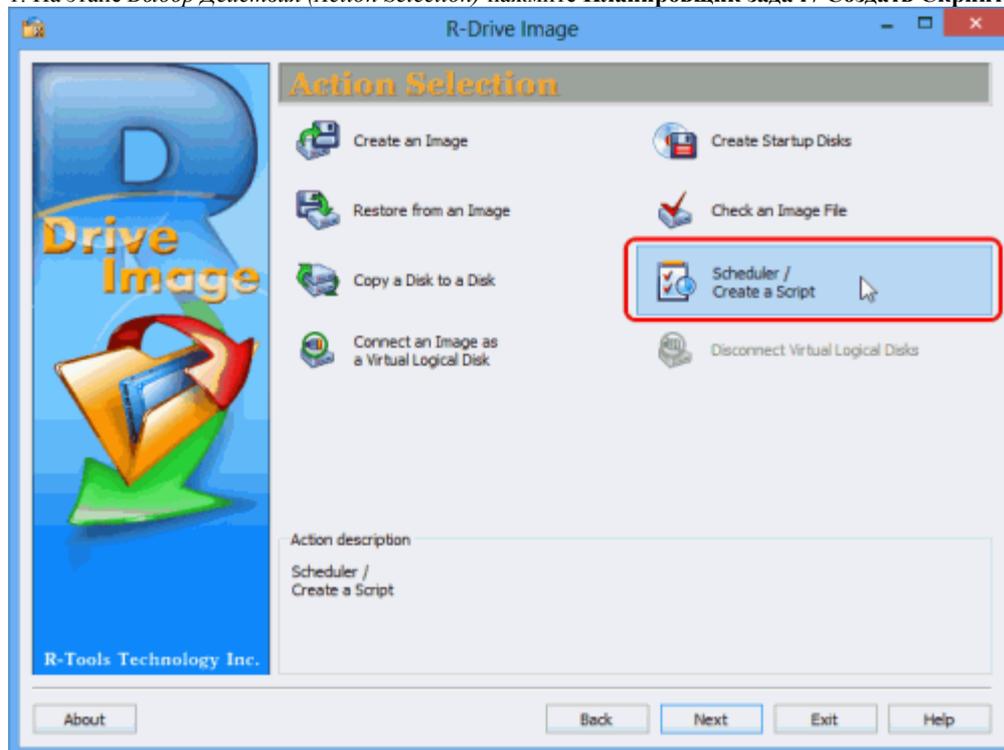


Рис. 2. Ежемесячный полный образ системного диска - этап Выбор Действия (Action Selection)

Кликните по изображению для его увеличения

2. На этапе *Расписание выполнения Задач (Scheduled Tasks)* нажмите кнопку **Создать задачу (Create a task)**.

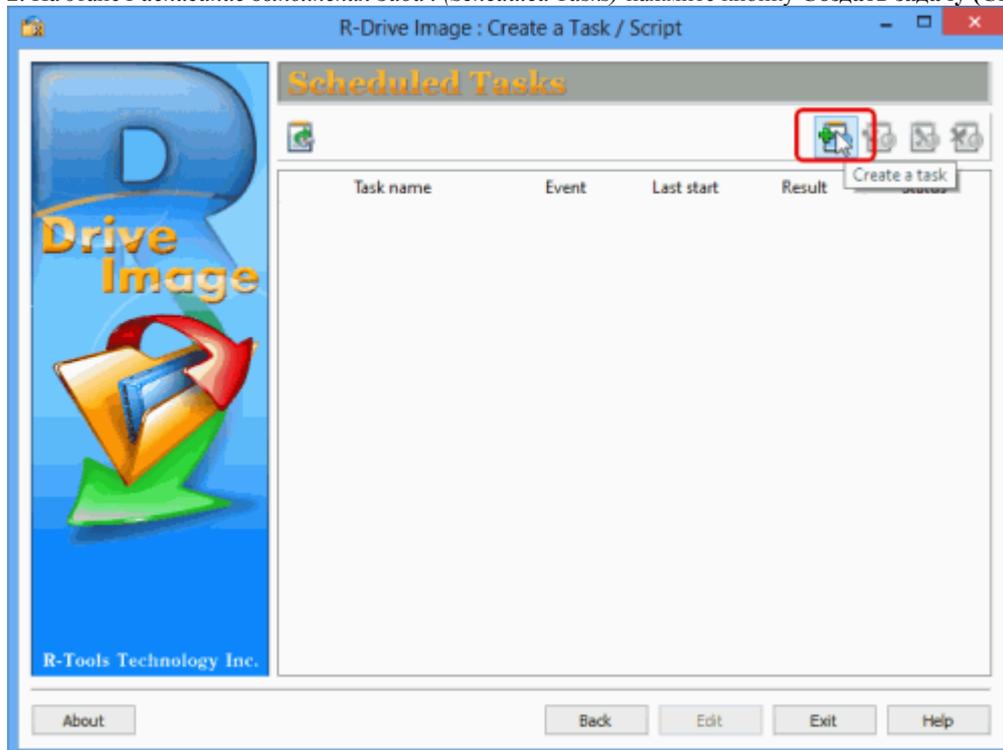


Рис. 3. Ежемесячный полный образ системного диска - этап Расписание выполнения Задач (Scheduled Tasks)

Кликните по изображению для его увеличения

3. На этапе *Выбор Раздела (Partition Selection)* выберите системный раздел резервную копию которого вы будете создавать. В нашем примере системный раздел это **C:**.

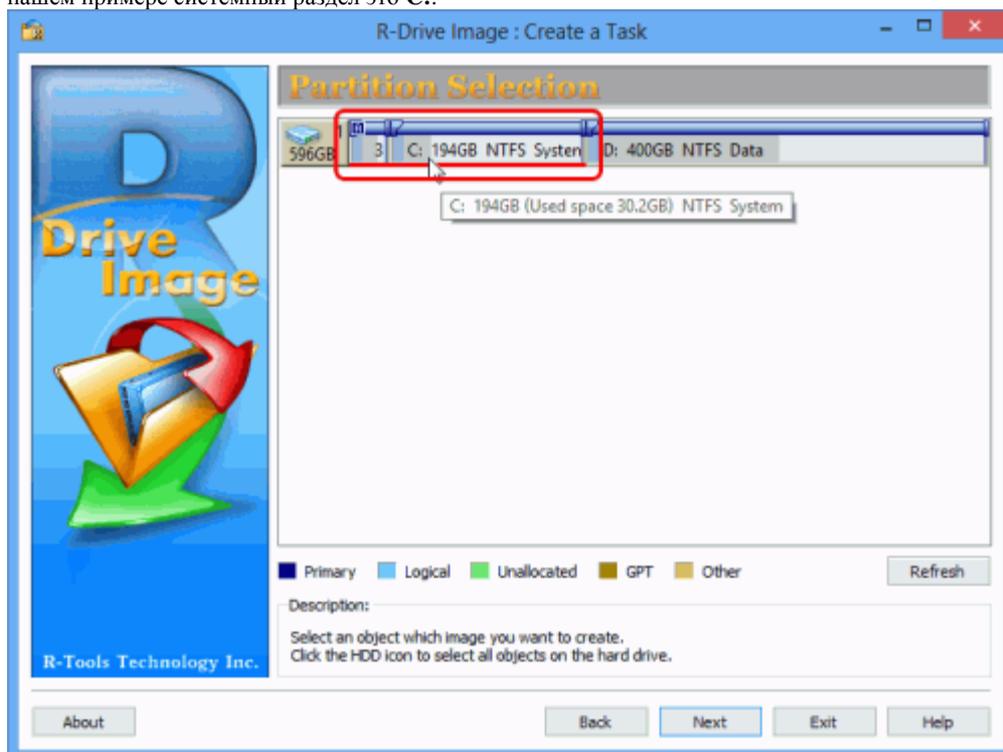


Рис. 4. Ежемесячный полный образ системного диска - этап Выбор Раздела (Partition Selection)

Кликните по изображению для его увеличения

При создании резервной копии системного диска Windows 7 и более поздней версии Windows не забудьте также выбрать небольшой активный раздел на котором находится загрузчик системы. В более ранних версиях Windows такого раздела нет.

4. На этапе *Месторасположение Образа (Image Destination)* выберите месторасположение файла образа и имя файла.

В нашем примере мы выберем находящийся в сети сервер для резервных копий **ВСК-UBUNTU** и папку **Backups**. Может потребоваться ввести имя пользователя и пароль для получения доступа к папке сетевого диска.

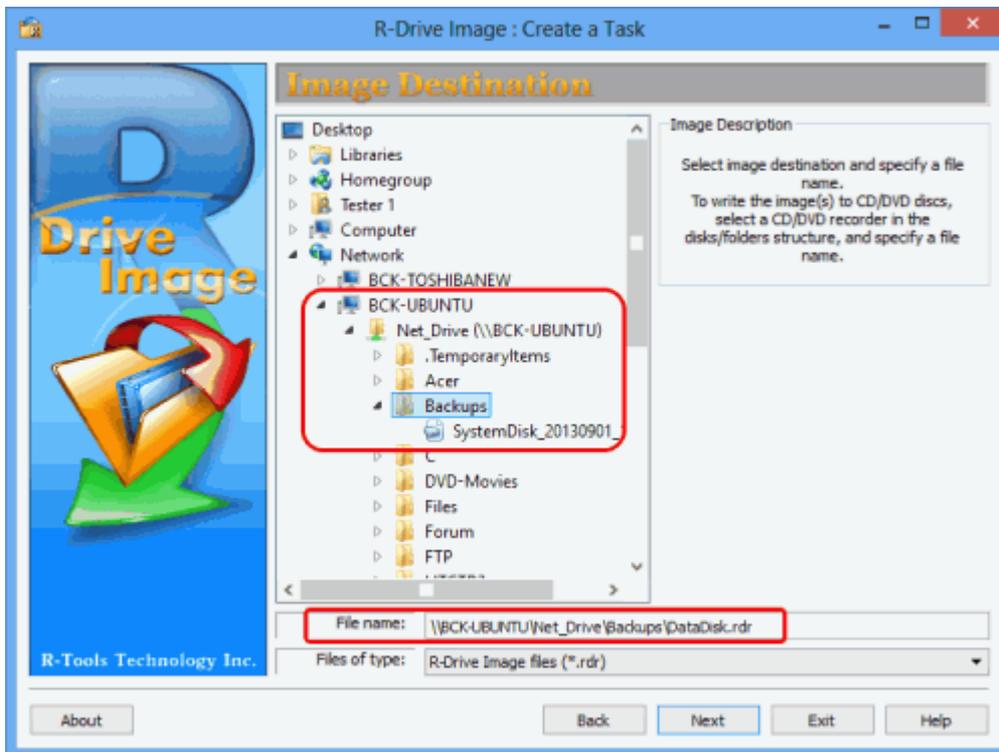


Рис. 5. Ежемесячный полный образ системного диска - этап Месторасположение Образа (Image Destination)

Кликните по изображению для его увеличения

5. Задайте параметры резервных комплектов на этапе *Режим Создания Образа (Imaging Mode)* как показано на нижеприведенном рисунке.

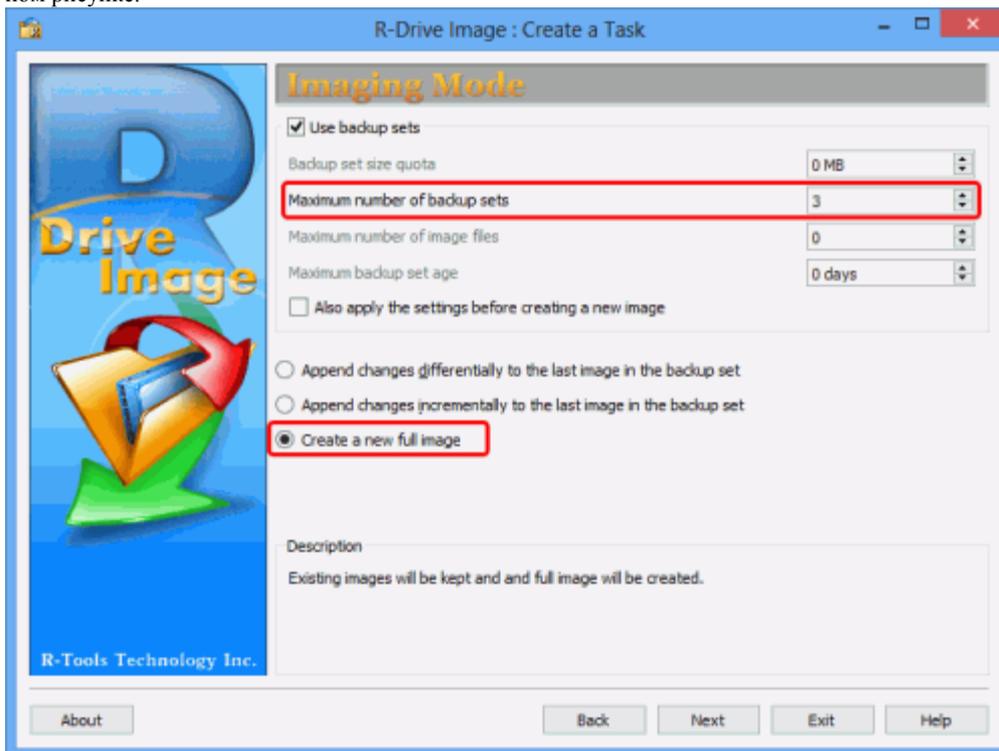


Рис. 6. Ежемесячный полный образ системного диска - этап Режим Создания Образа (Imaging Mode)

Кликните по изображению для его увеличения

Введите **3** в поле *Максимальное число резервных комплектов (Maximum number of backup sets)* и установите радиокнопку **Создать новый полный образ (Create a new full image)**. Более подробную информацию об остальных параметрах можно найти в R-Drive Image online Справке - раздел Резервные Комплекты (Backup Sets).

6. Задайте необходимые параметры на этапе *Параметры Образа (Image Options)* как показано на нижеприведенном рисунке.

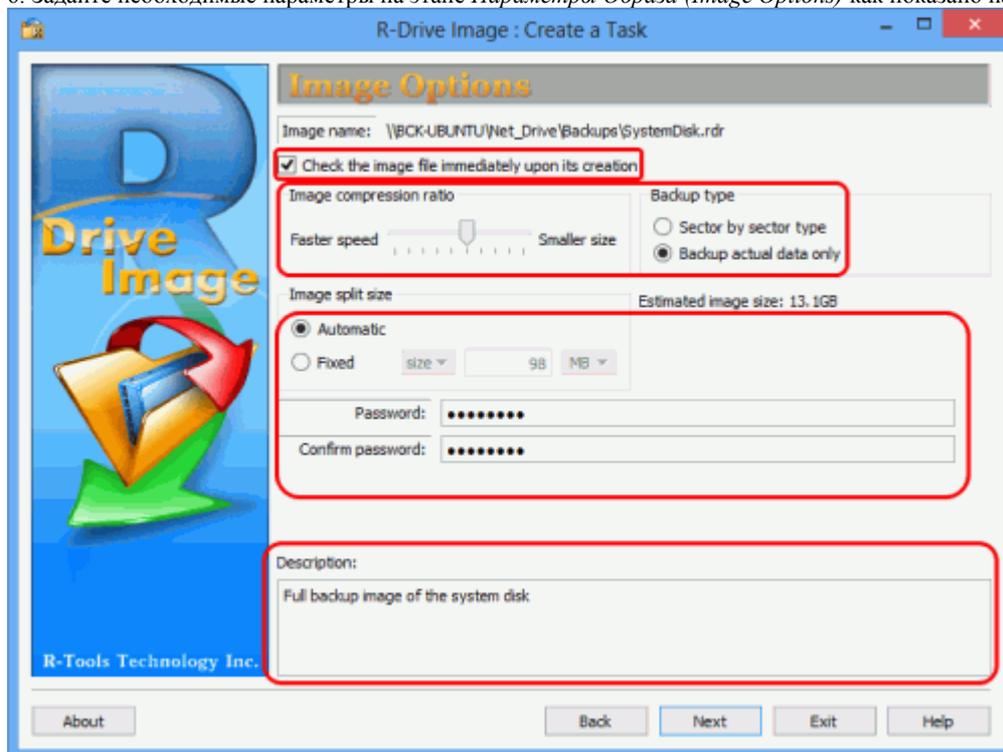


Рис. 7. Ежемесячный полный образ системного диска - этап Параметры Образа (Image Options)

Кликните по изображению для его увеличения

Более подробную информацию об остальных параметрах можно найти в R-Drive Image online Справке - раздел [Создание Образа \(Create an Image\)](#).

7. Задайте необходимые параметры на этапе *Параметры Резервного Копирования (Backup Options)* как показано на нижеприведенном рисунке.

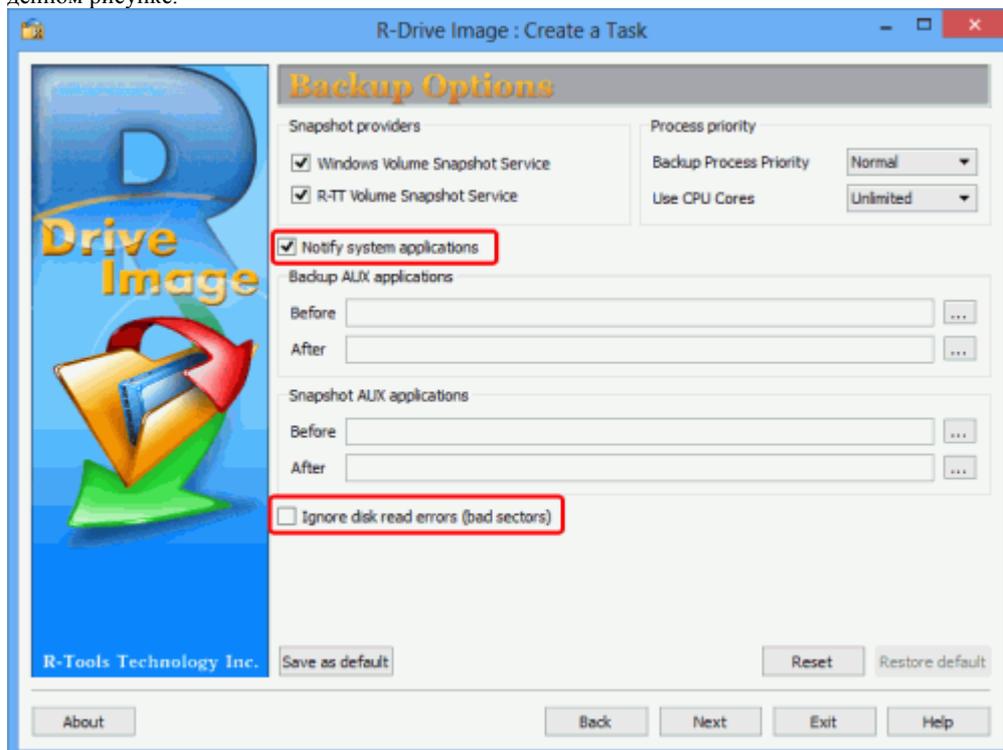


Рис. 8. Ежемесячный полный образ системного диска - этап Параметры Резервного Копирования (Backup Options)

Кликните по изображению для его увеличения

Более подробную информацию об остальных параметрах можно найти в R-Drive Image online Справке - раздел [Создание Образа \(Create an Image\)](#).

8. Задайте необходимые параметры на этапе *Время/Событие (Time/Event)* как показано на нижеприведенном рисунке.

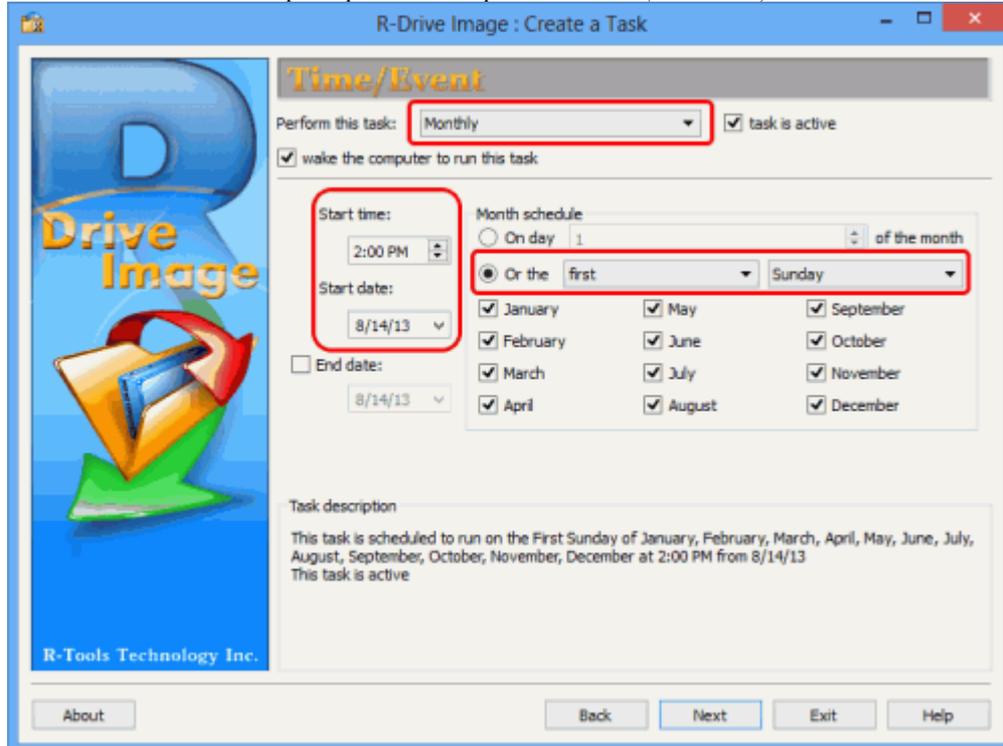


Рис. 9. Ежемесячный полный образ системного диска - этап *Время/Событие (Time/Event)*
Кликните по изображению для его увеличения

9. Задайте необходимые параметры на этапе *Пользователь/Пароль (User/Password)*.

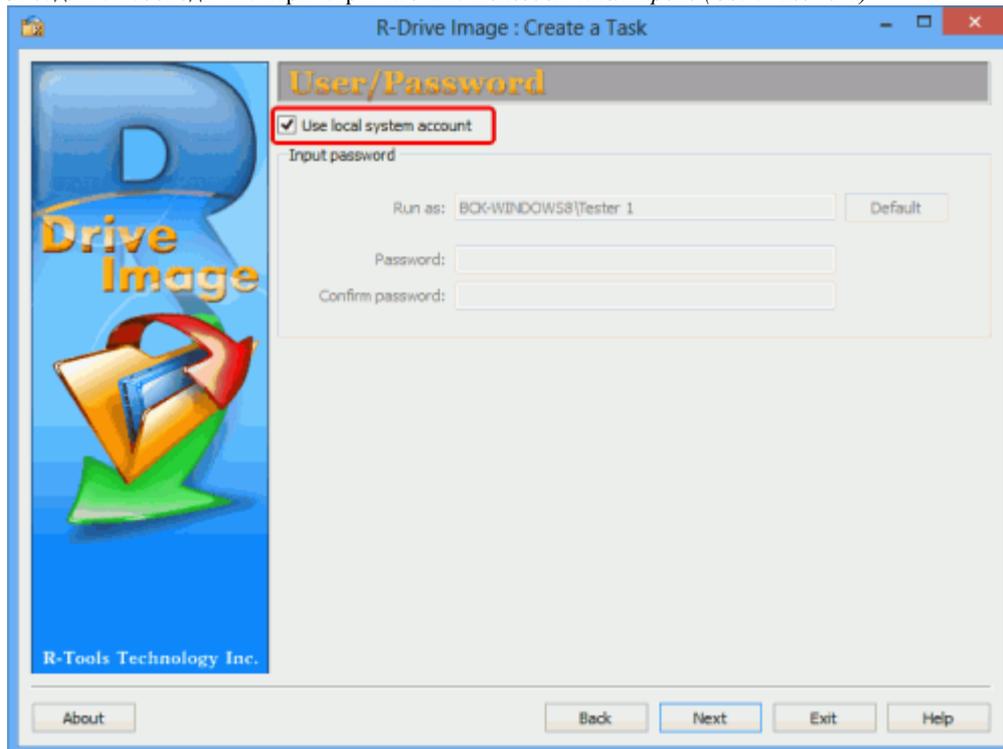


Рис. 10. Ежемесячный полный образ системного диска - этап *Пользователь/Пароль (User/Password)*
Кликните по изображению для его увеличения

10. Задайте параметры E-mail уведомления на этапе *E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)* как показано на нижеприведенном рисунке.

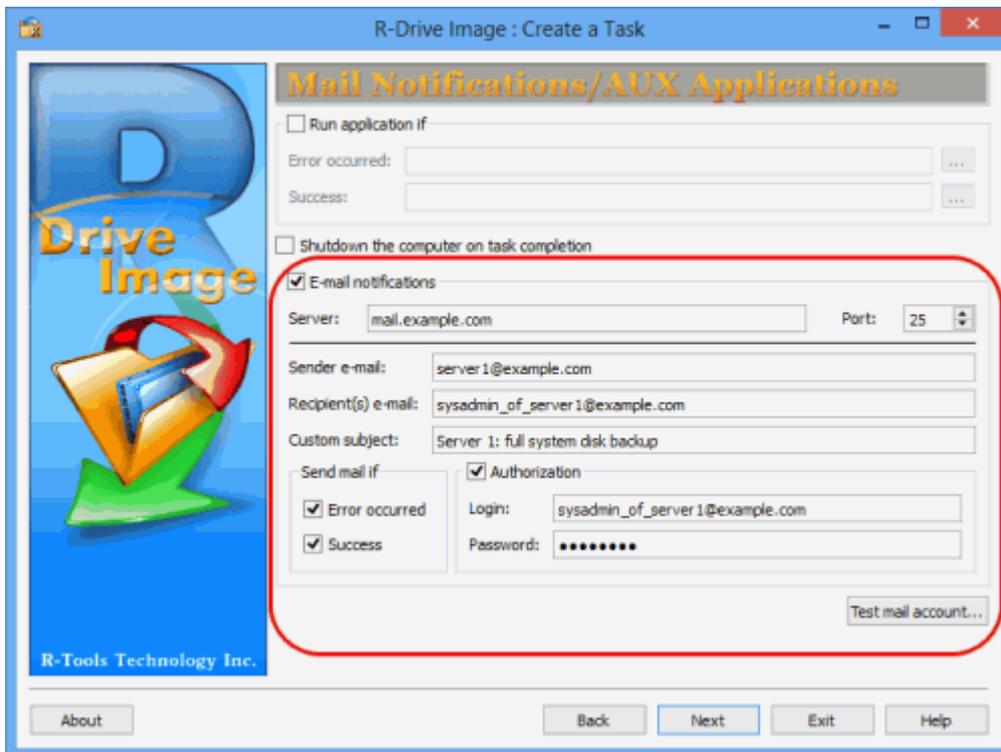


Рис. 11. Ежемесячный полный образ системного диска - этап E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)

Кликните по изображению для его увеличения

На этом же этапе можно проверить настройки e-mail нажав кнопку **Проверка E-mail... (Test mail account...)**. R-Drive Image отправит тестовое e-mail сообщение на указанный в настройках адрес.

11. Подтвердите корректность параметров задачи на этапе **Обработка (Processing)** и нажмите кнопку **Сохранить (Save)**.

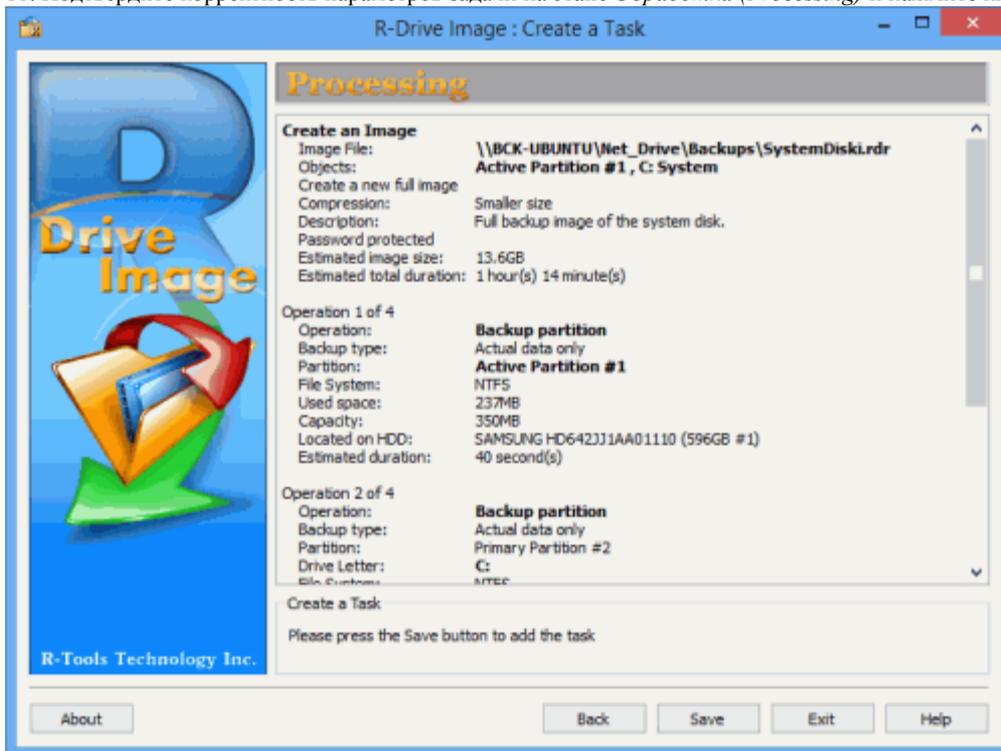


Рис. 12. Ежемесячный полный образ системного диска - этап Обработка (Processing)

Кликните по изображению для его увеличения

Чтобы изменить какой-либо параметр задачи нажмите кнопку **Назад (Back)** и вернитесь на соответствующий этап.

Если вы нажмете кнопку **Сохранить (Save)**, то созданная задача появится в списке на этапе **Расписание выполнения Задач (Scheduled Tasks)**.

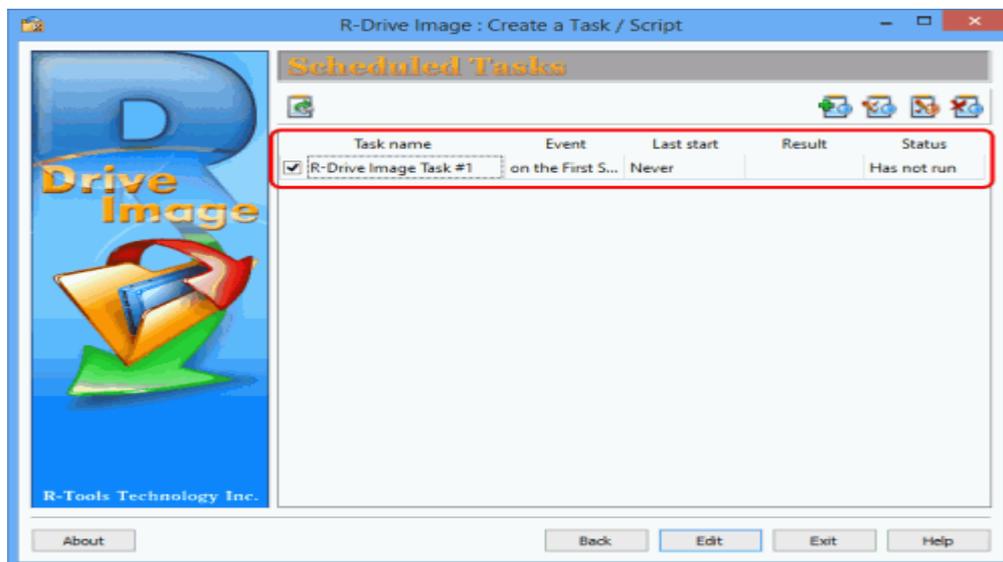


Рис. 13. Ежемесячный полный образ системного диска - этап Расписание выполнения Задач (Scheduled Tasks)

Кликните по изображению для его увеличения

Еженедельный образ системного диска в дифференциальном режиме

Следующий этап это создание еженедельного образа в дифференциальном режиме. В состав резервной копии войдут только измененные или добавленные данные с момента последнего создания полного образа. Размеры файлов образов в этом случае будут меньше и на их создание потребуется меньше времени, что является более удобным при выполнении еженедельных операций резервного копирования.

1. Нажмите кнопку **Создать задачу (Create a task)** на этапе *Расписание выполнения Задач (Scheduled Tasks)*, выберите системный раздел на этапе *Выбор Раздела (Partition Selection)* и выберите на этапе *Месторасположение Образа (Image Destination)* тот же самый файл образ что и при создании полного образа.

2. Задайте параметры резервных комплектов на этапе *Режим Создания Образа (Imaging Mode)*.

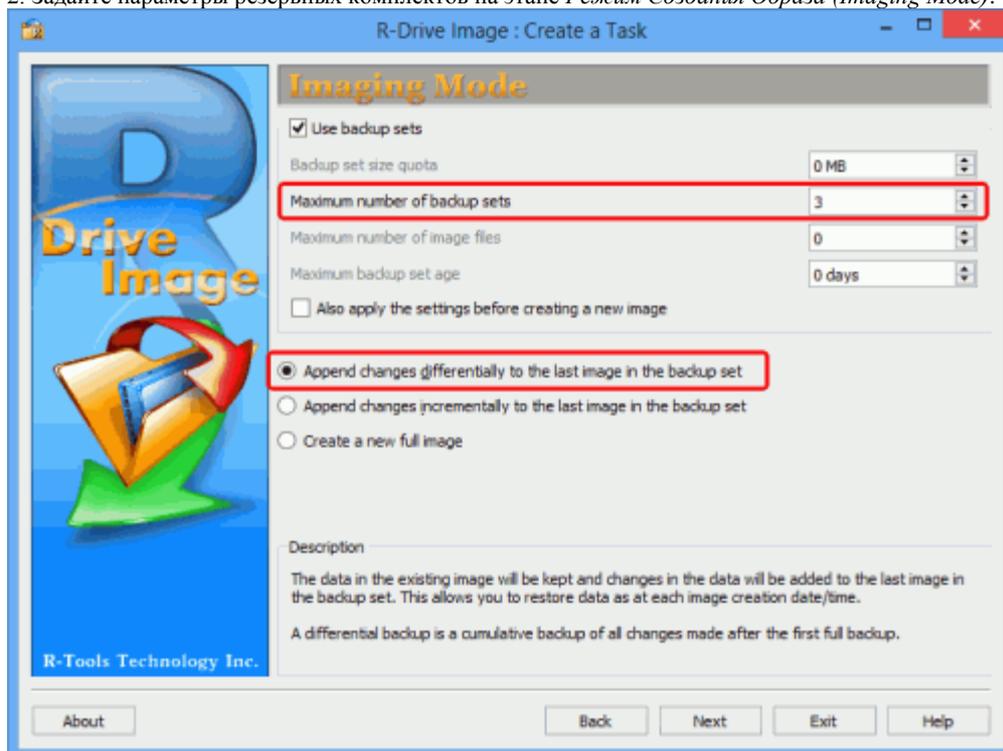


Рис. 14. Еженедельный образ системного диска в дифференциальном режиме - этап Режим Создания Образа (Imaging Mode)

Кликните по изображению для его увеличения

Введите **3** в поле *Максимальное число резервных комплектов (Maximum number of backup sets)* и установите радиокнопку **Добавлять изменения дифференциально к последнему образу в резервном комплекте (Append changes differentially to the last image in the backup set)**.

Если созданный полный образ диска был защищен паролем, то вам потребуется его ввести.

3. Задайте необходимые параметры на этапах *Параметры Образа (Image Options)* и *Параметры Резервного Копирования (Backup Options)* также как и при создании полного образа.

4. Задайте необходимые параметры на этапе *Время/Событие (Time/Event)*.

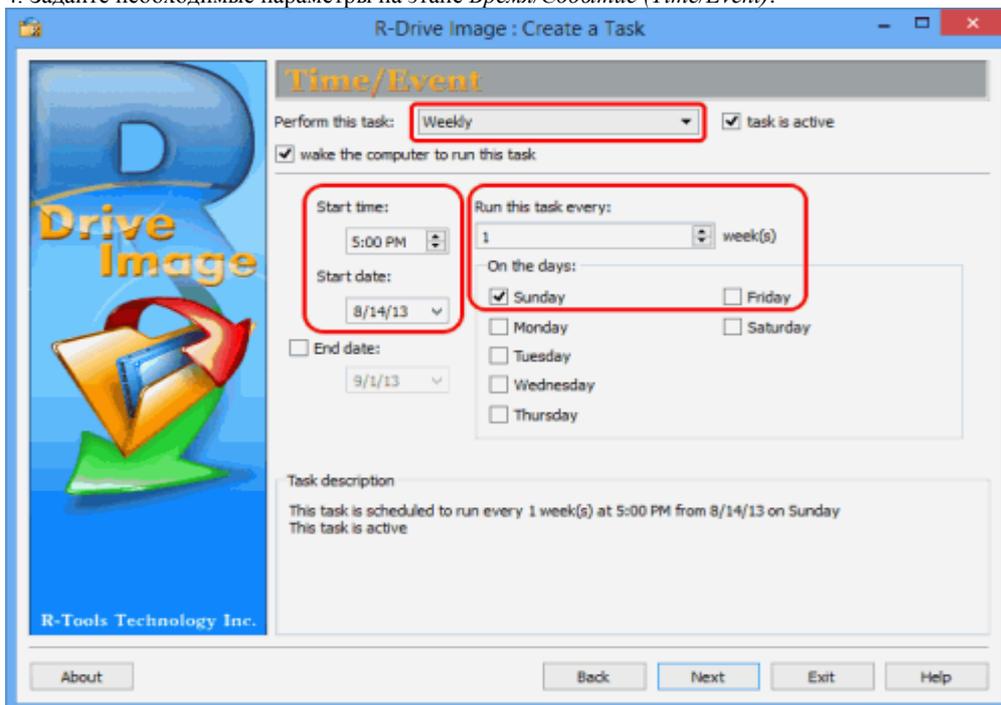


Рис. 15. Ежедневный образ системного диска в дифференциальном режиме - этап *Время/Событие (Time/Event)*

Кликните по изображению для его увеличения

Проверьте чтобы *Время начала (Start time)* не совпадало со временем создания полного образа. В нашем примере ежемесячный полный образ создается в 14:00 и еженедельный образ в дифференциальном режиме в 17:00.

5. Задайте необходимые параметры на этапе *Пользователь/Пароль (User/Password)*.

6. Задайте необходимые параметры на этапе *E-mail Уведомления/Внешние Утилиты (Mail Notifications/AUX Applications)* также как и при создании полного образа.

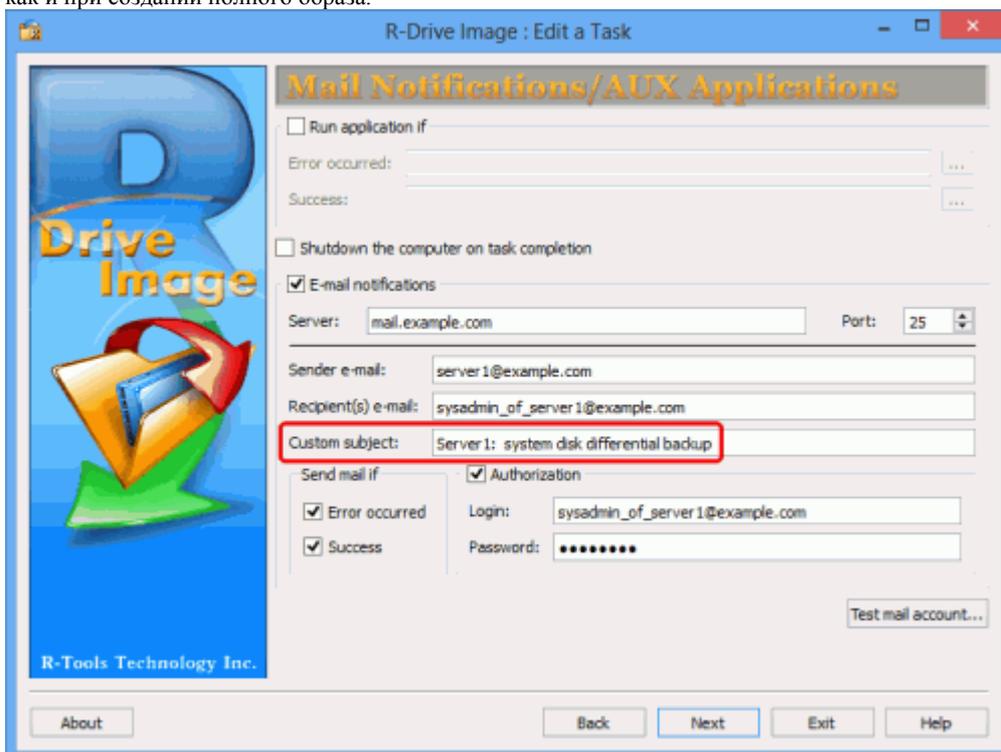


Рис. 16. Ежедневный образ системного диска в дифференциальном режиме - этап *E-mail Уведомления/Внешние Утилиты (Mail Notifications/AUX Applications)*

Кликните по изображению для его увеличения

Измените *Тему уведомления: (Custom subject:)* на **Server1: differential system disk backup**.

7. Подтвердите корректность параметров задачи на этапе *Обработка (Processing)* и нажмите кнопку **Сохранить (Save)**.

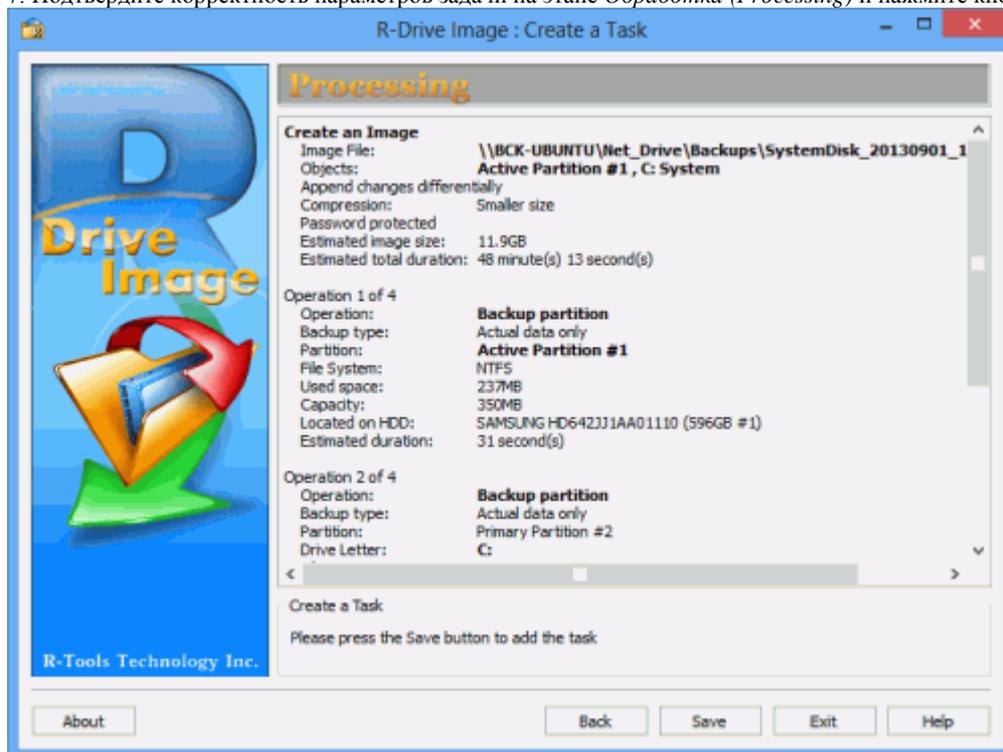


Рис. 17. Еженедельный образ системного диска в дифференциальном режиме - этап **Обработка (Processing)**
Кликните по изображению для его увеличения

Созданная задача появится в списке на этапе *Расписание выполнения Задач (Scheduled Tasks)*.

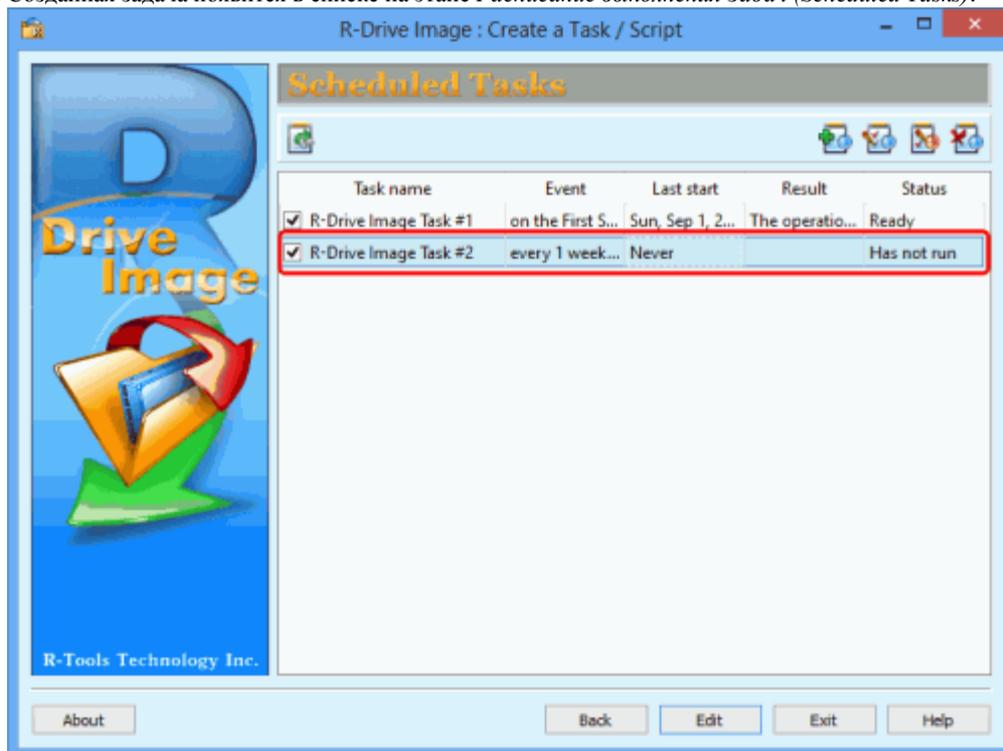


Рис. 18. Еженедельный образ системного диска в дифференциальном режиме - этап **Расписание выполнения Задач (Scheduled Tasks)**

Кликните по изображению для его увеличения

Копирование Диска с Данными

Ежемесячный полный образ диска с данными

Данный процесс во многом схож с копированием системного диска за исключением того, что в этом случае создается образ раздела с данными (D:).

1. На этапе *Расписание выполнения Задач (Scheduled Tasks)* нажмите кнопку **Создать задачу (Create a task)**.

2. Выберите раздел с данными (D:) на этапе *Выбор Раздела (Partition Selection)*.

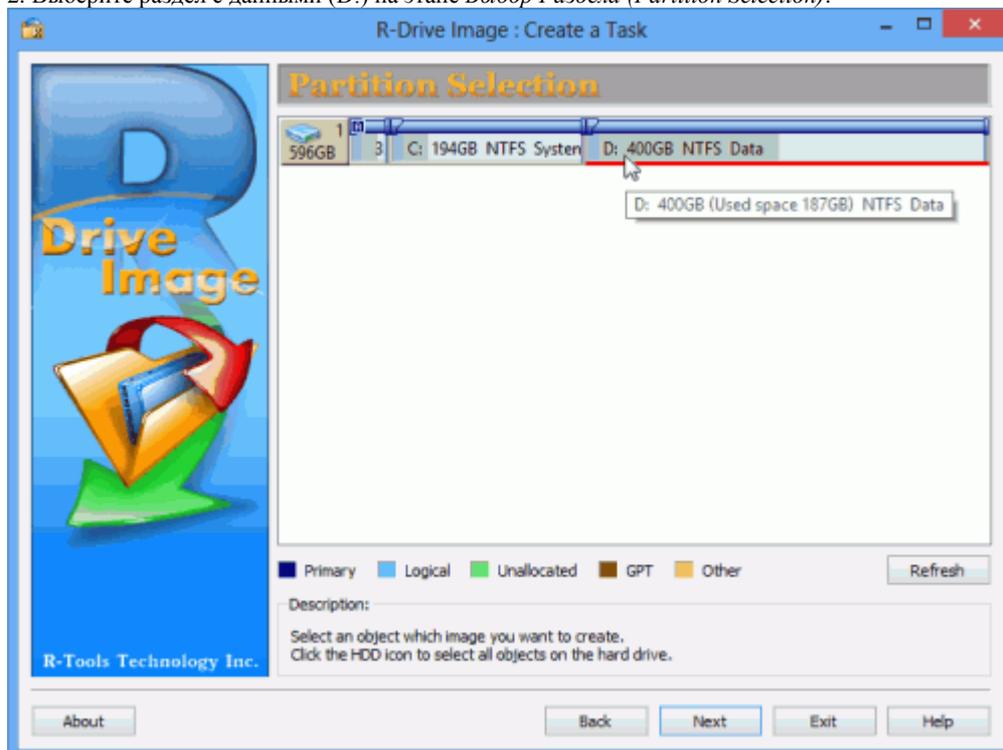


Рис. 19. Ежемесячный полный образ диска с данными - этап **Выбор Раздела (Partition Selection)**

Кликните по изображению для его увеличения

3. На этапе *Месторасположение Образа (Image Destination)* выберите месторасположение файла образа и имя файла.

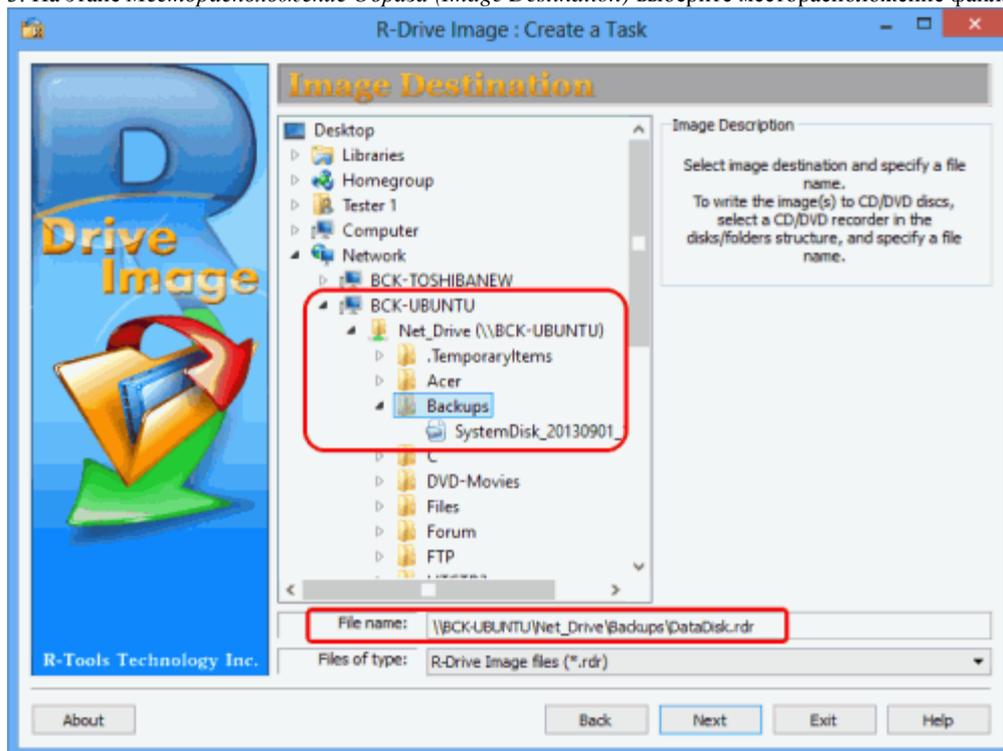


Рис. 20. Ежемесячный полный образ диска с данными - этап **Месторасположение Образа (Image Destination)**

Кликните по изображению для его увеличения

4. Задайте параметры резервных комплектов на этапе *Режим Создания Образа (Imaging Mode)*.

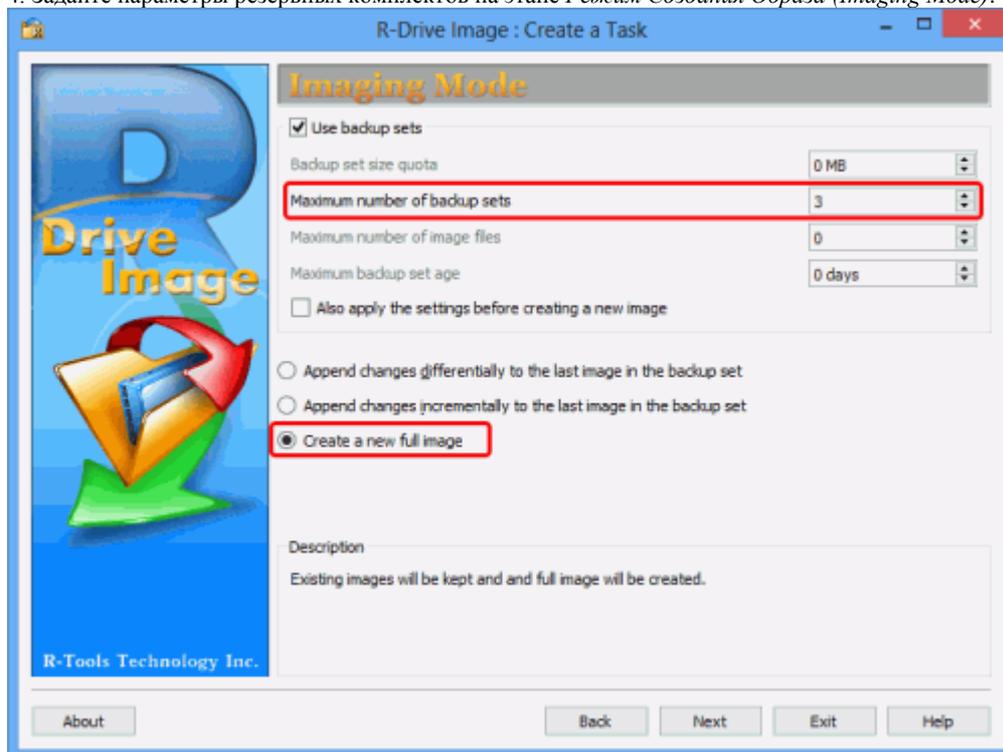


Рис. 21. Ежемесячный полный образ диска с данными - этап Режим Создания Образа (Imaging Mode)

Кликните по изображению для его увеличения

Установите 3 в поле *Максимальное число резервных комплектов (Maximum number of backup sets)* и установите радиокнопку *Создать новый полный образ (Create a new full image)*.

5. Задайте необходимые параметры на этапах *Параметры Образа (Image Options)* и *Параметры Резервного Копирования (Backup Options)* также как и при создании полного образа системного диска.

6. Задайте необходимые параметры на этапе *Время/Событие (Time/Event)*.

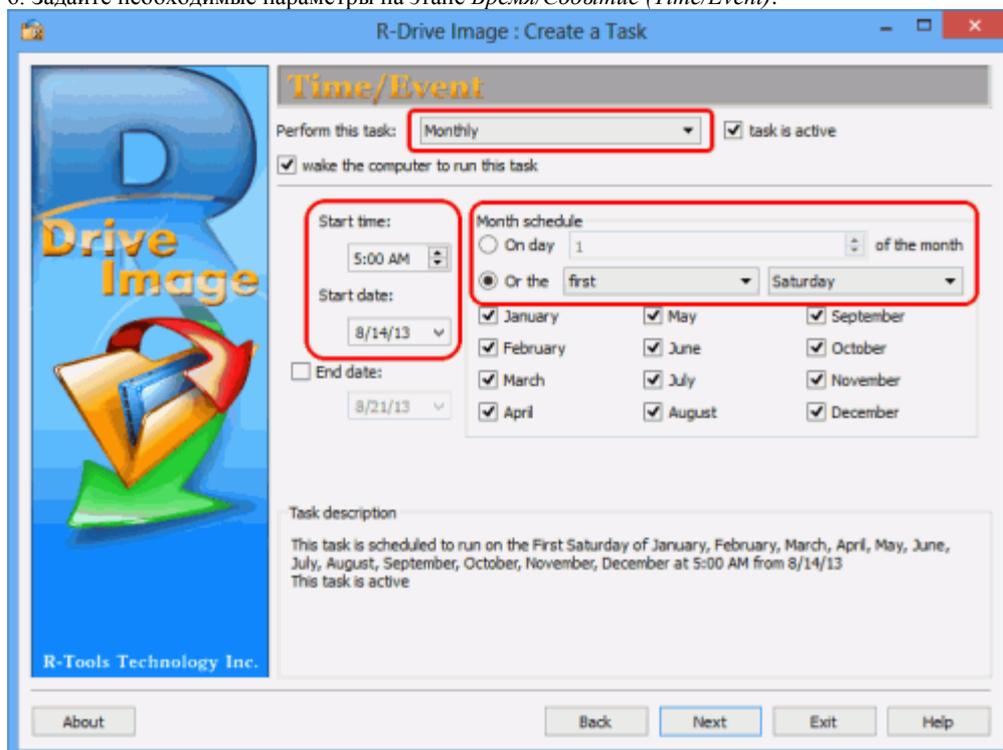


Рис. 22. Ежемесячный полный образ диска с данными - этап Время/Событие (Time/Event)

Кликните по изображению для его увеличения

7. Задайте необходимые параметры на этапе *Пользователь/Пароль (User/Password)*.

8. Задайте параметры E-mail уведомления на этапе *E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)*.

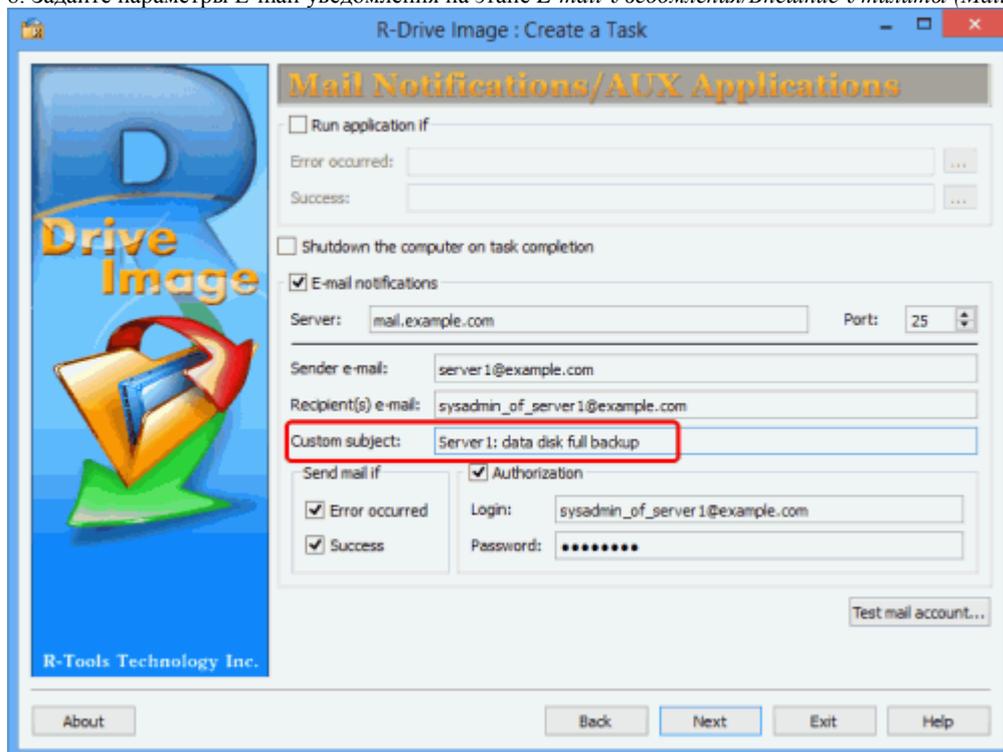


Рис. 23. Ежемесячный полный образ диска с данными - этап E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)

Кликните по изображению для его увеличения

9. Подтвердите корректность параметров задачи на этапе *Обработка (Processing)* и нажмите кнопку **Сохранить (Save)**.

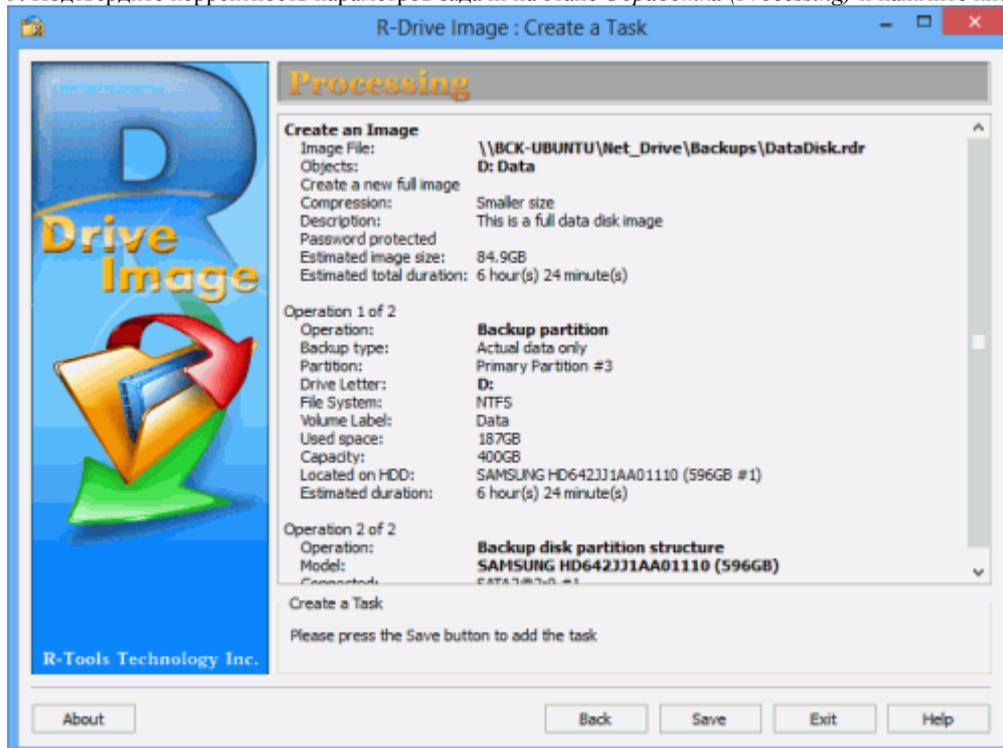


Рис. 24. Ежемесячный полный образ диска с данными - этап Обработка (Processing)

Кликните по изображению для его увеличения

Созданная задача появится в списке на этапе *Расписание выполнения Задач (Scheduled Tasks)*.

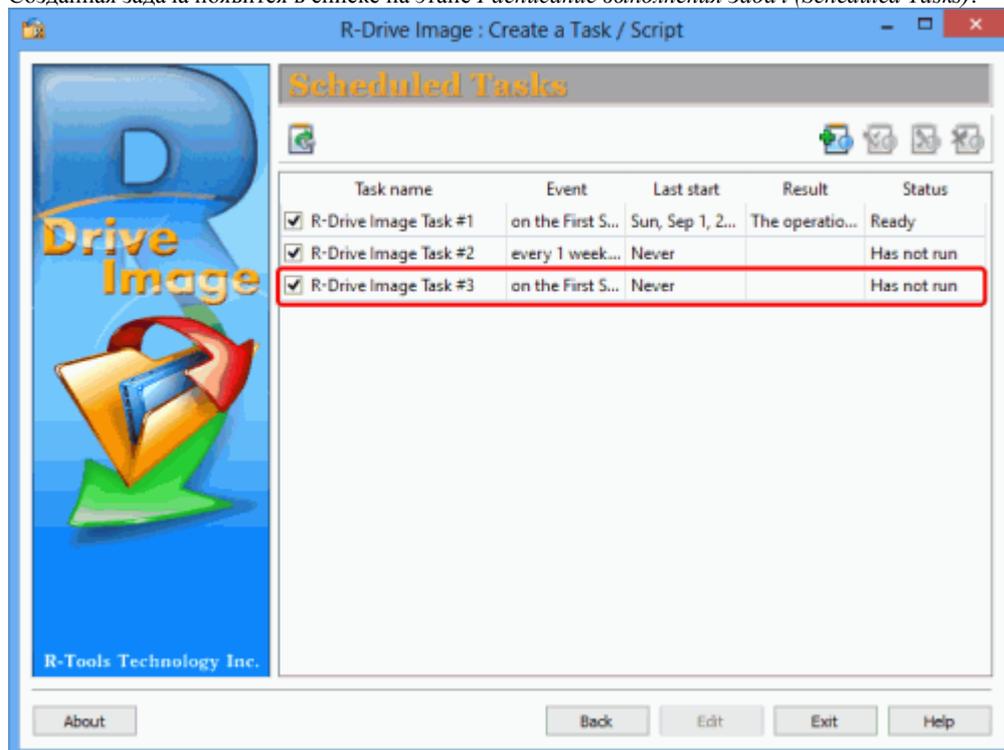


Рис. 25. Ежемесячный полный образ диска с данными - этап *Расписание выполнения Задач (Scheduled Tasks)*

Кликните по изображению для его увеличения

Еженедельный образ диска с данными в дифференциальном режиме

Данный процесс также во многом схож с созданием образа системного диска в дифференциальном режиме за исключением того, что в этом случае создается образ раздела с данными (D:).

1. На этапе *Расписание выполнения Задач (Scheduled Tasks)* нажмите кнопку **Создать задачу (Create a task)**, выберите раздел с данными (т.е. D:) на этапе *Выбор Раздела (Partition Selection)* и выберите месторасположение файла образа и имя файла (то же файл образ что и при создании полного образа диска с данными) на этапе *Месторасположение Образа (Image Destination)*.
2. Задайте параметры резервных комплектов на этапе *Режим Создания Образа (Imaging Mode)*.

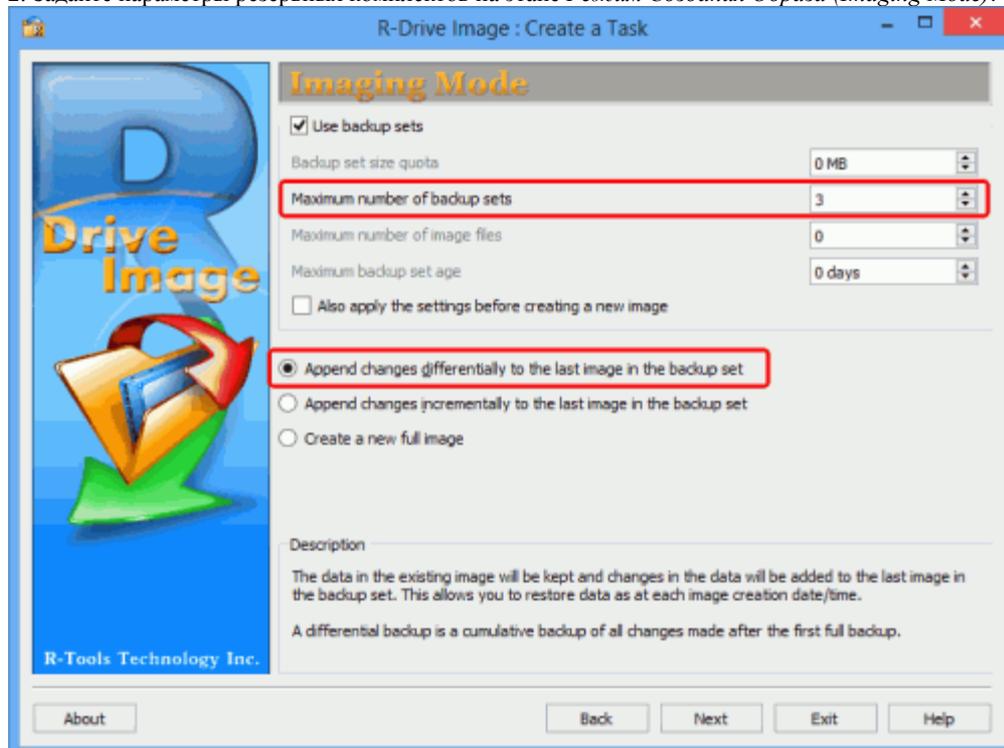


Рис. 26. Еженедельный образ диска с данными в дифференциальном режиме - этап *Режим Создания Образа (Imaging Mode)*

Кликните по изображению для его увеличения

Введите **3** в поле *Максимальное число резервных комплектов (Maximum number of backup sets)* и установите радиокнопку **Добавлять изменения дифференциально к последнему образу в резервном комплекте (Append changes differentially to the last image in the backup set)**.

3. Задайте необходимые параметры на этапах *Параметры Образа (Image Options)* и *Параметры Резервного Копирования* также как и при создании полного образа диска с данными.

4. Задайте необходимые параметры на этапе *Время/Событие (Time/Event)*.

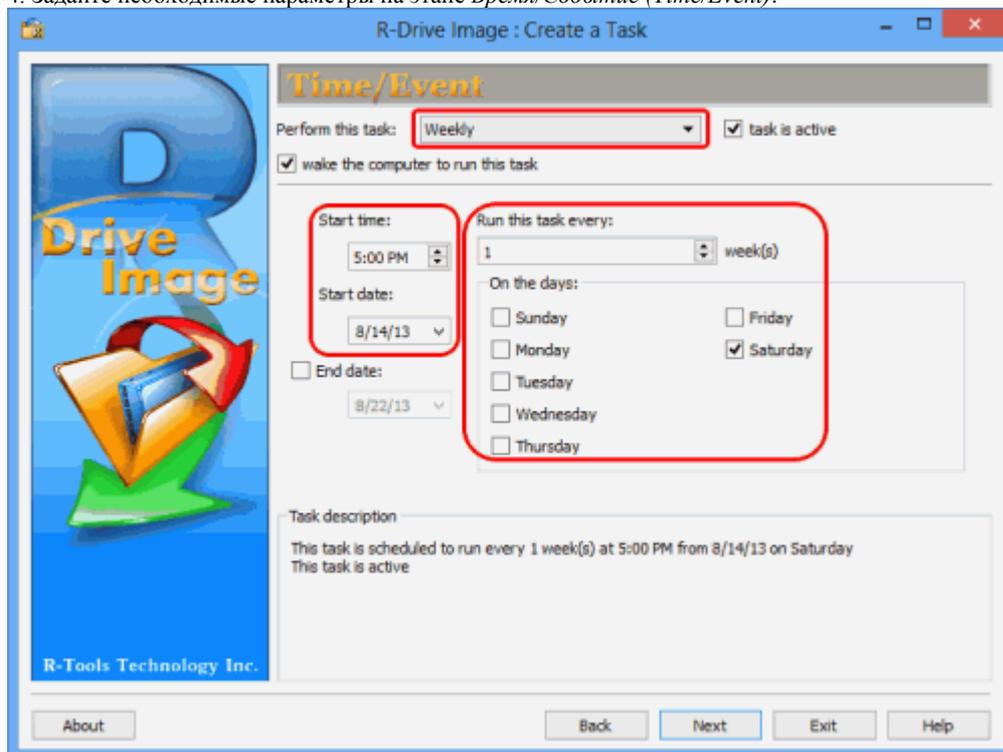


Рис. 27. Ежедневный образ диска с данными в дифференциальном режиме - этап *Время/Событие (Time/Event)*
Кликните по изображению для его увеличения

5. Задайте необходимые параметры на этапе *Пользователь/Пароль (User/Password)*.

6. Задайте параметры E-mail уведомления на этапе *E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)*.

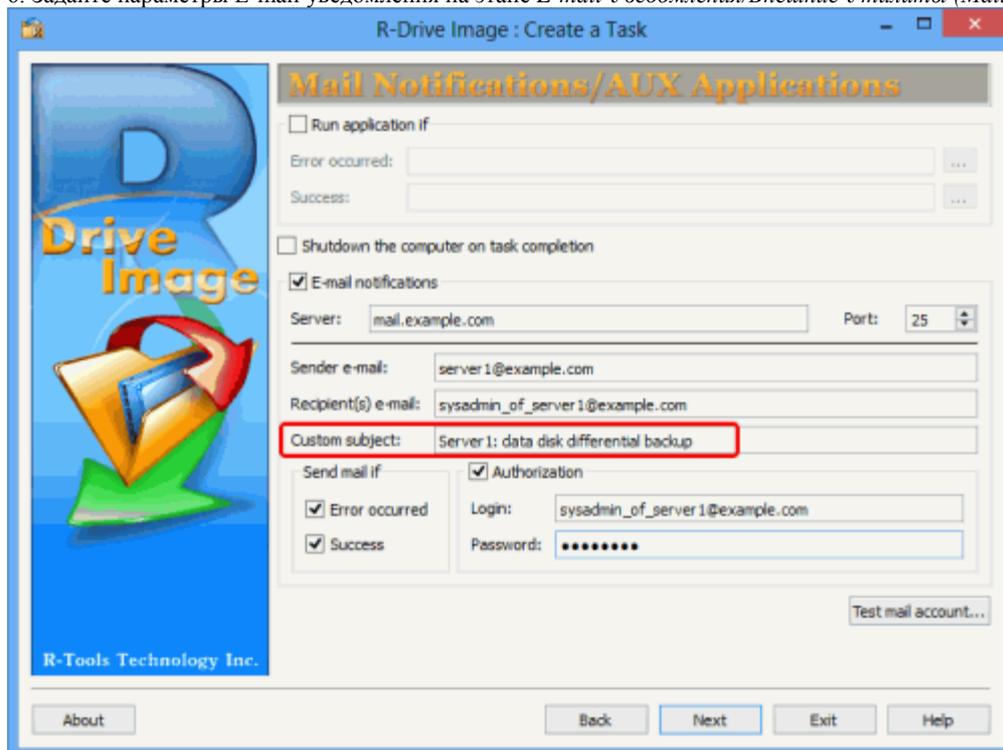


Рис. 28. Ежедневный образ диска с данными в дифференциальном режиме - этап *E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)*
Кликните по изображению для его увеличения

7. Подтвердите корректность параметров задачи на этапе *Обработка (Processing)* и нажмите кнопку **Сохранить (Save)**.



Рис. 29. Ежедневный образ диска с данными в дифференциальном режиме - этап **Обработка (Processing)**
 Кликните по изображению для его увеличения

Созданная задача появится в списке на этапе *Расписание выполнения Задач (Scheduled Tasks)*.

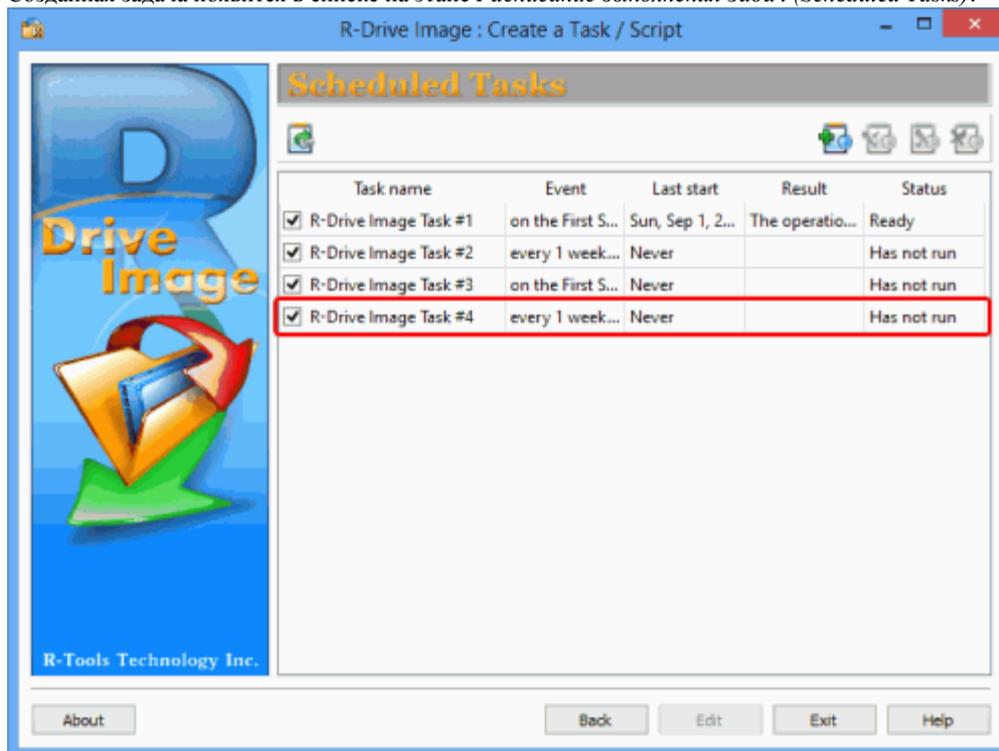


Рис. 30. Ежедневный образ диска с данными в дифференциальном режиме - этап **Расписание выполнения Задач (Scheduled Tasks)**

Кликните по изображению для его увеличения

Ежедневный образ диска с данными в инкрементальном режиме

Образ в инкрементальном режиме содержит измененные или добавленные данные с момента последнего любого резервного копирования (полного, дифференциального или инкрементального). Размеры файлов образов в этом случае будут меньше размеров файлов образов созданных в дифференциальном режиме, что является более удобным при выполнении ежедневных операций резервного копирования.

1. Нажмите кнопку **Создать задачу (Create a task)** на этапе *Расписание выполнения Задач (Scheduled Tasks)*, выберите раздел данными на этапе *Выбор Раздела (Partition Selection)* и выберите на этапе *Месторасположение Образа (Image Destination)* тот же самый файл образ что и при создании полного образа диска с данными.

2. Задайте параметры резервных комплектов на этапе *Режим Создания Образа (Imaging Mode)*.

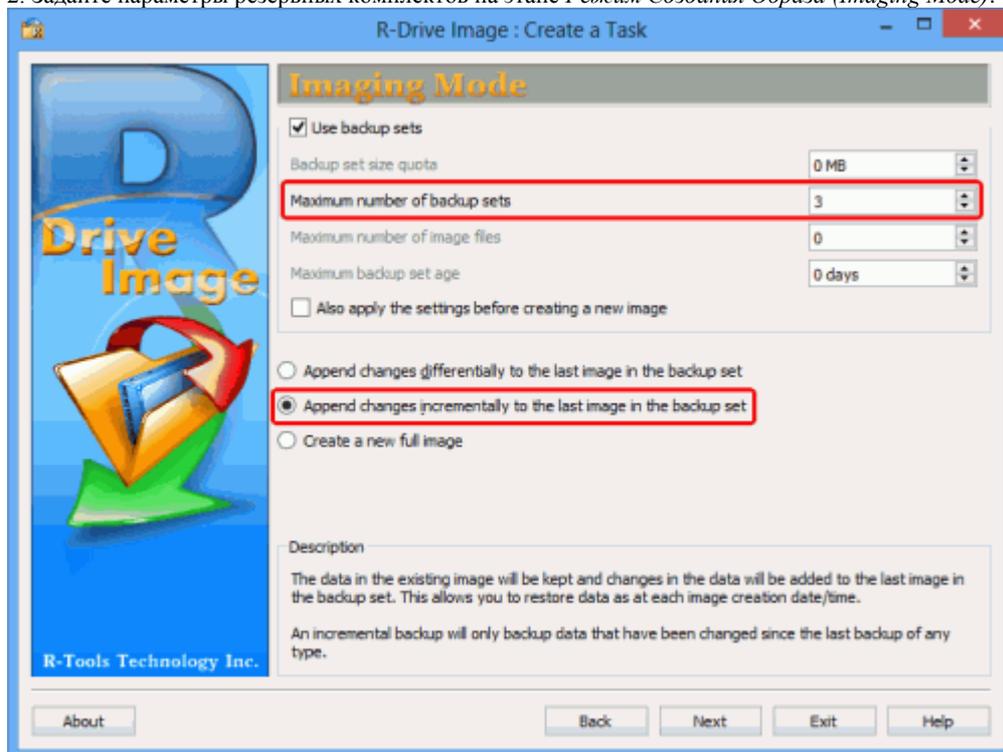


Рис. 31. Ежедневный образ диска с данными в инкрементальном режиме - этап Режим Создания Образа (Imaging Mode)
Кликните по изображению для его увеличения

Введите **3** в поле *Максимальное число резервных комплектов (Maximum number of backup sets)* и установите радиокнопку *Добавлять изменения инкрементально к последнему образу в резервном комплекте (Append changes incrementally to the last image in the backup set)*.

3. Задайте необходимые параметры на этапах *Параметры Образа (Image Options)* и *Параметры Резервного Копирования (Backup Options)* также как и при создании полного образа диска с данными.

4. Задайте необходимые параметры на этапе *Время/Событие (Time/Event)*.

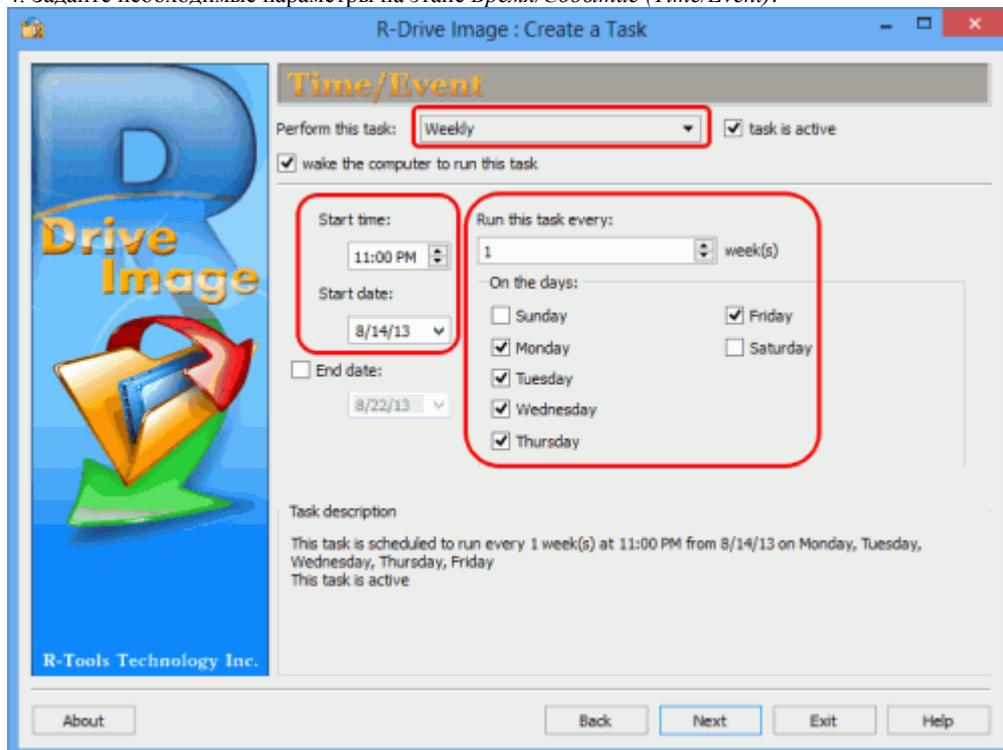


Рис. 32. Ежедневный образ диска с данными в инкрементальном режиме - этап Время/Событие (Time/Event)
Кликните по изображению для его увеличения

5. Задайте необходимые параметры на этапе *Пользователь/Пароль (User/Password)*.

6. Задайте параметры E-mail уведомления на этапе *E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)*

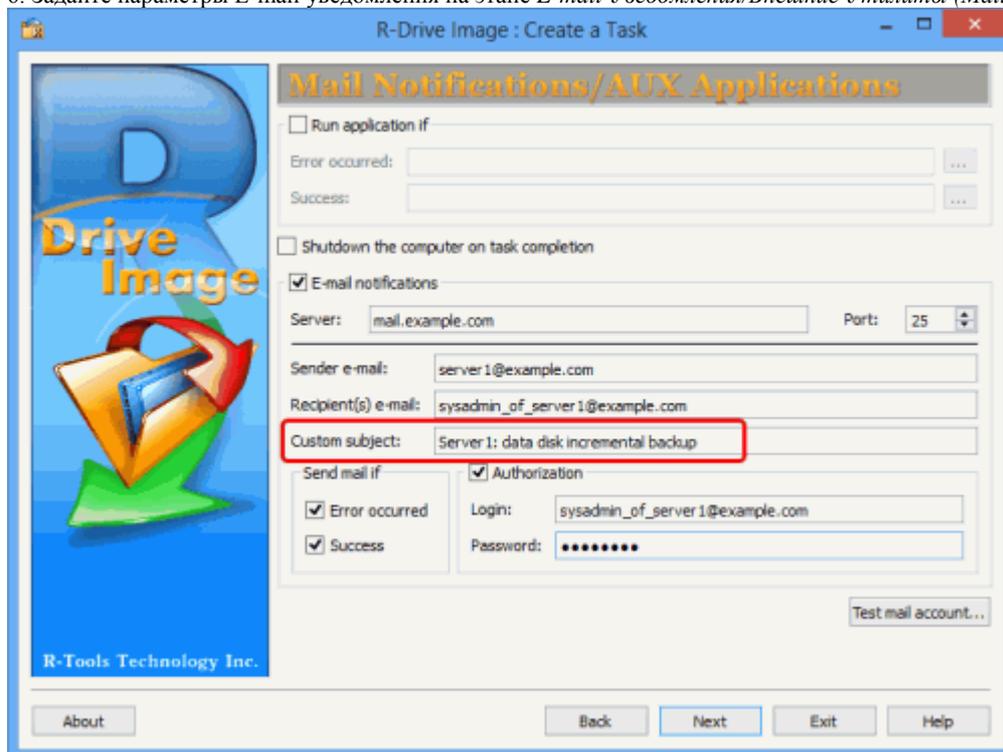


Рис. 33. Ежедневный образ диска с данными в инкрементальном режиме - этап E-mail Уведомления/Внешние Утилиты (Mail Notification/AUX Applications)

Кликните по изображению для его увеличения

7. Подтвердите корректность параметров задачи на этапе *Обработка (Processing)* и нажмите кнопку **Сохранить (Save)**.

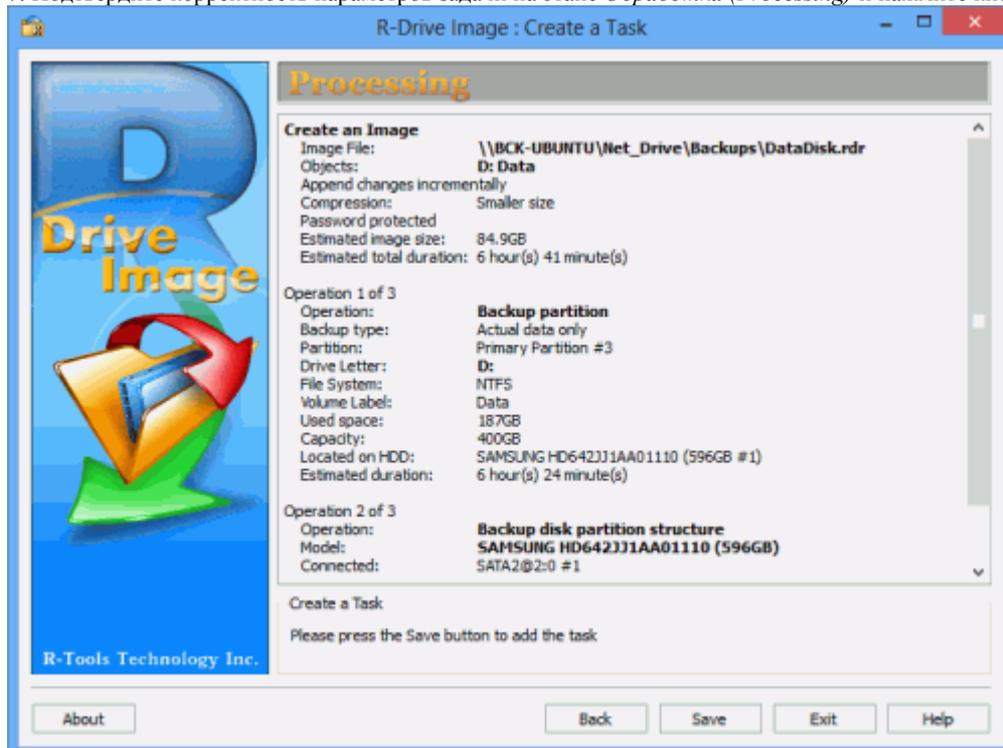


Рис. 34. Ежедневный образ диска с данными в инкрементальном режиме - этап Обработка (Processing)

Кликните по изображению для его увеличения

Созданная задача появится в списке на этапе *Расписание выполнения Задач (Scheduled Tasks)*.

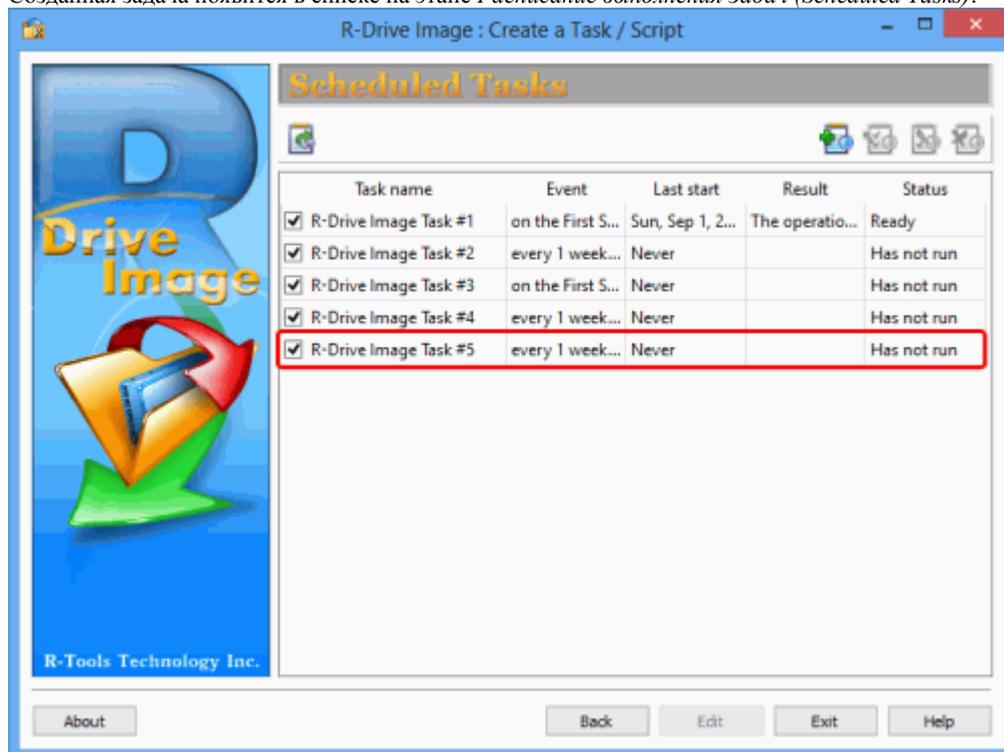


Рис. 35. Ежедневный образ диска с данными в инкрементальном режиме - этап Расписание выполнения Задач (Scheduled Tasks)

Кликните по изображению для его увеличения

Структура Файлов Плана Резервного Копирования

Теперь план резервного копирования готов. Вы начнете получать уведомления о завершении операций резервного копирования. Описание уведомления (примеры) вы найдете далее в этой статье.

Созданные R-Drive Image файлы образы будут иметь следующие имена:

- **Полный образ:** <имя_файла>_<дата_первого_образа>_<время_первого_образа>_1.rdr

- **Образы в инкрементальных и дифференциальных режимах:** <имя_файла>_<дата_первого_образа>_<время_первого_образа>_N+1.rdr
где N это число ранее созданных инкрементальных или дифференциальных образов.

В описанном выше примере R-Drive Image создаст следующие файлы:

Образ системного диска

Дата/День недели/Время	Файлы Новые создаваемые файлы выделены жирным	Тип образа	Номер Резервного Комплекта
<i>Начало Резервного Комплекта 1</i>			
2013-09-01 / Воскресенье / 14:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	
2013-09-01 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
2013-09-08 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	
	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_3.rdr	Дифф. образ	
2013-09-15 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	
	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_3.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_4.rdr	Дифф. образ	
2013-09-22 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	Резервный Комплект 1
	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_3.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_4.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_5.rdr	Дифф. образ	
2013-09-29 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	
	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_3.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_4.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_5.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_6.rdr	Дифф. образ	
<i>Начало Резервного Комплекта 2</i>			
2013-10-06 / Воскресенье / 14:00	SystemDisk_20130901_020000PM_1.rdr	Полный образ	Резервные Комплекты 1,2
	SystemDisk_20130901_020000PM_2.rdr	Дифф. образ	
2013-10-06 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_3.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_4.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_5.rdr	Дифф. образ	
	SystemDisk_20130901_020000PM_6.rdr	Дифф. образ	
	SystemDisk_20131006_020000PM_1.rdr	Полный образ	
	SystemDisk_20131006_020000PM_2.rdr	Дифф. образ	

2013-11-24 / Воскресенье / 17:00	SystemDisk_20130901_020000PM_1.rdr SystemDisk_20130901_020000PM_2.rdr SystemDisk_20130901_020000PM_3.rdr SystemDisk_20130901_020000PM_4.rdr SystemDisk_20130901_020000PM_5.rdr SystemDisk_20130901_020000PM_6.rdr SystemDisk_20131006_020000PM_1.rdr SystemDisk_20131006_020000PM_2.rdr SystemDisk_20131006_020000PM_3.rdr SystemDisk_20131006_020000PM_4.rdr SystemDisk_20131006_020000PM_5.rdr SystemDisk_20131103_020000PM_1.rdr SystemDisk_20131103_020000PM_2.rdr SystemDisk_20131103_020000PM_3.rdr SystemDisk_20131103_020000PM_4.rdr SystemDisk_20131103_020000PM_5.rdr	Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ	
<i>Начало Резервного Комплекта 4. Резервный Комплект 1 удален.</i>			
2013-12-01 / Воскресенье / 17:00	SystemDisk_20131006_020000PM_1.rdr SystemDisk_20131006_020000PM_2.rdr SystemDisk_20131006_020000PM_3.rdr SystemDisk_20131006_020000PM_4.rdr SystemDisk_20131006_020000PM_5.rdr SystemDisk_20131103_020000PM_1.rdr SystemDisk_20131103_020000PM_2.rdr SystemDisk_20131103_020000PM_3.rdr SystemDisk_20131103_020000PM_4.rdr SystemDisk_20131103_020000PM_5.rdr SystemDisk_20131201_020000PM_1.rdr SystemDisk_20131201_020000PM_2.rdr	Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ	Резервные Комплекты 2,3,4
2013-12-08 / Воскресенье / 17:00	SystemDisk_20131006_020000PM_1.rdr SystemDisk_20131006_020000PM_2.rdr SystemDisk_20131006_020000PM_3.rdr SystemDisk_20131006_020000PM_4.rdr SystemDisk_20131006_020000PM_5.rdr SystemDisk_20131103_020000PM_1.rdr SystemDisk_20131103_020000PM_2.rdr SystemDisk_20131103_020000PM_3.rdr SystemDisk_20131103_020000PM_4.rdr SystemDisk_20131103_020000PM_5.rdr SystemDisk_20131201_020000PM_1.rdr SystemDisk_20131201_020000PM_2.rdr SystemDisk_20131201_020000PM_3.rdr	Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ Дифф. образ Дифф. образ Дифф. образ Полный образ Дифф. образ Дифф. образ	

Образ диска с данными

Ниже приведен пример структуры файлов одного из резервных комплектов. Показано начало Резервного Комплекта 2.

Дата/День недели/Время	Файл	Тип образа
2013-08-31 / Суббота / 5:00	DataDisk_20130831_050000AM_1.rdr	Полный
2013-08-31 / Суббота / 15:00	DataDisk_20130831_050000PM_2.rdr	Дифф.
2013-09-02 / Понедельник / 23:00	DataDisk_20130831_050000AM_3.rdr	Инкр.
2013-09-03 / Вторник / 23:00	DataDisk_20130831_050000AM_4.rdr	Инкр.
2013-09-04 / Среда / 23:00	DataDisk_20130831_050000AM_5.rdr	Инкр.
2013-09-05 / Четверг / 23:00	DataDisk_20130831_050000AM_6.rdr	Инкр.
2013-09-06 / Пятница / 23:00	DataDisk_20130831_050000AM_7.rdr	Инкр.
2013-09-07 / Суббота / 3:00PM	DataDisk_20130831_050000AM_8.rdr	Дифф.
2013-09-09 / Понедельник / 23:00	DataDisk_20130831_050000AM_9.rdr	Инкр.
2013-09-10 / Вторник / 23:00	DataDisk_20130831_050000AM_10.rdr	Инкр.
2013-09-11 / Среда / 23:00	DataDisk_20130831_050000AM_11.rdr	Инкр.
2013-09-12 / Четверг / 23:00	DataDisk_20130831_050000AM_12.rdr	Инкр.
2013-09-13 / Пятница / 23:00	DataDisk_20130831_050000AM_13.rdr	Инкр.
2013-09-14 / Суббота / 15:00	DataDisk_20130831_050000AM_14.rdr	Дифф.
2013-09-16 / Понедельник / 23:00	DataDisk_20130831_050000AM_15.rdr	Инкр.
2013-09-17 / Вторник / 23:00	DataDisk_20130831_050000AM_16.rdr	Инкр.
2013-09-18 / Среда / 23:00	DataDisk_20130831_050000AM_17.rdr	Инкр.
2013-09-19 / Четверг / 23:00	DataDisk_20130831_050000AM_18.rdr	Инкр.
2013-09-20 / Пятница / 23:00	DataDisk_20130831_050000AM_19.rdr	Инкр.
2013-09-21 / Суббота / 15:00	DataDisk_20130831_050000AM_20.rdr	Дифф.
2013-09-23 / Понедельник / 23:00	DataDisk_20130831_050000AM_21.rdr	Инкр.
2013-09-24 / Вторник / 23:00	DataDisk_20130831_050000AM_22.rdr	Инкр.
2013-09-25 / Среда / 23:00	DataDisk_20130831_050000AM_23.rdr	Инкр.
2013-09-26 / Четверг / 23:00	DataDisk_20130831_050000AM_24.rdr	Инкр.
2013-09-27 / Пятница / 23:00	DataDisk_20130831_050000AM_25.rdr	Инкр.
2013-09-28 / Суббота / 15:00	DataDisk_20130831_050000AM_26.rdr	Дифф.
2013-09-30 / Понедельник / 23:00	DataDisk_20130831_050000AM_27.rdr	Инкр.

2013-10-01 /Вторник /23:00	DataDisk_20130831_050000AM_28.rdr	Инкр.
2013-10-02 /Среда /23:00	DataDisk_20130831_050000AM_29.rdr	Инкр.
2013-10-03 /Четверг /23:00	DataDisk_20130831_050000AM_30.rdr	Инкр.
2013-10-04 /Пятница /23:00	DataDisk_20130831_050000AM_31.rdr	Инкр.
<i>Начало Резервного Комплекта 2</i>		
2013-10-05 /Суббота /5:00	DataDisk_20131005_050000AM_1.rdr	Полный
2013-10-05 /Суббота /15:00	DataDisk_20131005_050000AM_2.rdr	Дифф.

Вы можете удалять ежедневные инкрементальные образы созданные в период времени между созданием полных и дифференциальных образов не теряя при этом целостности данных. Например, после создания файла DataDisk_20130831_050000AM_8.rdr можно удалить файлы начиная с DataDisk_20130831_050000AM_3.rdr и заканчивая DataDisk_20130831_050000AM_7.rdr.

Протоколирование

Можно сохранить файл журнала операций R-Drive Image. Для этого нажмите кнопку **О программе (About)** на этапе *Выбор Действия (Action Selection)*, установите флажок **Протоколирование (Logging)** и задайте имя и путь к файлу.

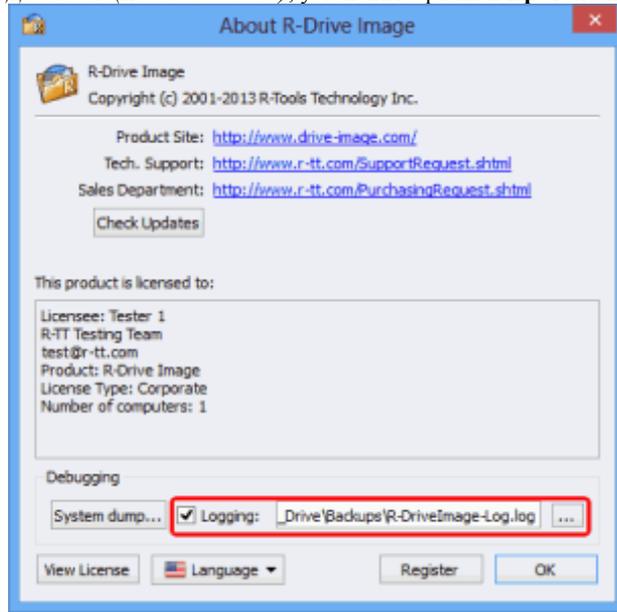


Рис. 36. R-Drive Image: Протоколирование

Кликните по изображению для его увеличения

Примеры E-mail Уведомления

После завершения операции резервного копирования R-Drive Image будет отправлять E-mail уведомление.

При *Успешном завершении операции* уведомление будет иметь следующий вид:

R-Drive Image 5.1 (Build 5100)

```
Command: create /a /o -s="hdd_size=640135028736+part_num=1+hdd_num=1+hdd_target_id=0+hdd_bus_type=sata2
+part_ofs=1048576+hdd_name=SAMSUNG&#32;HD642JJ1AA01110+part_size=367001600+hdd_port_num=2
+hdd_serial=S1AFJ1MQ400283+part_fs=ntfs+hdd_vtype=real,hdd_size=640135028736+part_num=2
+hdd_num=1+hdd_target_id=0+hdd_bus_type=sata2+part_label=System+part_ofs=368050176
+part_mounted=C:\+hdd_name=SAMSUNG&#32;HD642JJ1AA01110+part_size=209348198400
+hdd_port_num=2+hdd_serial=S1AFJ1MQ400283+part_fs=ntfs+hdd_vtype=real"
-a="\\BCK-UBUNTU\Net_Drive\Backups\SystemDisk.rdr" -p="*****" -r="Full backup image of the system disk."
-c="6" -u -check -bs -bs-num-b="3" -ms="mail.example.com:25" -ml="*****" -ma="server1@example.com"
-mr="sysadmin_of_server1@example.com" -mc="Server 1: full system disk backup" -mx -me
Start at: Sun, 01 Sep 2013 14:00:00 -0500
Finish at: Sun, 01 Sep 2013 15:08:39 -0500
Success
```

Operations:

```
Create an Image: \\BCK-UBUNTU\Net_Drive\Backups\SystemDisk.rdr
Backup partition [SAMSUNG HD642JJ1AA01110 (596GB #1)]
Active Partition #1 (NTFS 350MB)
C: System (NTFS 194GB #2)
Backup disk partition structure
SAMSUNG HD642JJ1AA01110 (596GB #1)
Check an Image File
```

Execution log:

```
* Create an Image: \\BCK-UBUNTU\Net_Drive\Backups\SystemDisk.rdr
Backup partition [SAMSUNG HD642JJ1AA01110 (596GB #1)]
Active Partition #1 (NTFS 350MB)
C: System (NTFS 194GB #2)
Backup disk partition structure
SAMSUNG HD642JJ1AA01110 (596GB #1)
Check an Image File
* Operation completed successfully
```

Если операция завершилась неудачно, то уведомление будет выглядеть так:

R-Drive Image 5.1 (Build 5100)

```
Command: create /a /o -s="hdd_size=522713088+part_num=1+hdd_num=2+hdd_target_id=0
+hdd_bus_type=usb+part_label=RS+part_ofs=65536+hdd_name=Flash&#32;Disk4.00
+part_size=522647552+hdd_port_num=0+hdd_serial=078163578514+part_fs=fat16+hdd_vtype=real"
-a="D:\Backups\HDD2_2-image.rdr" -u -check -ms="smtp.example.com:25"
-ml="*****" -ma="sender@example.com" -mr="receiver@example.com"
-mc="Flash backup" -mx -me
Start at: Sun, 01 Sep 2013 18:06:47 -0500
Finish at: Sun, 01 Sep 2013 18:06:49 -0500
ERROR: Internal error (error #0:3831)
```

Operations:

```
Create an Image: D:\Backups\HDD2_2-image.rdr
Backup partition [KingstonDataTraveler 400PMAP (3.74GB #2)]
Active Partition #1 NEW VOLUME (FAT32 3.73GB)
Backup disk partition structure
KingstonDataTraveler 400PMAP (3.74GB #2)
Check an Image File
```

Execution log:

```
! KingstonDataTraveler 400PMAP: Partition at 32 extends beyond disk bounds
! KingstonDataTraveler 400PMAP: Partition at 32 extends beyond disk bounds
* Create an Image: D:\Backups\HDD2_2-image.rdr
Backup partition [KingstonDataTraveler 400PMAP (3.74GB #2)]
Active Partition #1 NEW VOLUME (FAT32 3.73GB)
Backup disk partition structure
KingstonDataTraveler 400PMAP (3.74GB #2)
Check an Image File
! Read disk KingstonDataTraveler 400PMAP at position 16896 failed after 2 attempts. The handle is invalid (6)
! Read disk KingstonDataTraveler 400PMAP at position 16896 failed after 2 attempts. The handle is invalid (6)
! Read disk at position 2671616 failed after 2 attempts. The handle is invalid (6)
! Operation failed: Internal error (error #0:3831)
```

Заключение

Создать план резервного копирования при помощи R-Drive Image можно достаточно быстро, эффективно и экономично. Имея R-Drive Image, устройство NAS и план резервного копирования можно оградить себя от всех неудобств и значительных финансовых затрат имеющих место при утрате данных. В данной статье было показано насколько просто можно создать план резервного копирования. Для получения более подробной информации обо всех имеющихся параметрах резервного копирования и составления своего плана обратитесь к R-Drive Image [online справке](#).

Практическое занятие

форма текущего контроля

по теме: Создание резервной копии информационной системы

Цель: создание резервной копии информационной системы

По завершению практического занятия студент должен уметь: создавать резервную копию информационной системы

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Персональный компьютер, программное обеспечение: среда PowerDesigner.

Задание

1. Резервное копирование

Многие программы-настройщики (иначе Твикеры) предлагают создать резервный диск восстановления Windows. То же предлагает сделать Антивирус Касперского, дабы восстановить работу Windows после серьезной вирусной атаки.

Можно заархивировать содержимое папки \Windows\System32\config через другой компьютер либо же с помощью загрузочной версии Windows, чтобы в случае появления сообщения "\Windows\System32\config файл поврежден" можно было его распаковать обратно и тем самым восстановить работу Windows.

Подобная ошибка появляется из-за повреждения кластеров, но может произойти из-за экстренного завершения работы.

При повреждении кластеров может помочь проверка на ошибки системного диска с исправлением ошибок, ее можно произвести с помощью другого компьютера, либо же Загрузочной версии Windows, но такой метод является экстренным и не желаемым, поскольку Windows скрывает поврежденные кластера, вместо того чтобы восстанавливать их. В этом случае оптимальным вариантом будет использование HDD Regenerator'a, поскольку он именно восстанавливает поврежденные кластера.

В некоторых случаях на системном диске повреждается файл NTLDR (NT Loader). В следствие чего появляется сообщение:

"NTLDR is Missing". Чтобы исправить данную ошибку в некоторые сборки Windows XP включается загрузочная программа

"Исправить "NTLDR is Missing"". В Windows Vista / 7 данной ошибки не наблюдается в связи с отсутствием файла NTLDR, его заменяет BootMGR (Boot Manager).

Основные средства восстановления работоспособности:

  Hiren's Boot CD   Windows XP Portable Edition



HDD Regenerator   Acronis Rescue Me-

Задание 1

2. Резервное копирование реестра в Windows XP

Способ 1.

Не используйте этот способ для экспорта всего реестра или его основных разделов, таких как HKEY_CURRENT_USER и т.п.

Прежде, чем начать редактирование реестра вручную с помощью REGEDIT, или REG-файла не помешает **сохранить ту часть реестра**: раздел или подраздел, которую вы будете изменять. Для этого:

- Запустите REGEDIT. "**Пуск- Выполнить- REGEDIT**".
- Найдите ветвь реестра содержащую ключ значение которого вы будете редактировать и кликните на ней, в левой части окна **REGEDIT**.
- В главном меню выберите "**Файл-Экспорт**" и укажите имя файла. Либо кликните правой кнопкой и укажите "**Экспортировать**".

Альтернативный вышеприведенному способ состоит в том, что можно выполнить команду или командный файл определённого содержания. Например, сохраним настройки популярной программы Mozilla или Google:

Выполните

Для Mozilla:

Пуск – Выполнить – и введите команду:

```
Regedit /e mozilla1.reg HKEY_CURRENT_USER\Software\Mozilla\FireFox\ u
```

```
Regedit /e mozilla2.reg HKEY_LOCAL_MACHINE\Software\Mozilla\FireFox\
```

Вся необходимая информация будет помещена в файлы mozilla1.reg и mozilla2.reg.

Для Google

```
Regedit /e Chrome1.reg HKEY_CURRENT_USER\Software\ Google \Chrome\ u
```

```
Regedit /e Chrome2.reg HKEY_LOCAL_MACHINE\Software\ Google \Chrome\
```

Вся необходимая информация будет помещена в файлы mozilla1.reg и mozilla2.reg.

Способ 2.

Для резервного копирования всего реестра используйте программу архивации данных "**Программы-Стандартные-Служебные-Архивация данных**" или просто введите команду `%SystemRoot%\system32\ntbackup.exe` в «Пуск-Выполнить»

В открывшемся окне нажмите кнопку **Далее**

В открывшемся окне поставьте галочку в пункте **Архивация файлов и параметров** и нажмите **Далее**

В открывшемся окне выберите пункт **Предоставить возможность выбора объектов для архивации** и нажмите **Далее**.

В открывшемся окне выберите папки или документы, которые должны быть заархивированы и нажмите **Далее**.

В открывшемся окне выберите место сохранения архива и нажмите **Далее** и **в новом окне нажмите Готово**. После нажатия кнопки **Готово** начнется процесс архивации.

Программа архивации позволяет архивировать и восстанавливать так называемые данные состояния системы, что включает в себя следующие системные компоненты:

- реестр;
- базу данных регистрации классов COM+
- загрузочные файлы : Ntldr и Ntdetect.com
- системные файлы;

Задание 2.

Пошаговые инструкции для архивации реестра Windows XP:

1. Войдите в систему с необходимыми правами, например, как администратор.
2. Запустите NTBackup ("Пуск – Стандартные – Служебные - Архивация данных").
3. Если NTBackup запустилась в режиме мастера, перейдите в "Расширенный режим".
4. Выберите закладку "Архивация".
5. В левом окне найдите и пометьте "птичкой" строку "Диск C:\Windows\System32".
6. Нажмите кнопку "Архивировать" и выберите "Дополнительно".
7. Снимите "галочку" с пункта "Автоматически архивировать защищенные системные файлы вместе с состоянием системы". Таким образом мы заархивируем только файлы реестра, что произойдет быстро и займет немного места на диске, примерно 17-20Мб.
8. На этой же вкладке "Тип архива" установите "Обычный".
9. "ОК" и нажмите "Архивировать". После архивации вы сможете просмотреть отчет.
10. Отчёты об архивации накапливаются в папке

x:\Documents and Settings%\User%\Local Settings\Application Data\Microsoft\Windows NT\NTBackup\data\

в пронумерованных файлах backup01.log, backup02.log и т.д.

NTBackup можно использовать и из командной строки, но мы не будем рассматривать этот способ, так как восстановить данные с командной строки нам не удастся и, кроме того, при архивации вместе с реестром будут заархивированы и все системные файлы, необходимые для загрузки Windows XP. А это потребует более долгого времени и займёт заметно больше места на жестком диске.

Восстановление реестра в Windows XP

В данном разделе мы практически повторим предыдущий, но с точки зрения восстановления реестра, а не архивации.

Задание 3

Способ 1.

При архивации части реестра, мы с помощью REGEDIT экспортировали данные в REG-файл. Теперь, чтобы извлечь их и восстановить исходный вид части реестра выполним следующие шаги:

1. Запустите REGEDIT. "Пуск-Выполнить-REGEDIT". 2. В главном меню выберите "Файл-Импорт" и укажите имя файла из задания 1 .

Или можно выполнить команду или командный файл определённого содержания. Например, восстановим настройки программы Mozilla:

Выбираем Пуск – Выполнить и вводим команду:

```
regedit -s mozilla1.reg regedit -s mozilla2.reg
```

Вся необходимая информация будет взята из файлов MOZILLA1.REG и MOZILLA2.REG.

Способ 2.

Пошаговые инструкции для полного восстановления реестра Windows XP:

1. Войдите в систему с необходимыми правами, например, как администратор.
2. Запустите NTBackup.
3. Если NTBackup запустилась в режиме мастера, нажмите кнопку "Расширенный" в окне мастера архивации.
4. Перейдите на вкладку "Восстановление и управление носителем"
5. Установите в списке "Установите флажки для всех объектов, которые вы хотите восстановить" флажок для объекта "Состояние системы". Это позволит восстановить данные состояния системы вместе с остальными данными, отмеченными в текущем задании восстановления.
6. Отчёты о проделанной работе находятся в папке x:

\\Documents and Settings%\%User%\Local Settings\Application Data\Microsoft\Windows NT\NTBackup\data\ в пронумерованных файлах типа backup01.log, backup02.log и т.д.

Восстановление повреждённого реестра когда Windows XP не загружается

А теперь мы посмотрим, что нужно делать, когда из-за ошибок в реестре Windows XP не загружается.

Описываемая процедура не гарантирует полное восстановление системы к предыдущему состоянию; однако, мы сможем восстановить наши данные.

Разрушенные файлы системного реестра могут вызывать ряд различных сообщений об ошибках.

Попробуйте при загрузке Windows XP нажать F8 и выбрать вариант "Загрузка последней удачной конфигурации" (Boot Using Last Known Good Configuration). При этом восстанавливаются только данные в разделе реестра HKLM\System\CurrentControlSet. Любые изменения в других разделах реестра сохраняются. Загрузка последней удачной конфигурации позволяет восстановить реестр в случае неполадок, вызванных, например, новым, несовместимым с имеющимся оборудованием, драйвером. Неполадки, возникшие вследствие повреждения или ошибочного удаления драйверов или файлов, не могут быть устранены таким образом.

Итак, при попытке запуска Windows XP вы получаете сообщение об ошибке, например, одно из указанных ниже:

Windows XP could not start because the following file is missing or corrupt: \WINDOWS\SYSTEM32\CONFIG\SYSTEM

Windows XP could not start because the following file is missing or corrupt: \WINDOWS\SYSTEM32\CONFIG\SOFTWARE

Stop: c0000218 {Registry File Failure} The registry cannot load the hive (file): \SystemRoot\System32\Config\SOFTWARE or its log or alternate

Очень хорошо, теперь настала пора применить ваши знания на практике. Если вы когда-либо выполняли NTBACKUP и завершили системное копирование успешно, то вы можете сразу приступить к **4-ому шагу**.

Шаг 2.

Чтобы выполнить процедуру, описанную в этом разделе, вы должны войти как администратор, или как пользователь приравненный к администратору. Т.е. пользователь имеющий учетную запись в группе Администраторы.

Выполняем следующие действия:

1. Перегрузите компьютер.
2. При загрузке Windows XP нажмите F8.
3. Выберите безопасный режим.

Если вы используете проводник в качестве файл-менеджера, то придётся выполнить несколько действий, чтобы сделать папку System Restore видимой:

1. Запускаем "Проводник".
2. В меню "Сервис" выбираем "Свойства папки" и далее закладку "Вид".
3. Раскрываем опцию "Скрытые файлы и папки" и щёлкаем на "Показывать скрытые файлы и папки".
4. Далее щёлкаем на "Применить" и "Ок".

Теперь:

1. Открываем раздел жёсткого диска где установлена Windows XP и находим папку System Volume Information. Примечание: Это скрытая системная папка. Она содержит одну или более папок с именами вида _restore {GUID} , например, _restore{87BD3667-3246-476B-923F-F86E30B3E7F8}
2. Откройте папку, которая была создана НЕ в текущее время. Это может быть одна или больше папок, имена которых начинаются с "RP". Это - точки восстановления.
3. Откройте выбранную папку и затем папку с именем Snapshot. Например, c:\System Volume Information_restore{DBB3294C-F5C9-43A9-9010-A75010CD2631}\RP2\snapshot
4. Из папки Snapshot в папку C:\Windows\Tmp, уже созданную на первом этапе, скопируйте следующие файлы:
 - REGISTRY_USER_DEFAULT
 - REGISTRY_MACHINE_SECURITY
 - REGISTRY_MACHINE_SOFTWARE
 - REGISTRY_MACHINE_SYSTEM
 - REGISTRY_MACHINE_SAM

Эти файлы созданы службой восстановления системы - System Restore. Так как на предыдущем шаге мы использовали файлы системного реестра, созданные при начальной установке Windows XP, то этот "новый" системный реестр не знает, что "старые" точки восстановления существуют и доступны. При загрузке Windows XP создана новая папка с новым GUID и с новым System Volume Information, и создана новая точка восстановления, которая включает копию файлов нового системного реестра. Вот почему важно не использовать самую новую папку, особенно, если время её создания - текущее время.

Таким образом конфигурация существующей системы не знает о предыдущих точках восстановления. Нам нужна предыдущая, "старая" копия системного реестра от предыдущей, "старой" точки восстановления, чтобы сделать все предыдущие, "старые" точки восстановления доступными. Я надеюсь, что вы меня поняли.

Файлы системного реестра были скопированы в папку C:\Windows\Tmp чтобы сделать их доступными, когда мы будем находиться в Recovery Console. Мы будем использовать эти файлы, чтобы заменить ими файлы текущего системного реестра в папке C:\Windows\System32\Config. Дело в том, что в Recovery Console папка с System Volume Information в общем случае недоступна.

Практическое занятие

форма текущего контроля

по теме: Создание резервной копии базы данных

Цель: научиться создавать резервные копии базы данных

По завершению практического занятия студент должен уметь: создавать резервные копии базы данных

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

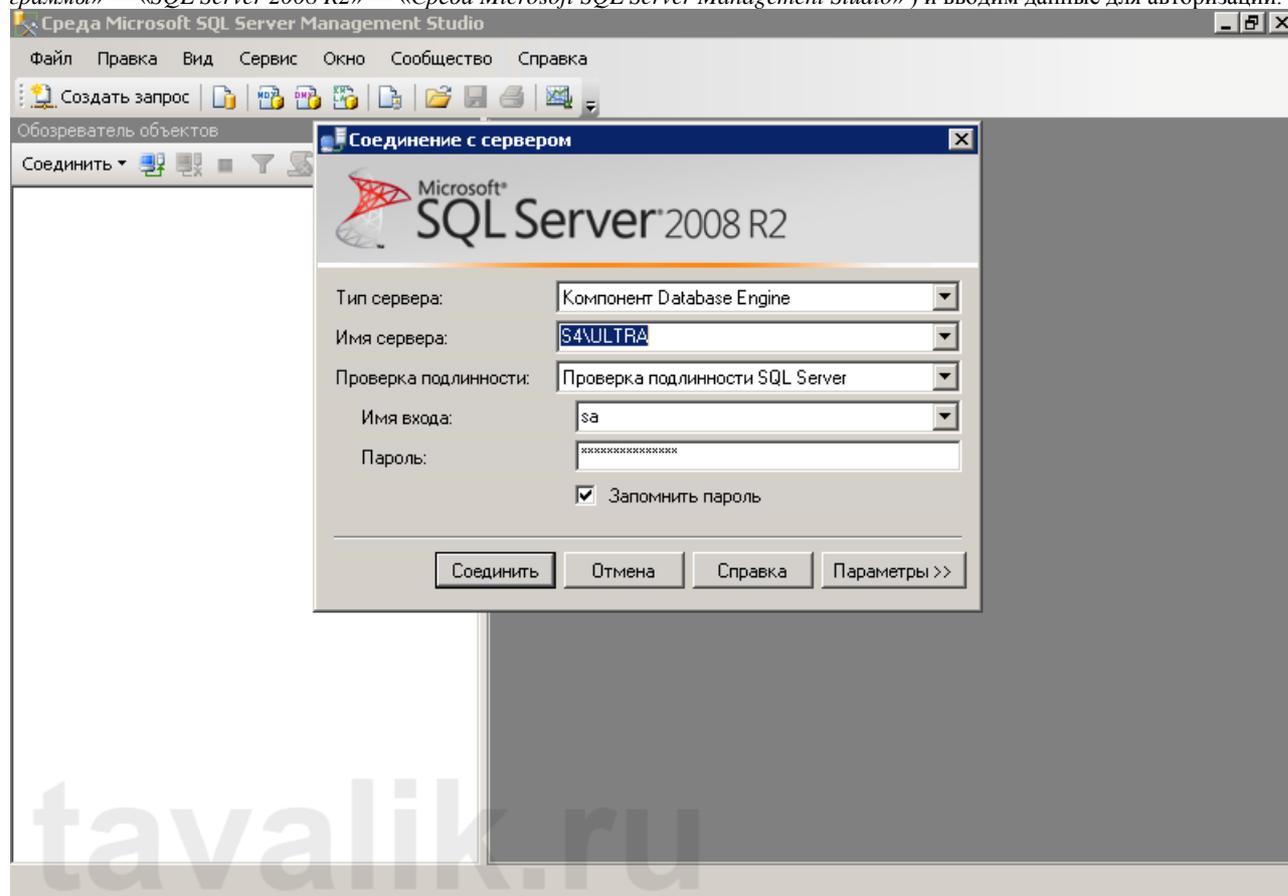
Персональный компьютер, программное обеспечение: среда PowerDesigner.

Задание

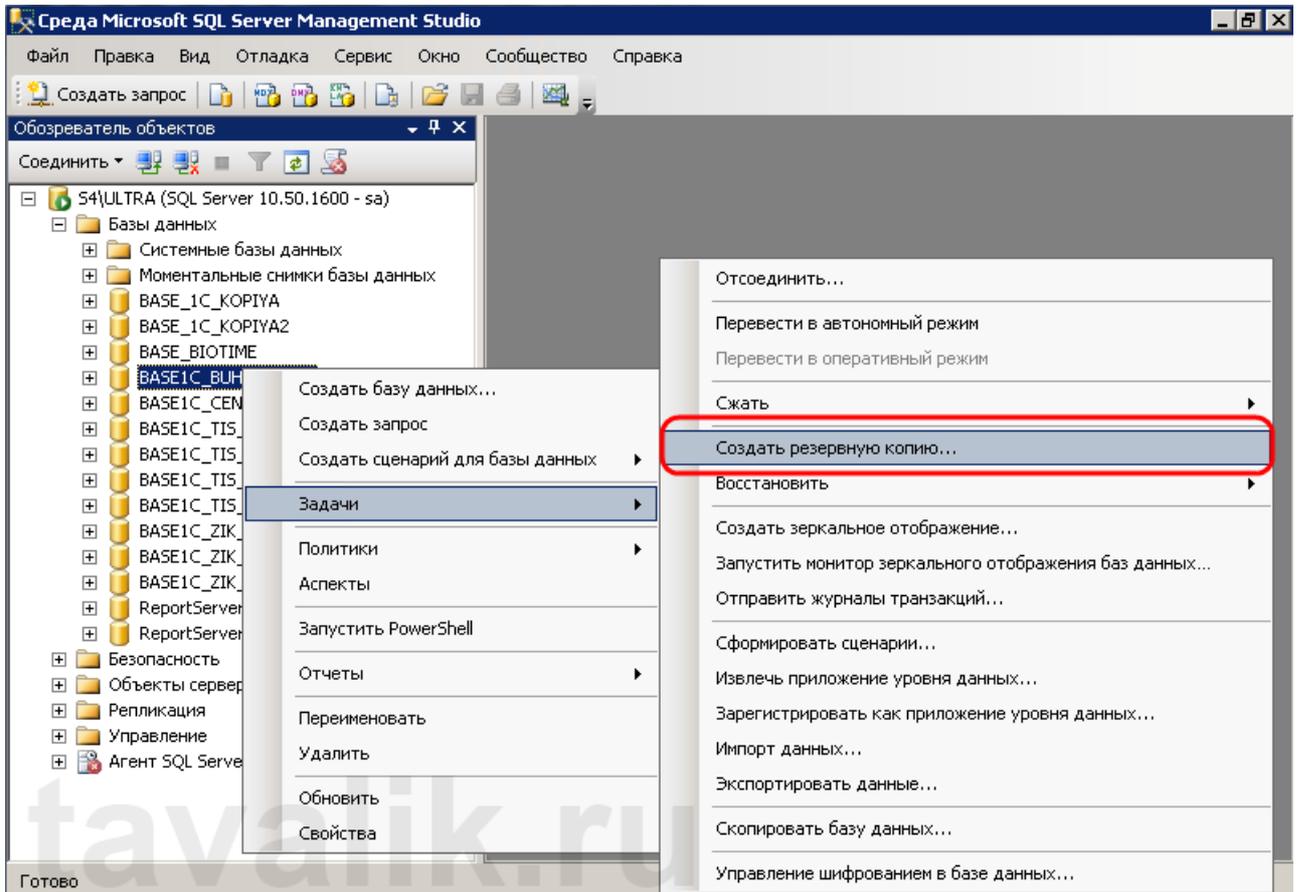
как вручную сделать полную резервную копию базы данных в SQL Server 2008 R2 с помощью программы «Среда Microsoft SQL Server Management Studio».

1. Создание резервной копии

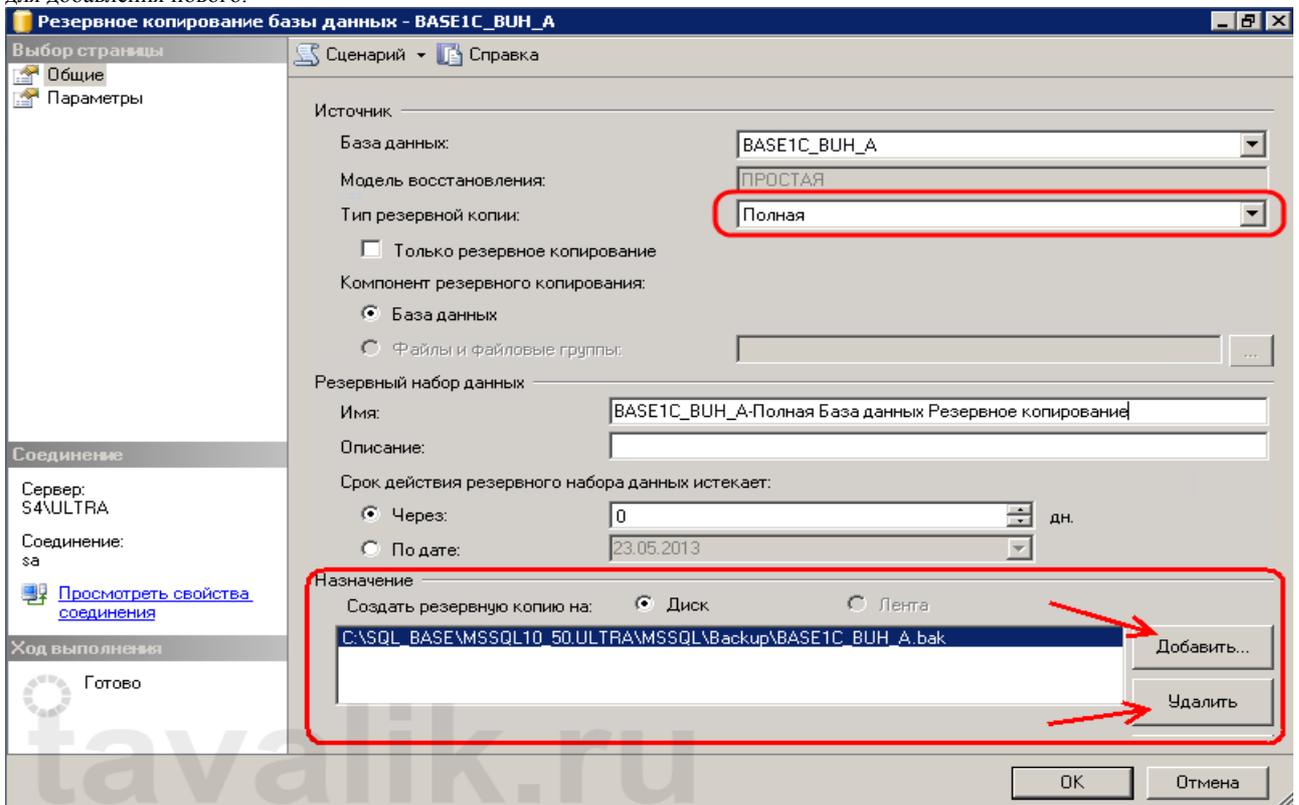
На самом деле все довольно просто. Запускаем оснастку «Среда Microsoft SQL Server Management Studio» («Пуск» — «Все программы» — «SQL Server 2008 R2» — «Среда Microsoft SQL Server Management Studio») и вводим данные для авторизации.



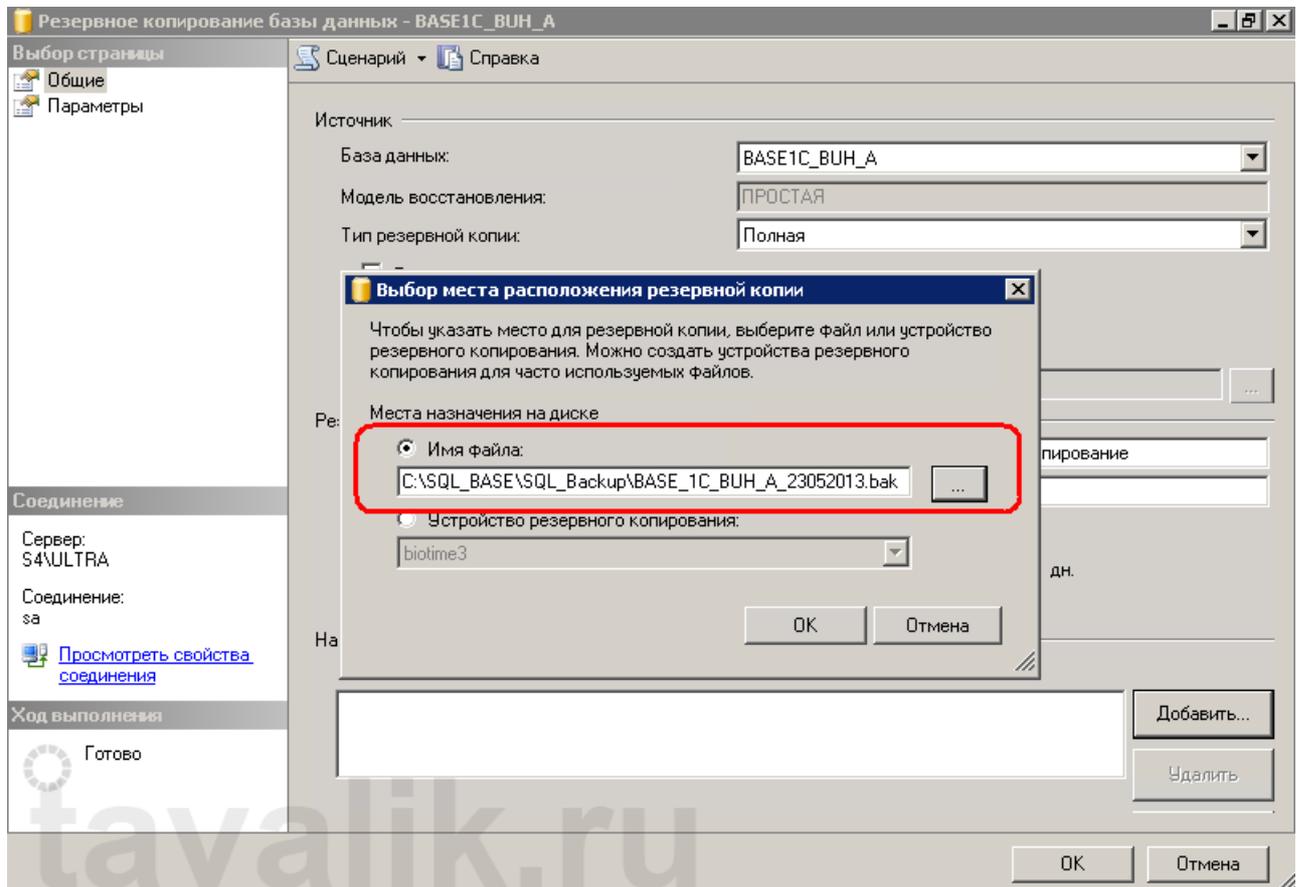
После чего в Обозревателе объектов раскрываем вкладку «Базы данных» и кликаем правой кнопкой мыши по той базе данных, для которой необходимо сделать резервную копию. В появившемся контекстном меню выберем «Задачи» (Tasks) — «Создать резервную копию» (Back Up...).



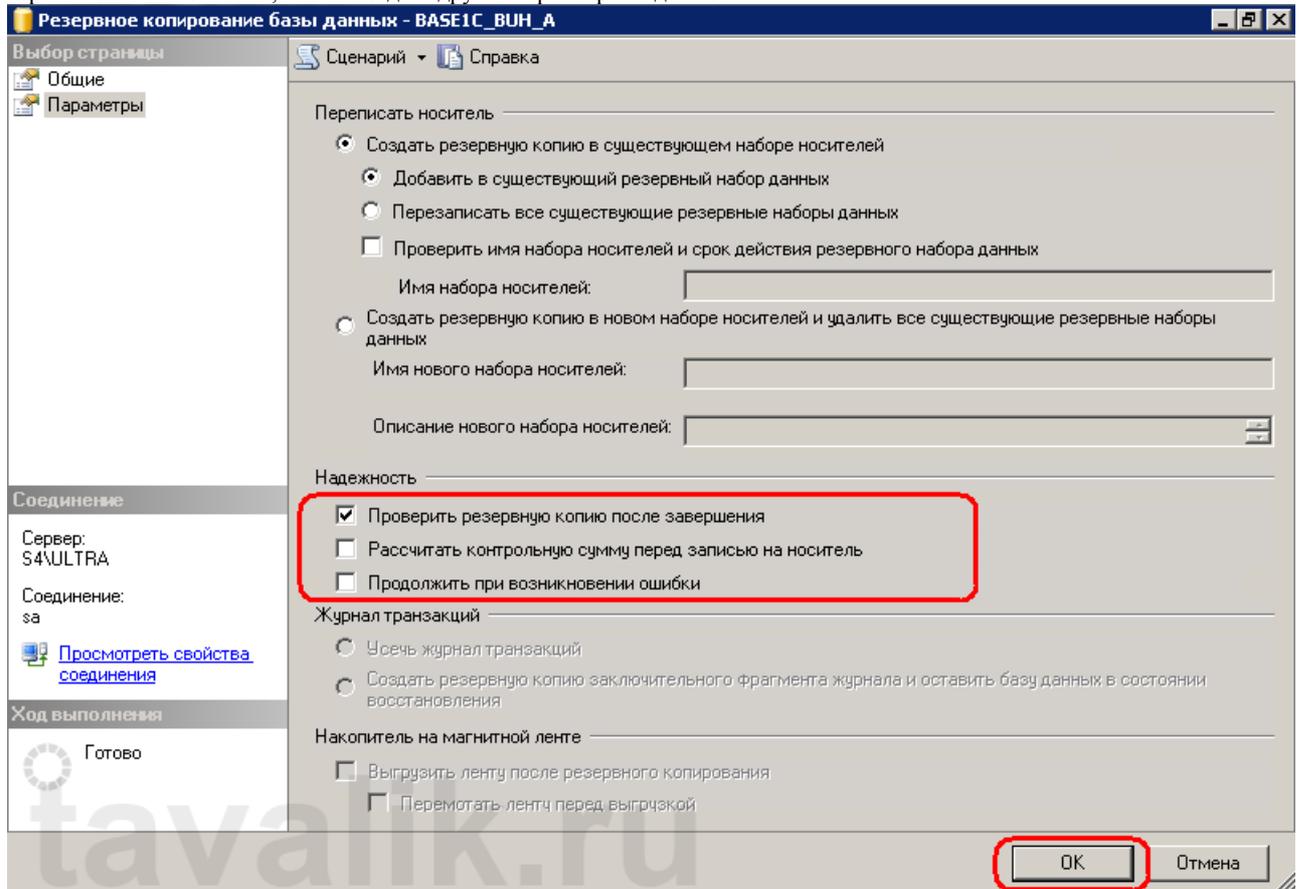
Запустится окно «Резервная копия базы данных» (Back Up Data Base). Убедимся, что тип резервной копии стоит «*Полная*» (Full), при необходимости зададим имя и описание, а также укажем назначение резервной копии. По умолчанию выбран путь на жестком диске компьютера в папку Backup основного расположения баз SQL-сервера. Для того чтобы изменить место размещения копии, сначала нажмем «Удалить» (Remove), чтобы удалить существующее назначение, а затем «Добавить» (Add...) для добавления нового.



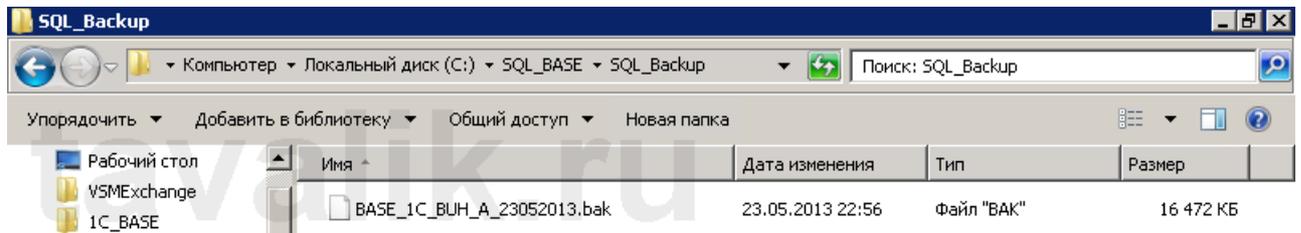
Здесь зададим расположение и имя файла резервной копии и нажмем «OK». Таких мест назначений можно задать несколько. В этом случае резервная копия будет разбита на равные части, каждая часть в указанном файле.



Далее перейдем на вкладку «Параметры» (Options), на которой можно указать, что резервную копию необходимо проверить после выполнения, а также задать другие параметры надежности.

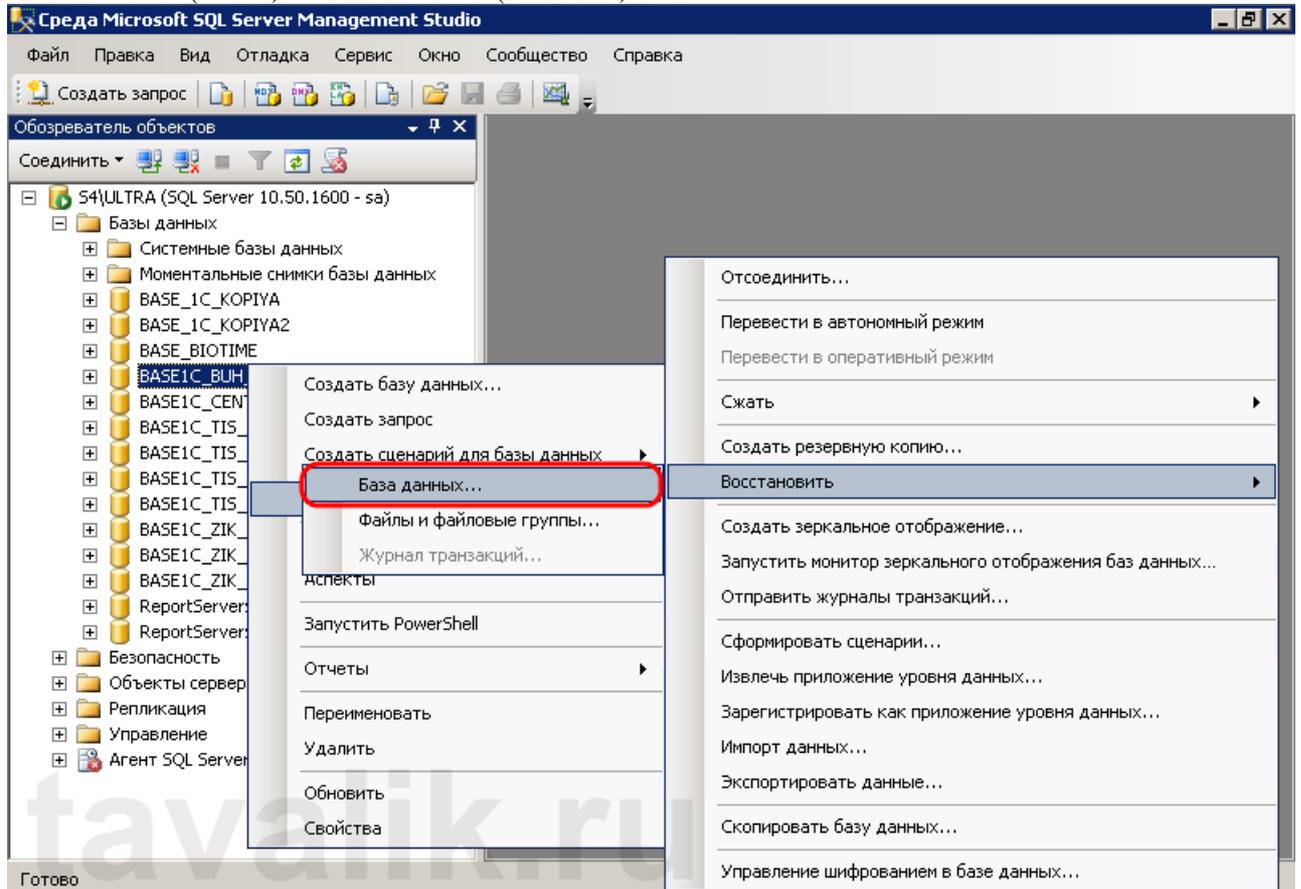


Когда все настройки установлены, нажимаем «OK» и ждем завершения задачи. Если все сделано правильно, в указанной директории мы найдем файл резервной копии базы данных SQL.

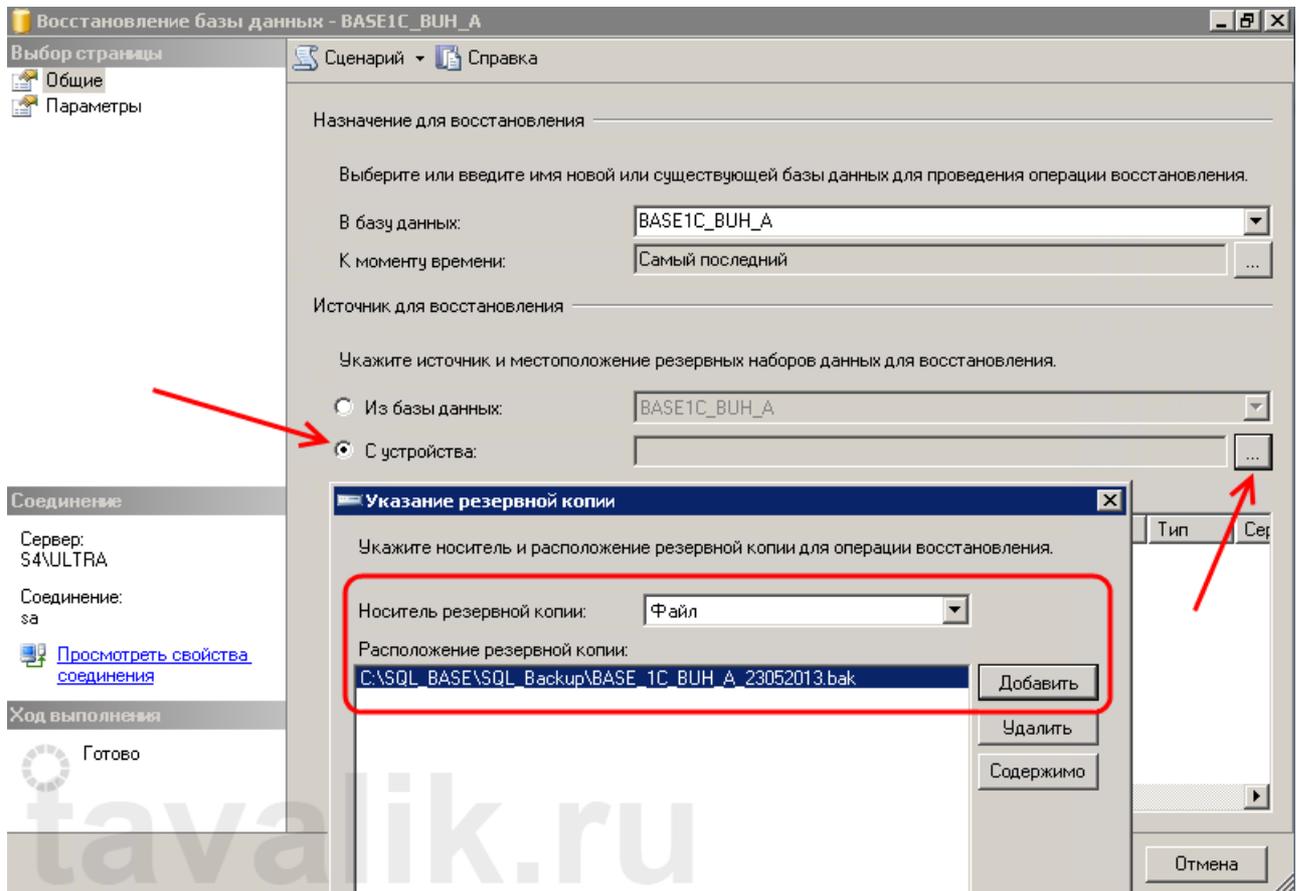


2. Восстановление базы данных из резервной копии

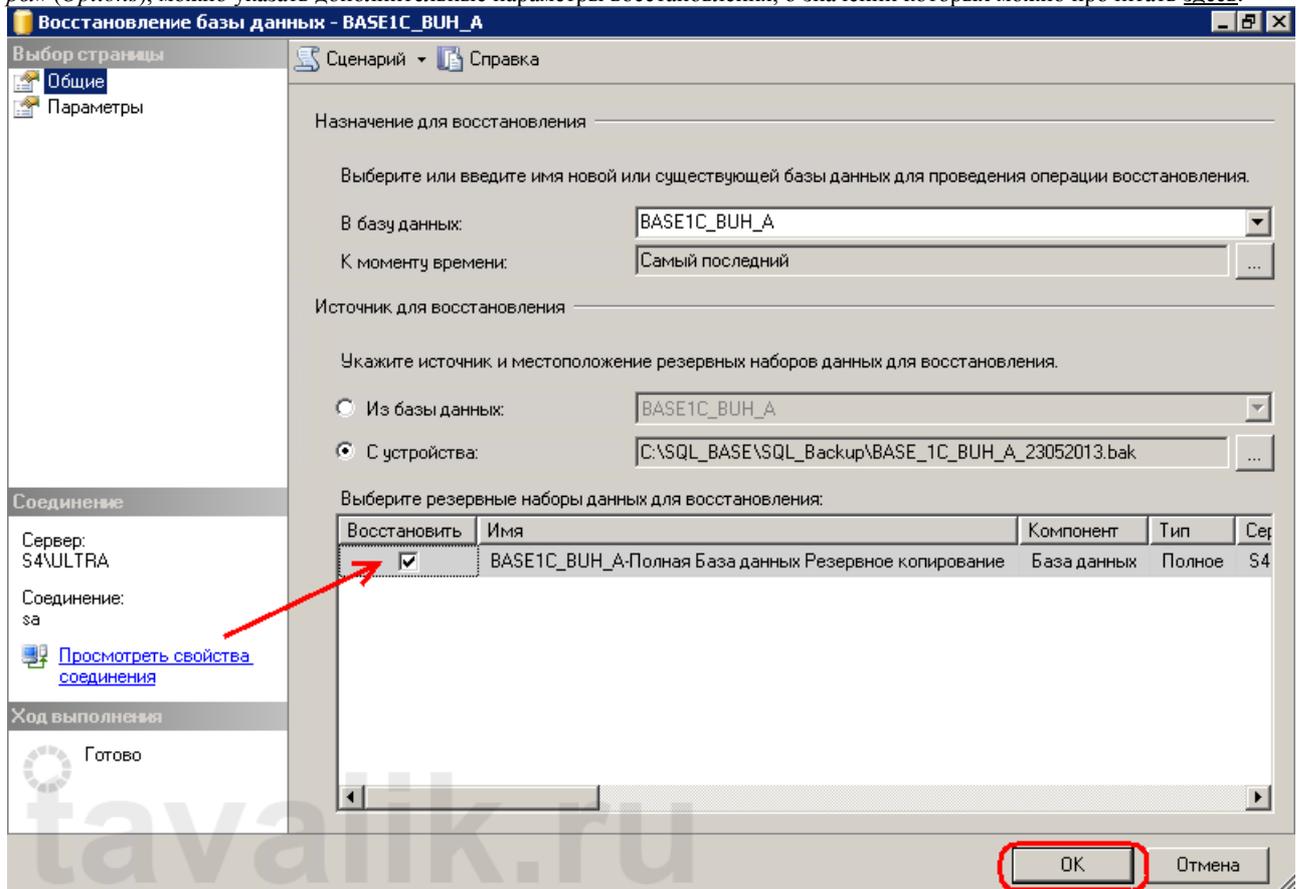
Восстановление происходит по аналогичной схеме. В «Среде Microsoft SQL Server Management Studio» выбираем базу **из которой сделана резервная копия**, кликаем по ней правой кнопкой мыши, в контекстном меню выбираем «Задачи» (Tasks) — «Восстановить» (Restore) — «База данных...» (Database...).



Откроется окно «Восстановление базы данных» (Restore Database). Здесь, в качестве источника укажем «С устройства» (From device) и выберем файл резервной копии (созданных в пункте 1).



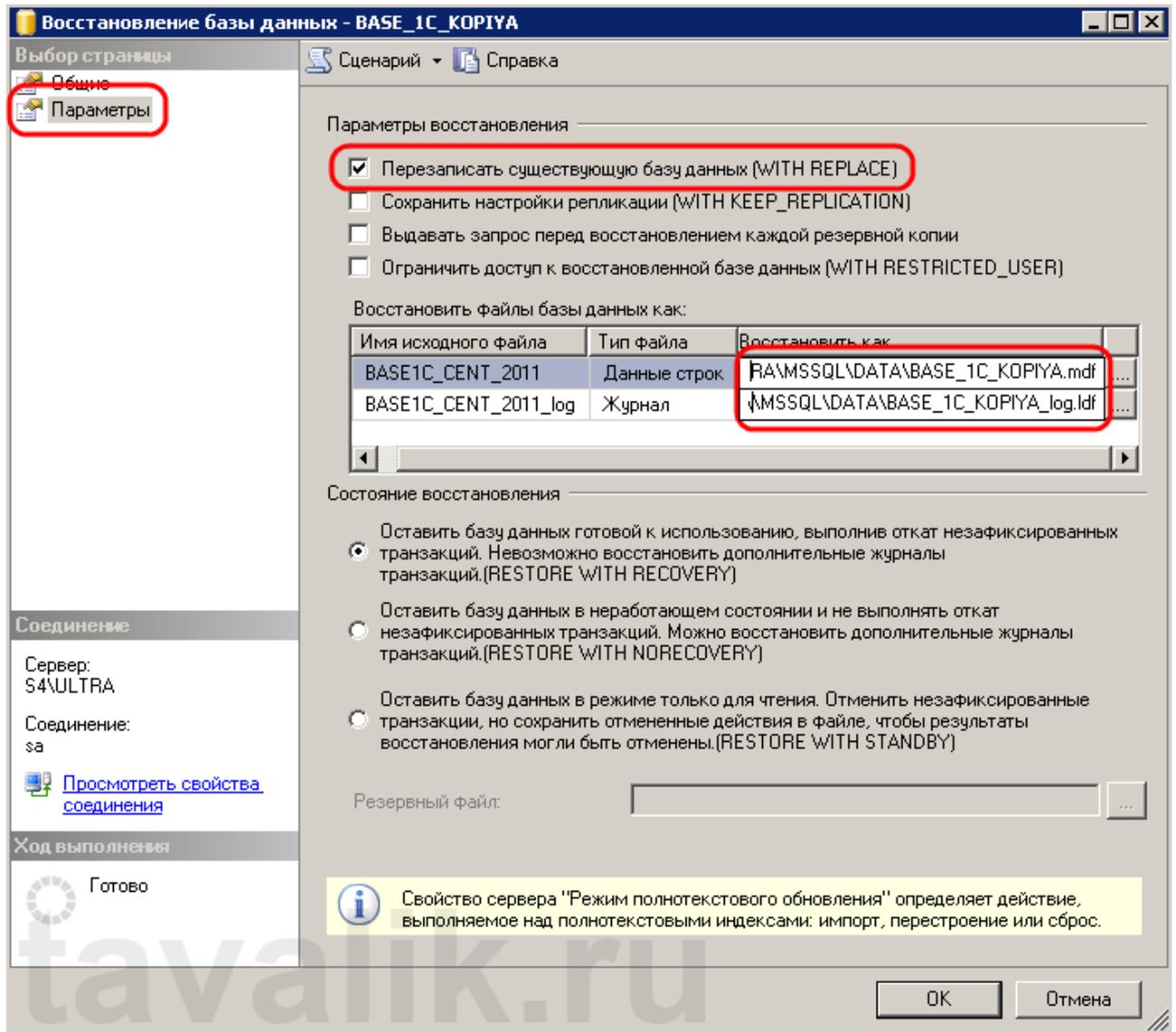
Установим флаг «Восстановить» (Restore) напротив выбранной резервной копии. При необходимости, на вкладке «Параметры» (Options), можно указать дополнительные параметры восстановления, о значении которых можно прочитать [здесь](#).



После того, как все настройки сделаны, жмем «OK» и ждем сообщения об успешном восстановлении базы данных.

3. Восстановление резервной копии в другую базу данных (копирование данных)

Если же необходимо загрузить данные в базу данных, **отличную от той из которой была сделана резервная копия**, то при загрузке помимо действий, описанных в пункте 2, необходимо на вкладке «Параметры» (Options) задать имена файлов этой базы данных и установить флаг «Перезаписывать существующую базу данных» (WITH REPLACE).



Текущий контроль бсеместр

Практическое занятие форма текущего контроля

по теме: Восстановление данных

Цель: научиться восстанавливать данные

По завершению практического занятия студент должен уметь: восстанавливать данные

Продолжительность: 3 аудиторных часа (135 минут)

Необходимые принадлежности

Персональный компьютер.

Задание

"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

1

АДМИНИСТРИРОВАНИЕ СЛУЖБ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ (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. Запротолировать 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.
- АДМИНИСТРИРОВАНИЕ СЛУЖБ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ 14.** Внести изменения в базу данных (5+N транзакций).
15. Просмотреть и зафиксировать параметры введенных транзакций (от начала последнего ввода транзакций до момента сбоя в системе включительно).
 16. Выполнить восстановление базы данных, осуществляя последовательно восстановление с использованием полной, дифференциальной копий и копии журнала транзакций.
 17. Ввести вручную зафиксированные транзакции, чтобы привести базу данных к виду, в котором она находилась на момент сбоя.
 18. Проверить правильность восстановления информации. Содержание отчета Отчет о лабораторной работе должен содержать:
 - 1) краткие теоретические сведения;
 - 2) описание содержимого базы данных (исходной, после полного и дифференциального резервного копирования) и журнала транзакций (после сохранения и до момента сбоя в информационной системе);
 - 3) описание условий, при которых производится копирование информации.

Текущий контроль
бсеместр
Практическое занятие
форма текущего контроля

по теме: Восстановление работоспособности системы

Цель: научиться восстанавливать работоспособность системы

По завершению практического занятия студент должен уметь: восстанавливать работоспособность системы

Продолжительность: 4 аудиторных часа (180 минут)

Необходимые принадлежности

Персональный компьютер, программное обеспечение.

Задание

Как запустить восстановление системы в Windows 10



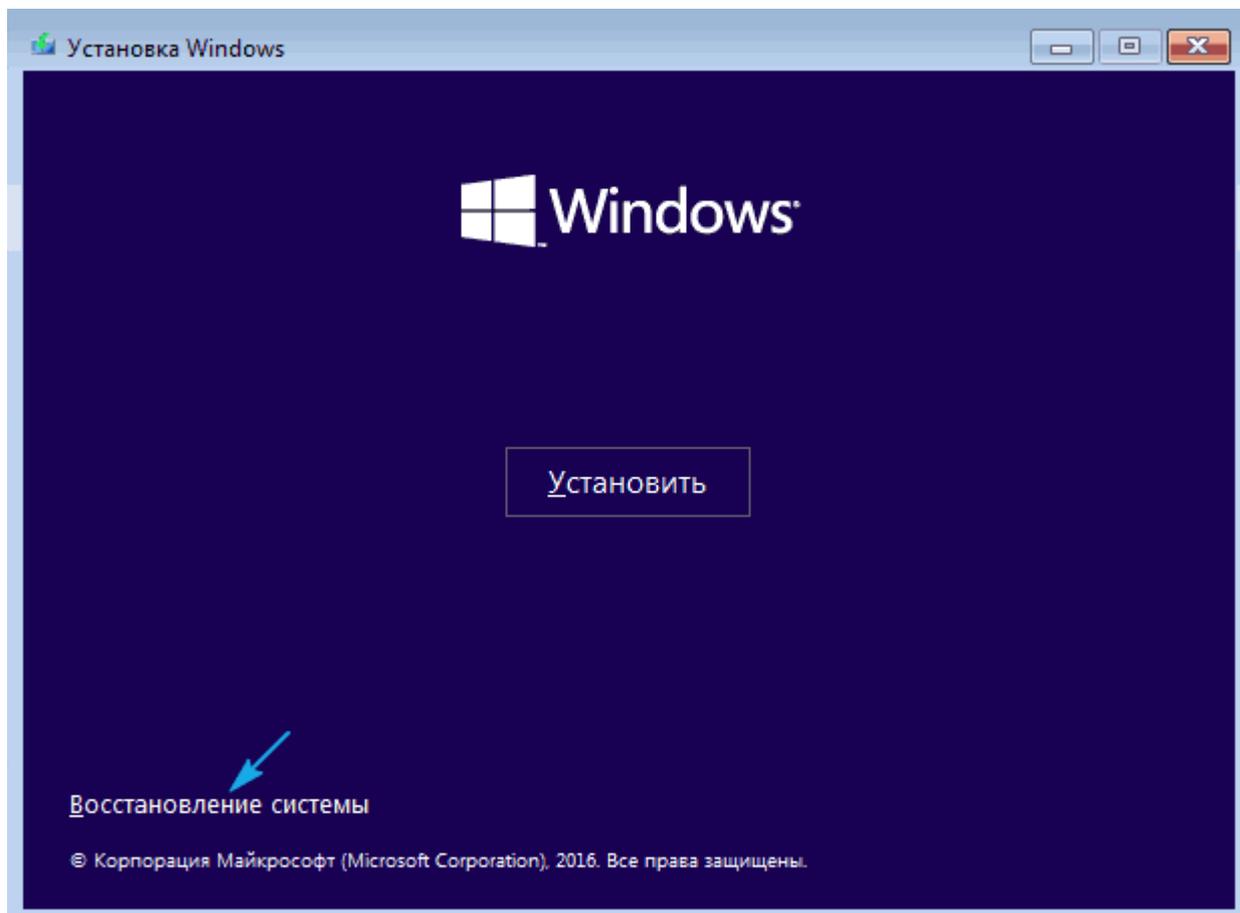
Восстановление Windows 10 позволяет вернуть операционную систему к работоспособному или исходному состоянию из созданной автоматически или вручную точки отката системы или хранимого на винчестере полного образа системы. Также в наборе инструментов «десятки» числится средство выполнения сброса ОС, которое избавит от длительной переустановки Windows 10, и создание флешки восстановления, необходимой для возобновления функционирования операционной системы в критических ситуациях (когда Windows 10 не загружается и не предоставляет возможности попасть в среду восстановления). Предложенная статья-инструкция рассказывает обо всех инструментах, из которых состоит среда восстановления системы в Windows 10, механизмах их функционирования, способах использования той или иной функции и эффективности методов восстановления в определенных ситуациях. Прежде чем приступить к изучению материала, рекомендуется ознакомиться с инструкциями на тему восстановления загрузчика операционной системы, проверки ее файлов на целостность и восстановления поврежденных файлов реестра.

- **1 Безопасный режим**
- **2 Возвращаем компьютер/ноутбук в исходное состояние**
- **3 Флешка восстановления Windows 10**
- **4 Создаем полный образ реанимации системы**
- **5 Точки отката Windows 10**
- **6 История файлов**

Безопасный режим

Первое, что следует попробовать при появлении неполадок, загрузиться в безопасном режиме. Рассмотрим ситуацию, когда «десятка» не загружается и не позволяет выполнить перезагрузку с соответствующими параметрами (попасть в этот режим через msconfig или особые варианты загрузки не получится).

1. Запускаемся из загрузочного носителя с дистрибутивом Windows 10, воспользовавшись Boot Menu.
2. Указываем «Русский» язык жмем «Далее».
3. В следующем окошке жмем по ссылке «Восстановление системы».



4. Выполняем команду «bcdedit /set safeboot minimal» для последующего запуска компьютера в безопасном режиме.
5. Перезагружаемся, закрыв все окна.

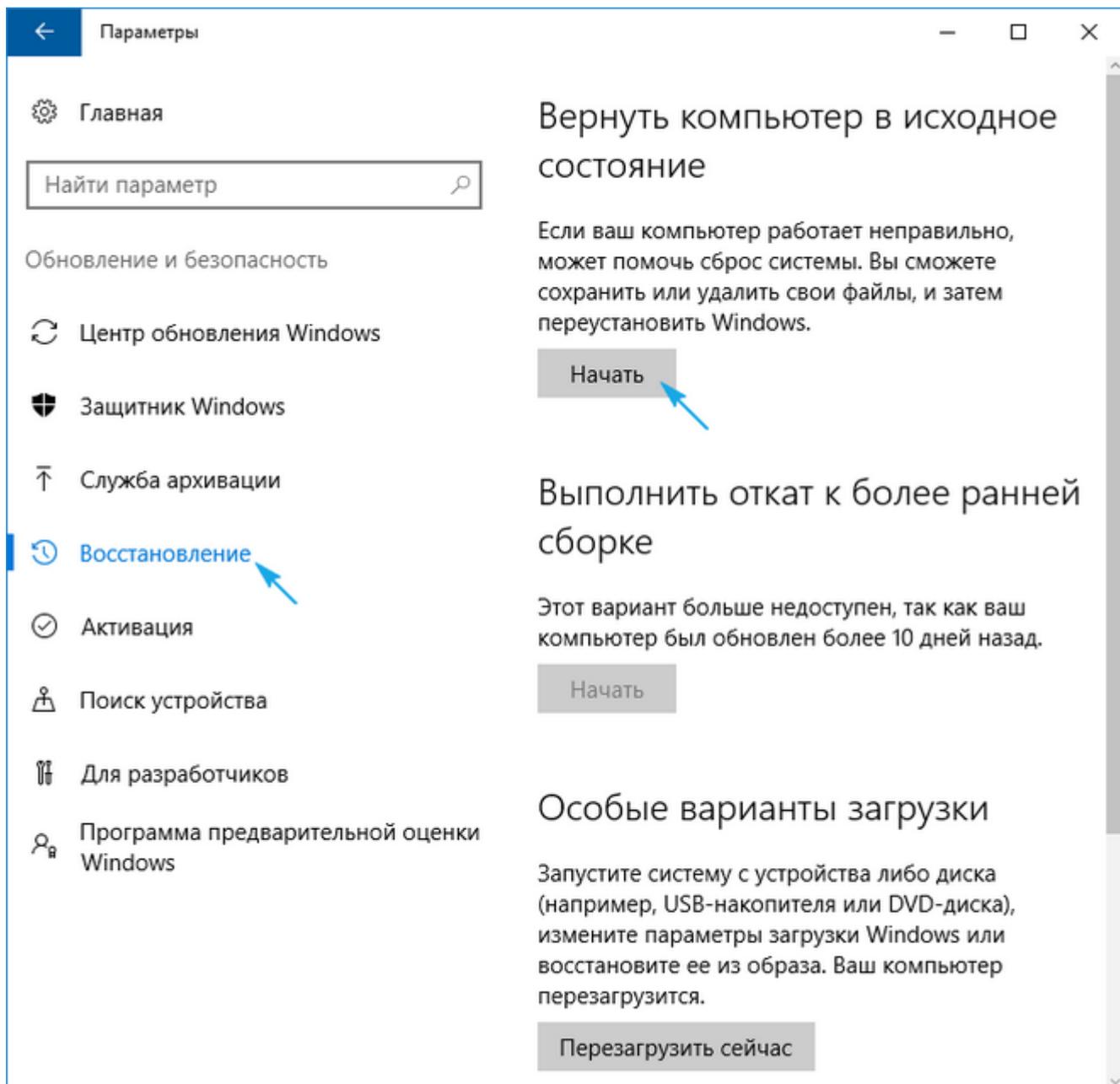
После запуска компьютера можно заняться решением проблемы, которая препятствует нормальному запуску/функционированию ПК.

Возвращаем компьютер/ноутбук в исходное состояние

Самая примечательная функция восстановления, которая появилась в Windows 10, — это возврат Виндовс к исходному состоянию. Воспользоваться ею можно через «Параметры».

1. Вызываем меню при помощи Win→I.
2. Переходим в раздел «Обновление/безопасность».
3. Нажимаем по вкладке «Восстановление».

Пользоваться этой возможностью следует в самую последнюю очередь, когда приведенные ниже варианты не помогли решить проблему, ведь в итоге получите чистую только проинсталлированную операционную систему с личными данными или без них, в зависимости от указанных параметров.



4. Жмем «Начать», после чего появится диалог с предложением указать параметры сброса операционной системы. Первый вариант — это полная очистка системного тома от всех файлов и быстрая переустановка «десятки» без задействования установочного диска с дистрибутивом операционной системы, второй способ — быстрая установка Windows 10 с сохранением файлов пользователя и настроек установленных на ПК приложений, сами же программы сохранены не будут.

Выбор действия

Сохранить мои файлы

Удаление приложений и настроек с сохранением личных файлов.

Удалить все

Удаление всех личных файлов, приложений и настроек.

Отмена

Существует еще один путь вызвать диалог сброса операционной системы даже без авторизации в системе. Осуществляется все на экране авторизации. Для получения доступа к функции жмём по пиктограмме «Перезагрузка» с зажатой клавишей Shift. После перезапуска компьютера выполняем клик по пиктограмме «Диагностика», затем жмем по кнопке возврата системы в исходное состояние.

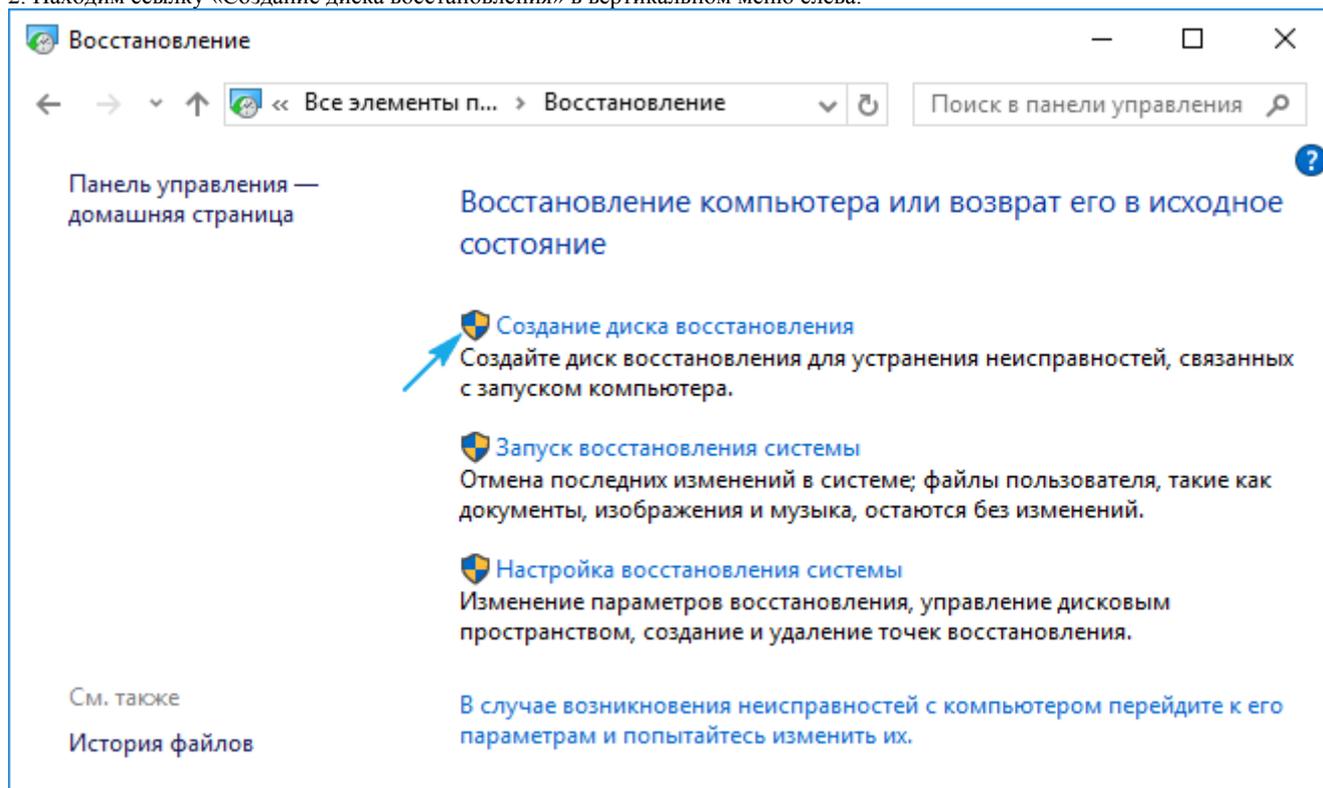
Преимуществами способа являются отсутствие необходимости иметь установочный диск/флешку и выполнение всех действий в автоматическом режиме без какого-либо вмешательства со стороны пользователя. Недостаток всего один — при удалении пользователем образа системы или расположении этого файла в поврежденных секторах жесткого диска совершить быструю переустановку не удастся, но здесь в арсенале «десятки» есть несколько дополнительных инструментов: использование диска восстановления системы при его наличии (очень редкое явление) и резервирование Windows 10 при помощи инструментов ОС на томе, отличающемся от системного.

Флешка восстановления Windows 10

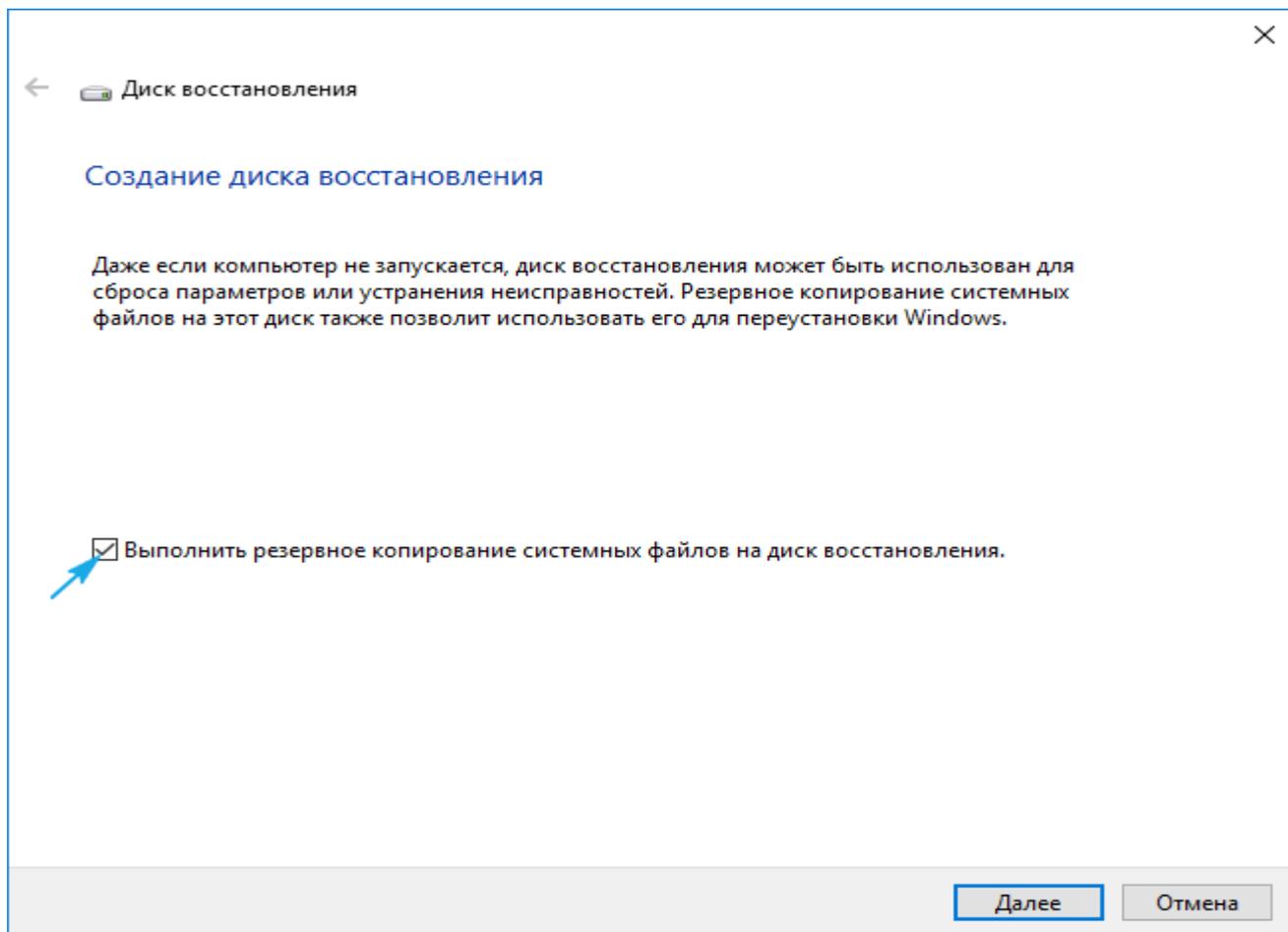
Инструмент называется диском восстановления Виндовс, но в «десятке» его следовало бы переименовать во флешку восстановления (будем пользоваться именно этим термином). Если ранее в ОС от Microsoft были утилиты для выполнения автоматической реанимации системы, которые в большинстве случаев только пытались что-то сделать, то в «десятке» присутствует опция создания образа системы для последующего возврата системного тома к запечатленному в этом образе состоянию посредством автоматической переустановки ОС, о чем говорилось разделом выше.

Создается подобный образ следующим путем:

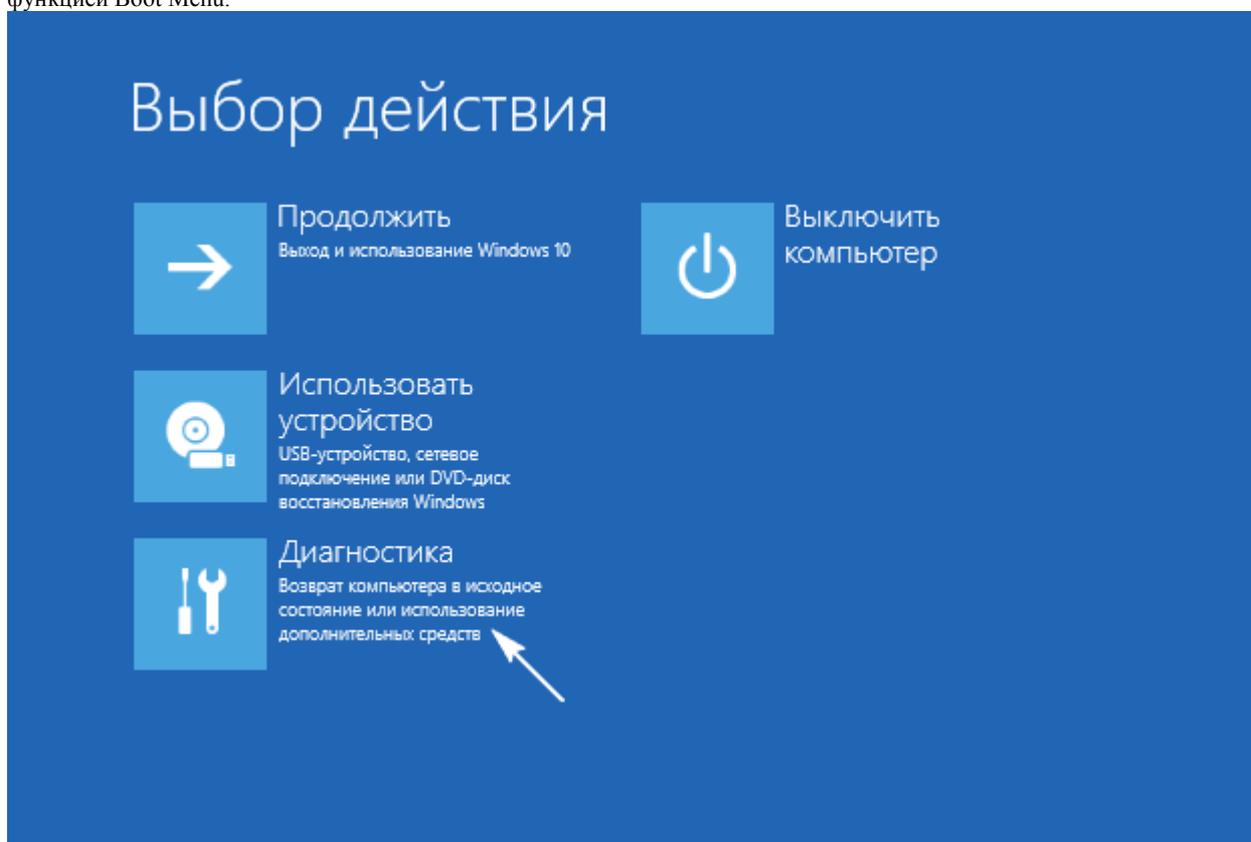
1. Вызываем апплет Панели управления под названием «Восстановление».
2. Находим ссылку «Создание диска восстановления» в вертикальном меню слева.



3. Отмечаем опцию резервирования системных файлов на флешку восстановления, чтобы получить возможность совершать мгновенную переустановку «десятки».



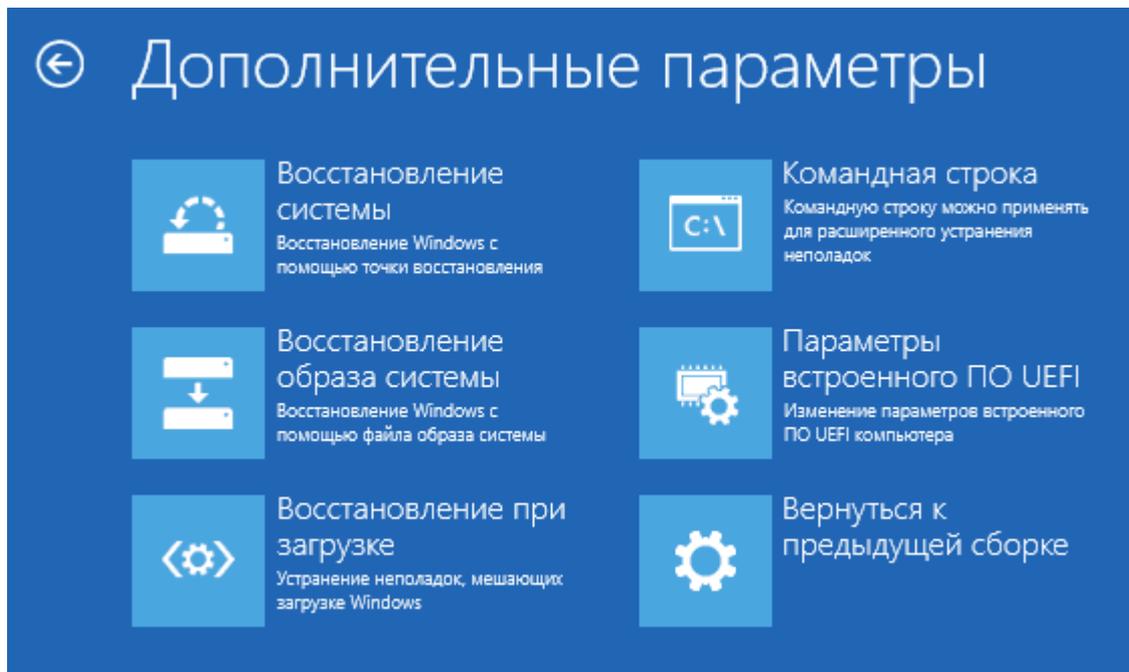
4. По окончании всех операций записи или в случае необходимости загружаемся с созданного накопителя, воспользовавшись функцией Boot Menu.



5. В окне выбора действия переходим в раздел «Диагностика».

Находясь в нем, откроем возможность выполнить следующие операции:

- воспользовавшись флешкой с образом, вернуть Windows 10 к прежнему состоянию;
- посетить параметры UEFI/BIOS;
- прибегнуть к реанимации «десятки» посредством точки отката;
- запустить через командную строку, например, для создания копии загрузчика на соответствующем томе;
- восстановить Windows 10 из полного образа ОС.



Наличие такой флешки в случае с «десяткой» намного полезнее, чем даже установочной, хотя и последняя позволяет запустить некоторые из операций восстановления операционной системы на экране с кнопкой «Установить» сразу после выбора языка.

Создаем полный образ реанимации системы

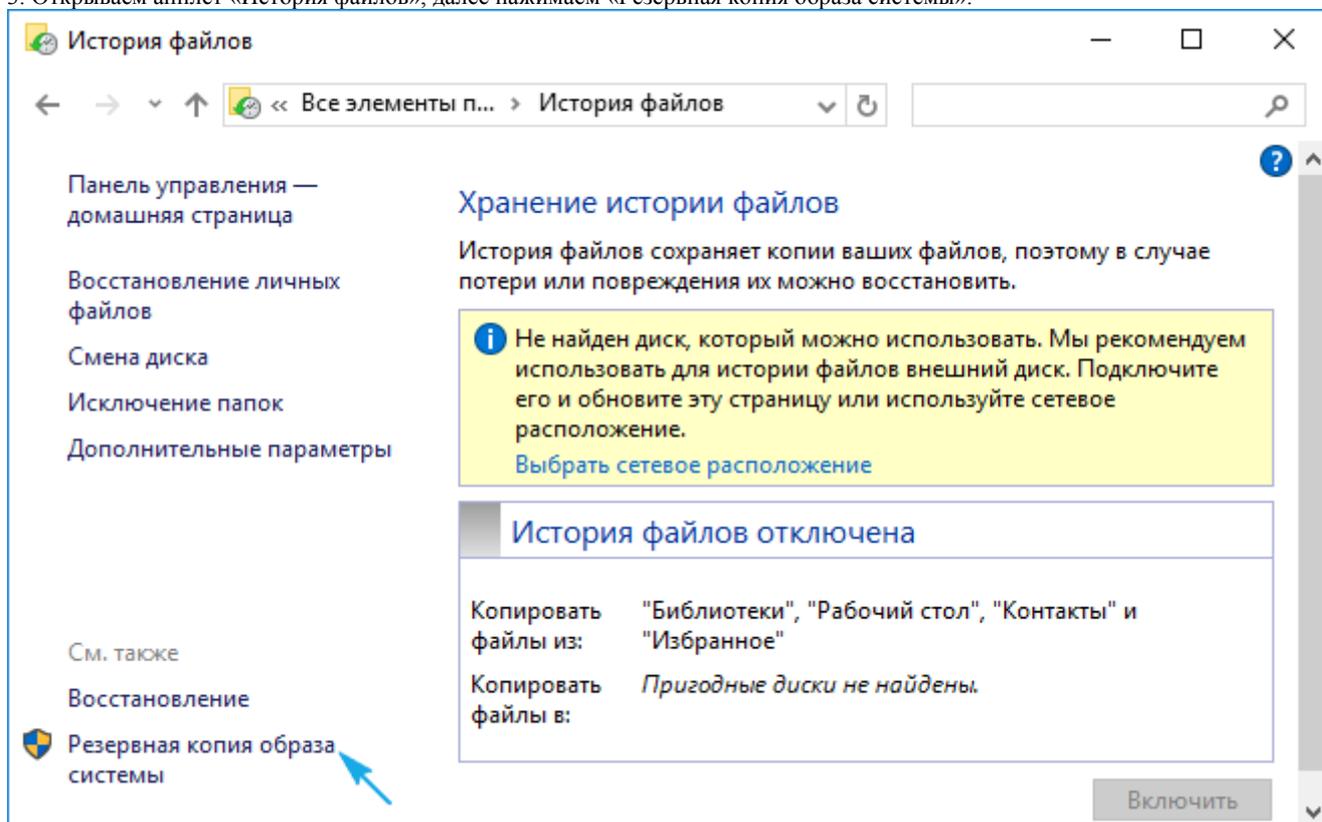
Подготовка автоматического восстановления заключается в создании снимка Windows 10 на время ее нынешнего состояния.

Лучше всего создавать такой образ сразу после инсталляции операционной системы со всеми драйверами и софтом, пока системный том не замусорен, как и реестр. **Не обязательно формировать снимок в первые часы функционирования новой ОС, это можно сделать спустя пару дней после ее переустановки, чтобы Windows притерлась и была доведена до нормального функционирующего состояния, но не успела обзавестись мусорными файлами и ключами реестра.**

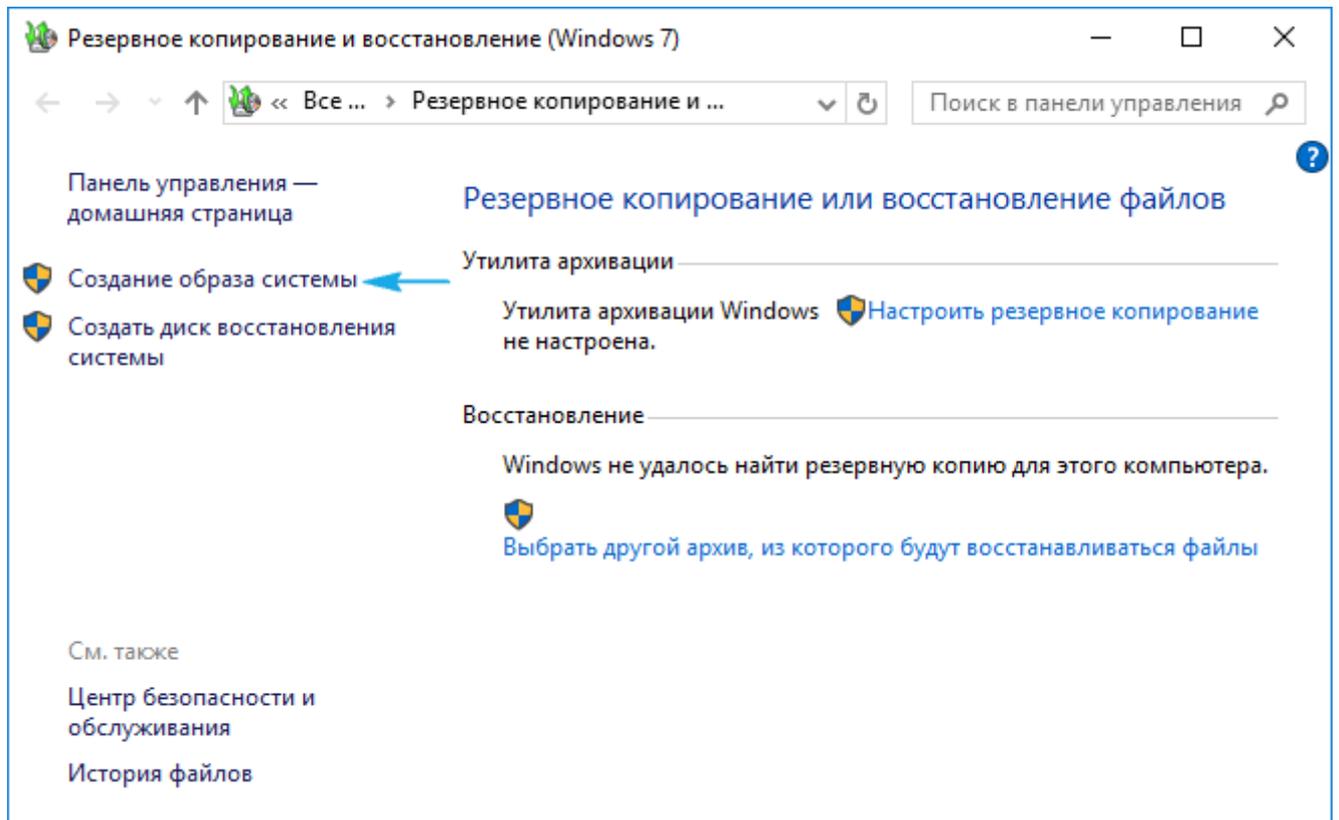
1. Процесс начинается с очистки от мусора диска C: системного реестра и деинсталляции программ, которые оказались ненужными.

2. Далее посещаем Панель управления.

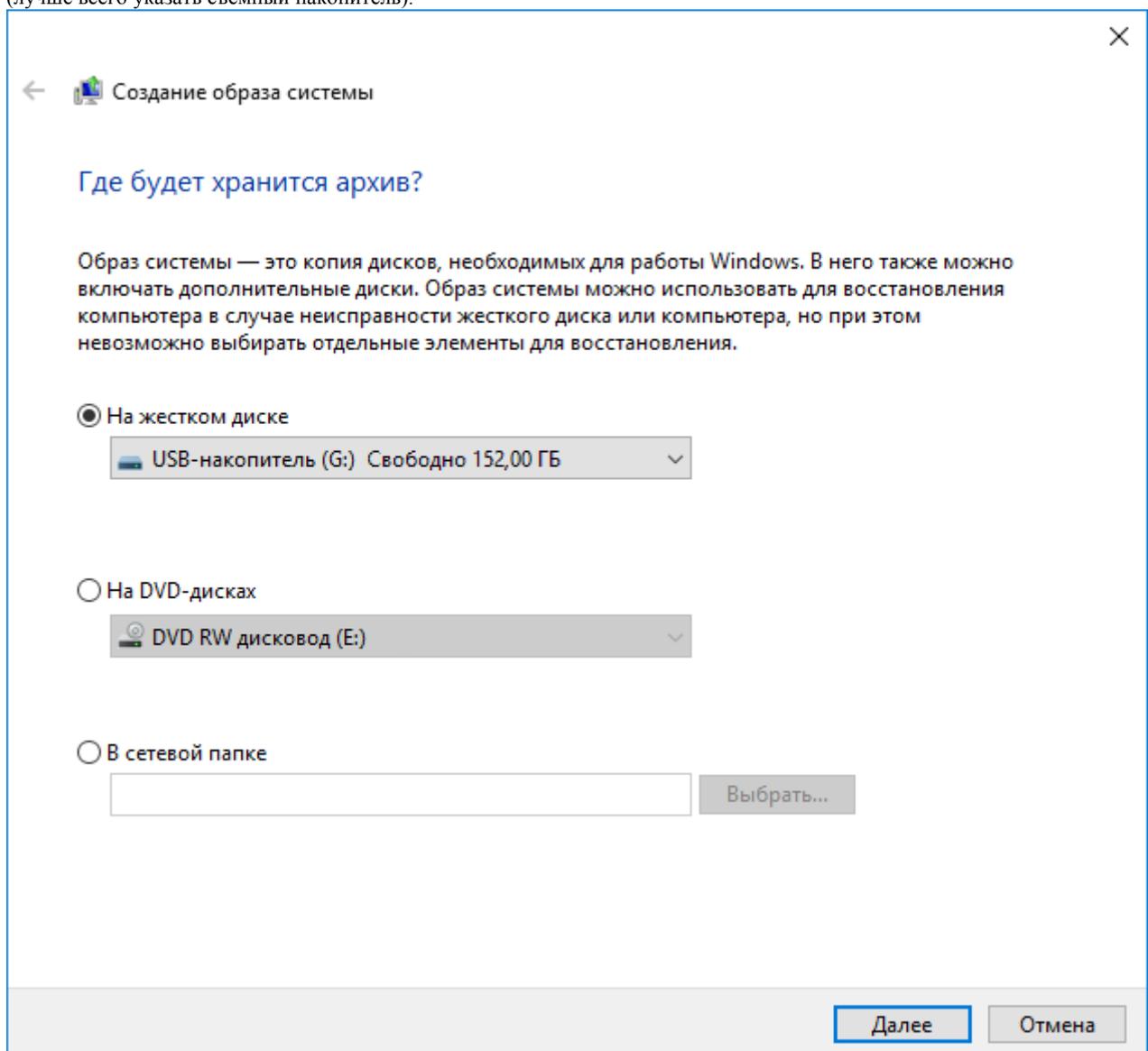
3. Открываем апплет «История файлов», далее нажимаем «Резервная копия образа системы».



4. В вертикальном меню переходим по ссылке «Создание образа системы».



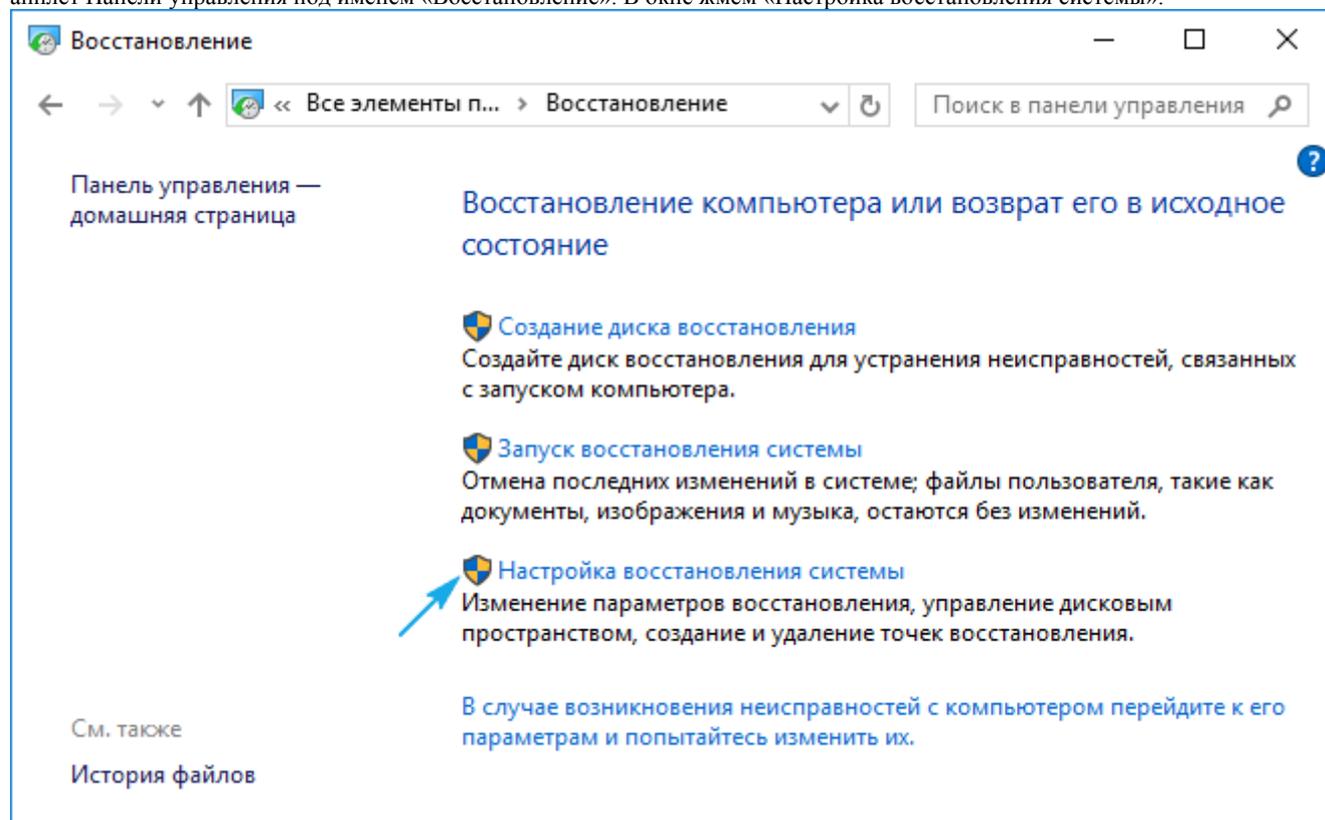
5. Определяемся с местом хранения снимка операционной системы и разделами, которые будут подвергаться резервированию (лучше всего указать съемный накопитель).



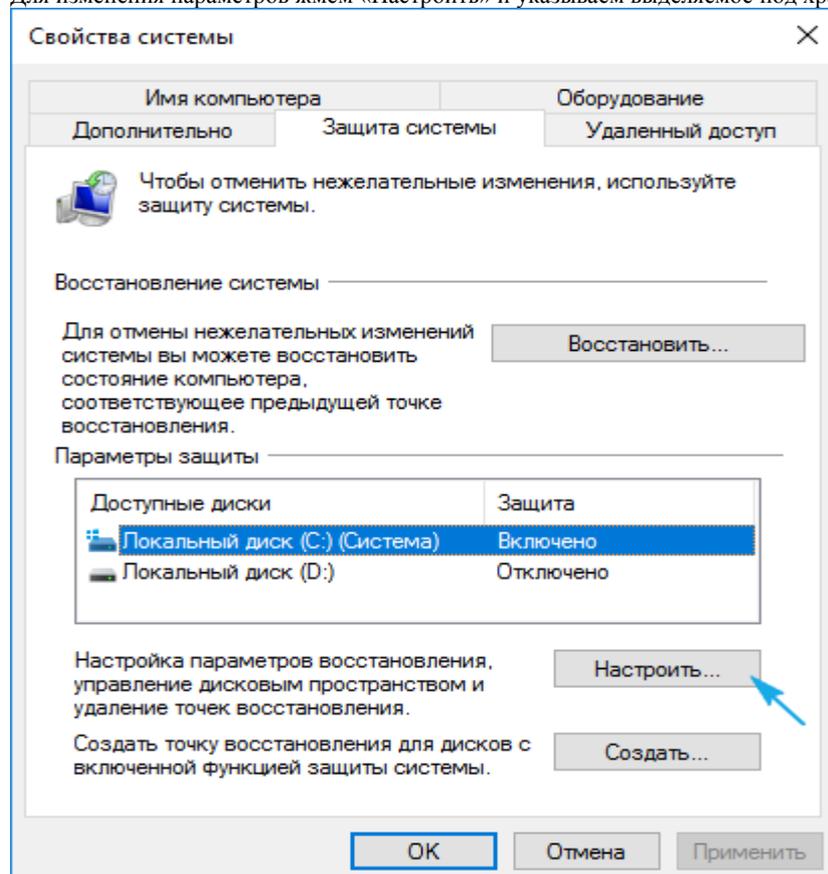
После завершения сжатия системных файлов и их перенесения на указанный цифровой носитель его можно будет использовать для быстрого возврата Windows 10 к запечатленному в образе состоянию. Для того чтобы запустить восстановление с образа необходимо выполнить загрузку компьютера с флешки, на которой файл хранится, или через инсталлятор Windows 10 («Диагностика» — «Расширенные параметры» — «Восстановление образа ОС»).

Точки отката Windows 10

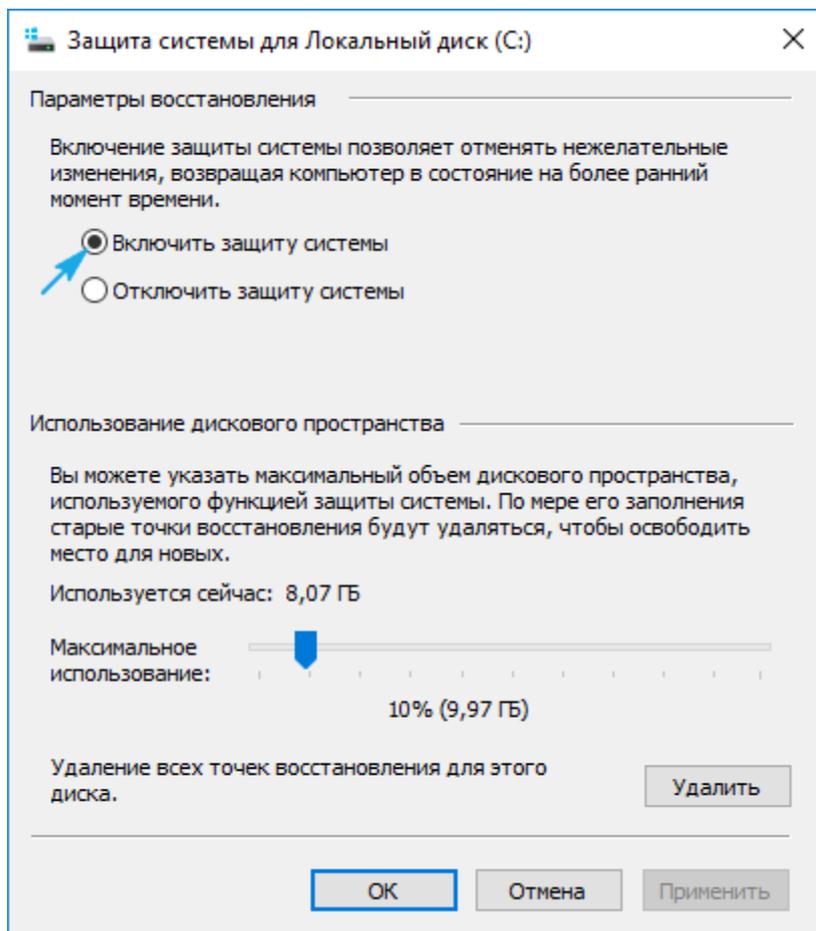
С этой функцией нет никаких новшеств, все ее возможности работают, как в предыдущих версиях ОС. Она предоставляет шанс вернуть систему к одному из сохранившихся состояний через среду восстановления или в работающей операционной системе. Чтобы воспользоваться всеми преимуществами функции, она должна быть активированной. Проверить состояние можно через апплет Панели управления под именем «Восстановление». В окне жмем «Настройка восстановления системы».



Для изменения параметров ждем «Настроить» и указываем выделяемое под хранение точек отката место на системно диске.



При использовании лицензионного образа эта функция активирована, но, если установили Windows 10, скачанную с пиратских сайтов, возможно, автор сборки деактивировал эту функцию. Как включить восстановление системы? Выбираем системный раздел, ждем «Настроить» и перемещаем чекбокс к позиции «Включить защиту системы».



Обычно, точки отката формируются автоматически в случаях внесения со стороны пользователя или каких-либо приложений изменений, касающихся системных файлов, настроек, служб и параметров реестра. Также доступна возможность ручного создания точек восстановления. В окне «Свойства системы» нажимаем «Создать» и вводим название снимка, чтобы было проще идентифицировать его.

Для эксплуатации функции отката системы посредством одной из точек восстановления заходим в тот же апплет и жмем «Запуск восстановления системы». В случае когда Windows 10 не запускается, выполняем загрузку с диска восстановления или установочного дистрибутива и вызываем «Восстановление системы» через «Дополнительные параметры» в окне диагностики.

История файлов
 Очередное новшество Windows 10, позволяющее делать и хранить резервные копии указанных файлов (зачастую текстовых документов и различных проектов) и извлекать из резерва нужную копию файла в случае необходимости. Как можно увидеть, «десятка» обладает расширенным набором эффективных функций для возврата системы к работоспособному состоянию в любых случаях и без переустановки ОС. В добавок ко всем перечисленным инструментам обязательно следует отнести функцию восстановления загрузчика средствами Windows 10.

**Текущий контроль
 бсеместр
 Лабораторная работа
 форма текущего контроля**

по теме: Сбор информации об ошибках. Формирование отчетов об ошибках

Цель: научиться собирать информацию об ошибках и формировать отчеты

По завершению практического занятия студент должен уметь: собирать информацию об ошибках и формировать отчеты

Продолжительность: 3 аудиторных часа (135минут)

Необходимые принадлежности

Персональный компьютер, программное обеспечение: среда PowerDesigner.

Задание

Отчет об ошибках содержит следующую информацию:

Примечание: Для некоторых ошибок может быть получена не вся информация.

МЕТКА	Предопределенное название события.
ИД	Числовой идентификатор события.
Дата/Время	Дата и время события.
Порядковый номер	Уникальный номер события.
ИД системы	Идентификатор системного блока.
ИД узла	Мнемоническое имя системы.
Класс	Общий источник ошибки. Существуют следующие классы ошибок:

	H	Аппаратного обеспечения. (При получении сообщения об ошибке аппаратного обеспечения обратитесь к руководству оператора системы за инструкциями по проведению диагностики отказавшего устройства или другого оборудования. Диагностическая программа определяет состояние устройства, проверяя устройство и анализируя связанные с ним записи протокола ошибок.)
	S	Программного обеспечения.
	O	Информационные сообщения.
	U	Неопределенные (например, сбой сети).
Тип		Серьезность обнаруженной ошибки. Существует пять типов ошибок: PEND Устройство или компонент может стать недоступным. PERF Производительность устройства или компонента понизилась ниже допустимого уровня. PERM Неисправимая ошибка. Этот тип относится к самым серьезным ошибкам и свидетельствует о неисправности устройства или модуля программного обеспечения. Ошибки всех типов, кроме PERM, обычно не означают неисправности, но записываются для анализа в диагностических программах. TEMP Ошибка, которая была исправлена после нескольких неудачных попыток. Этот тип ошибки также применяется для записи информационных сообщений, например, статистики передачи данных устройств DASD. UNKN Невозможно определить серьезность ошибки. INFO Запись протокола ошибок носит информационный характер и не свидетельствует об ошибке.
Имя ресурса		Имя ресурса, обнаружившего ошибку. В случае ошибки программного обеспечения, означает имя компонента программного обеспечения или программы. В случае ошибки аппаратного обеспечения - имя устройства или компонента системы. Это не означает, что компонент неисправен и требует замены. Это значение лишь определяет модуль диагностики, применяемый для анализа ошибки.
Класс ресурса		Общий класс ресурса, обнаружившего ошибку (например, класс устройства дисковый накопитель).
Тип ресурса		Тип ресурса, обнаружившего ошибку (например, тип устройства 355mb).
Код расположения		Путь к устройству. Может содержать до четырех полей, соответствующих корпусу, разъему, кабелю и порту.
VPD		Сведения о продукте. В этом поле может быть указана различная информация. Запись об устройстве в протоколе ошибок обычно содержит информацию о производителе устройства, серийном номере, Уровнях конструкторских изменений и уровнях ПЗУ.
Описание		Краткое описание ошибки.
Возможная причина		Список возможных источников ошибки.
Ошибки пользователя		Список возможных ошибок пользователя, вызвавших сбой. Примером таких ошибок являются неправильно вставленные диски или внешние устройства (такие как модемы и принтеры), питание которых отключено.
Рекомендуемые действия		Инструкции по устранению ошибок, вызванных пользователем.
Ошибка установки		Список возможных ошибок при установке и настройке, вызвавших сбой. Примерами такого типа ошибки являются несовместимость программного и аппаратного обеспечения, неправильное подключение кабелей или их отсоединение, а также неправильно настроенные системы.
Рекомендуемые действия		Инструкции по устранению ошибок, вызванных неправильной установкой.
Возможный сбой		Список возможных неполадок программного и аппаратного обеспечения. Примечание: Раздел протокола ошибок "возможный сбой" обычно свидетельствует о неполадке программного обеспечения. Если же в протоколе есть записи об ошибке пользователя или установке, но нет записи о возможном сбое, то это обычно означает, что программное обеспечение не является причиной неполадки.
Рекомендуемые действия		Если вы считаете, что причиной является ошибка программного обеспечения или вам не удастся исправить ошибку пользователя или установки, сообщите о неполадке в отдел по обслуживанию программного обеспечения. Инструкции по устранению сбоя. В случае ошибок аппаратного обеспечения список рекомендуемых действий содержит запись ВЫПОЛНИТЕ ПРОЦЕДУРУ ЛОКАЛИЗАЦИИ НЕПОЛАДКИ . Это значит, что необходимо запустить диагностическую программу.

Подробные сведения

Уникальные для каждой записи протокола ошибок данные об ошибке, например, код ошибки устройства.

Некоторые ошибки можно исключить из отчета. Для просмотра ошибок, исключенных из отчета, введите команду:

```
errpt -t -F report=0 | pg
```

Если такие ошибки есть, включите в отчет все ошибки с помощью команды **errupdate**.

Некоторые ошибки могут не регистрироваться в протоколе. Для просмотра ошибок, исключенных из протокола, введите команду:

```
errpt -t -F log=0 | pg
```

Если такие ошибки есть, включите регистрацию в протоколе для всех ошибок с помощью команды **errupdate**. Регистрация всех ошибок в протоколе необходима для воссоздания ошибки системы.

Примеры подробных отчетов об ошибках

Ниже приведен пример записей отчета об ошибках, созданного с помощью команды **errpt -a**.

Класс ошибки **H** и тип ошибки **PERM** означают, что в системе была обнаружена ошибка устройства (драйвера адаптера SCSI), которую не удалось устранить.

С этим типом ошибки могут быть связаны данные диагностики.

Эта информация находится в конце сообщения об ошибке.

МЕТКА: SCSI_ERR1

ИД: 0502F666

Дата/Время: Jun 19 22:29:51

Порядковый номер: 95

ИД системы: 123456789012

ИД узла: host1

Класс: H

Тип: PERM

Имя ресурса: scsi0

Класс ресурса: adapter

Тип ресурса: hscsi

Расположение: 00-08

VPD:

Device Driver Level.....00

Diagnostic Level.....00

Displayable Message.....SCSI

EC Level.....C25928

FRU Number.....30F8834

Manufacturer.....IBM97F

Part Number.....59F4566

Serial Number.....00002849

ROS Level and ID.....24

Read/Write Register Ptr....0120

Описание ADAPTER ERROR

Возможные причины

ADAPTER HARDWARE CABLE

CABLE TERMINATOR DEVICE

Возможные сбои ADAPTER

CABLE LOOSE OR DEFECTIVE

Рекомендуемые действия

PERFORM PROBLEM DETERMINATION PROCEDURES

CHECK CABLE AND ITS CONNECTIONS

Подробные сведения

SENSE DATA

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

Порядковый номер протокола диагностики: 153

Проверенный ресурс: scsi0

Описание ресурса: SCSI I/O Controller

Расположение: 00-08

SRN: 889-191

Описание: Анализ

протокола ошибок указывает на неполадку аппаратного обеспечения.

Возможные FRU:

Шина SCSI FRU: нет 00-08

Вентилятор

SCSI2 FRU: 30F8834 00-08

Контроллер ввода-вывода SCSI

Класс ошибки **H** и тип ошибки **PEND** означают, что устройство (Token Ring) может в ближайшее время стать недоступным из-за большого количества ошибок, обнаруженных системой.

МЕТКА: TOK_ESERR

ИД: AF1621E8

Дата/Время: Jun 20 11:28:11

Порядковый номер: 17262

ИД системы: 123456789012

ИД узла: host1

Класс: H

Тип: PEND

Имя ресурса: TokenRing

Класс ресурса: tok0

Тип ресурса: Adapter
Расположение TokenRing

Описание
EXCESSIVE TOKEN-RING ERRORS

Возможные причины
TOKEN-RING FAULT DOMAIN

Возможные сбои TOKEN-RING FAULT DOMAIN

Рекомендуемые действия
REVIEW LINK CONFIGURATION DETAIL DATA
CONTACT TOKEN-RING ADMINISTRATOR RESPONSIBLE FOR THIS LAN

Подробные сведения
SENSE DATA
0ACA 0032 A440 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 2080 0000 0000 0010 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 78CC 0000 0000 0005 C88F 0304 F4E0 0000 1000 5A4F 5685
1000 5A4F 5685 3030 3030 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000

Класс ошибки **S** и тип ошибки **PERM** означают, что в системе была обнаружена ошибка программного обеспечения, которую не удалось устранить.

МЕТКА: DSI_PROC
ИД: 20FAED7F

Дата/Время: Jun 28 23:40:14
Порядковый номер: 20136
ИД системы: 123456789012
ИД узла: 123456789012
Класс: S
Тип: PERM
Имя ресурса: SYSVMM

Описание
Data Storage Interrupt, Processor

Возможные причины
SOFTWARE PROGRAM

Возможные сбои SOFTWARE PROGRAM

Рекомендуемые действия
IF PROBLEM PERSISTS THEN DO THE FOLLOWING
CONTACT APPROPRIATE SERVICE REPRESENTATIVE

Подробные сведения
Data Storage Interrupt Status Register
4000 0000
Data Storage Interrupt Address Register
0000 9112
Segment Register, SEGREG
D000 1018
EXVAL
0000 0005

Класс ошибки **S** и тип ошибки **TEMP** означают, что в системе была обнаружена ошибка программного обеспечения. После нескольких попыток системе удалось устранить неполадку.

МЕТКА: SCSI_ERR6
ИД: 52DB7218

Дата/Время: Jun 28 23:21:11
Порядковый номер: 20114
ИД системы: 123456789012
ИД узла: host1
Класс: S
Тип: INFO
Имя ресурса: scsi0

Описание
SOFTWARE PROGRAM ERROR

Возможные причины

SOFTWARE PROGRAM

Возможные сбои SOFTWARE PROGRAM

Рекомендуемые действия
IF PROBLEM PERSISTS THEN DO THE FOLLOWING
CONTACT APPROPRIATE SERVICE REPRESENTATIVE

Подробные сведения

SENSE DATA

```
0000 0000 0000 0000 0000 0011 0000 0008 000E 0900 0000 0000 FFFF
FFFE 4000 1C1F 01A9 09C4 0000 000F 0000 0000 0000 0000 FFFF FFFF
0325 0018 0040 1500 0000 0000 0000 0000 0000 0000 0000 0800
0000 0100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000
```

Класс ошибки **O** означает информационное сообщение.

МЕТКА: OPMSG

ИД: AA8AB241

Дата/Время: Jul 16 03:02:02

Порядковый номер: 26042

ИД системы: 123456789012

ИД узла: host1

Класс: O

Тип: INFO

Имя ресурса: OPERATOR

Описание

OPERATOR NOTIFICATION

Ошибки пользователя

errlogger COMMAND

Рекомендуемые действия
REVIEW DETAILED DATA

Подробные сведения

MESSAGE FROM errlogger COMMAND

hdisk1: Анализ протокола ошибок указывает на неполадку аппаратного обеспечения.

Пример краткого отчета об ошибках

Ниже приведен пример краткого отчета об ошибках, созданного с помощью команды **errpt**. Каждой записи об ошибке соответствует одна строка информации.

ERROR_

ИДЕНТИФИКАТОР СИСТЕМНОЕ_ВРЕМЯ Т КЛ ИМЯ_РЕСУРСА ОПИСАНИЕ_ОШИБКИ

192AC071 0101000070 I O errdemon Ведение протокола ошибок выключено

0E017ED1 0405131090 P H mem2 Сбой памяти

9DVCFDDEE 0101000070 I O errdemon Ведение протокола ошибок включено

038F2580 0405131090 U H scdisk0 НЕОПРЕДЕЛЕННАЯ ОШИБКА

AA8AB241 0405130990 I O OPERATOR ИЗВЕЩЕНИЕ ОПЕРАТОРА

Создание отчета об ошибках

Выполните следующие действия, чтобы создать отчет об ошибках программного обеспечения или неполадках аппаратного обеспечения.

1. Определите, включено ли ведение протокола ошибок. Для этого проверьте, содержит ли протокол ошибок записи:
2. `errpt -a`

Команда **errpt** создает отчет об ошибках из записей системного протокола ошибок.

Если протокол ошибок пуст, ведение протокола ошибок было отключено. Активизируйте средство ведения протокола ошибок с помощью следующей команды:

```
/usr/lib/errdemon
```

Примечание: Для запуска этой команды необходимы права доступа пользователя root.

Демон **errdemon** запускает ведение протокола ошибок. Если демон не работает, протокол ошибок не ведется.

3. Создайте отчет об ошибках с помощью команды **errpt**. Например, для просмотра всех ошибок дискового накопителя `hdisk1` введите команду:
4. `errpt -N hdisk1`
5. Создайте отчет об ошибках с помощью SMIT. Например, с помощью команды **smit errpt**:
6. `smit errpt`

Выберите **1**, чтобы направить отчет об ошибках в стандартный вывод, или **2**, чтобы отправить отчет на принтер.

Выберите **yes**, чтобы просматривать или распечатывать записи протокола ошибок по мере их добавления, в противном случае выберите **no**.

Укажите нужное имя устройства в опции **Выбрать имена ресурсов** (например `hdisk1`).

Выберите **Do**.

Завершение ведения протокола ошибок

В данном разделе описано завершение работы средства ведения протокола ошибок. Как правило, нет необходимости отключать средство ведения протокола ошибок. Вместо этого следует удалить из протокола ошибок старые и ненужные записи. Инструкции по очистке протокола ошибок приведены в разделе Очистка протокола ошибок.

Средство ведения протокола ошибок следует отключать при установке или проверке нового программного или аппаратного обеспечения. В этом случае демон ведения протокола ошибок не будет отнимать время центрального процессора на регистрацию известных вам ошибок.

Примечание: Для запуска применяемой в этой процедуре команды у вас должны быть права доступа пользователя root.

Введите команду **errstop**, чтобы отключить ведение протокола ошибок:

```
errstop
```

Команда **errstop** завершает работу демона ведения протокола.

Очистка протокола ошибок

Этот раздел содержит информацию по удалению из протокола ошибок старых и ненужных записей. Обычно очистка протокола автоматически выполняется ежедневно с помощью команды **cron**.

Если эта процедура не выполняется автоматически, следует время от времени очищать протокол ошибок вручную, предварительно проверив его на наличие записей о серьезных неполадках.

Кроме того, можно удалить записи о конкретных ошибках. Например, после замены дискового накопителя можно удалить из протокола ошибок записи об ошибках старого дискового накопителя.

Для удаления всех записей протокола ошибок выполните одно из следующих действий:

- Вызовите команду **errclear-d**. Например, для удаления всех записей об ошибках программного обеспечения, введите команду:

```
errclear -d S 0
```

Команда **errclear** удаляет из протокола ошибок записи, внесенные раньше определенного числа дней. В предыдущем для удаления всех записей указано значение 0.
- Введите команду **smit errclear**:
- ```
smit errclear
```

#### **Копирование протокола ошибок на дискету или магнитную ленту**

Выполните следующие действия, чтобы скопировать протокол ошибок:

- С помощью команд **ls** и **backup** скопируйте протокол ошибок на дискету. Вставьте отформатированную дискету в дисковод и введите команду:  

```
ls /var/adm/ras/errlog | backup -ivp
```
- Для копирования протокола ошибок на магнитную ленту вставьте магнитную ленту в лентопротяжное устройство и введите команду:  

```
ls /var/adm/ras/errlog | backup -ivpf/dev/rmt0
```

ИЛИ
- С помощью команды **snap** соберите информацию о конфигурации системы в файл **tar** и скопируйте его на дискету. Вставьте отформатированную дискету в дисковод и введите команду:

**Примечание:** Для запуска команды **snap** у вас должны быть права доступа пользователя root.

```
snap -a -o
/dev/rfd0
```

В этом примере для сбора всей информации о конфигурации системы в команде **snap** указан флаг **-a**. Флаг **-o** позволяет скопировать сжатый файл **tar** на указанное устройство. **/dev/rfd0** указывает дисковод.

Введите следующую команду, чтобы собрать всю информацию о конфигурации в файле **tar** и скопировать его на магнитную ленту:

```
snap -a -o /dev/rmt0
```

**/dev/rmt0** указывает лентопротяжное устройство.

### **Текущий контроль бсеместр**

#### **Лабораторная работа** *форма текущего контроля*

**по теме:** Выявление и устранение ошибок программного кода информационных систем

**Цель:** выявлять и устранять ошибки программного кода информационных систем

**По завершению практического занятия студент должен уметь:** выявлять и устранять ошибки программного кода информационных систем

Продолжительность: 4 аудиторных часа (180 минут)

#### **Необходимые принадлежности**

Персональный компьютер, программное обеспечение: среда PowerDesigner.

#### **Задание**

Системы моделирования обладают множеством достоинств. Обычно в их составе имеются отладчики и средства вывода информации на печать, однако системы моделирования это всего лишь имитаторы. Отлаживаемая программа может успешно исполняться в системе моделирования и быть полностью неработоспособной в реальных условиях. Так что системы моделирования это лишь частичное решение. Ошибки программного обеспечения вполне могут пройти мимо системы моделирования и всплыть в реальном оборудовании.

Именно в этом и скрыта главная проблема: как показано на рис. 1, исправление ошибок, которые выявляются не на этапе тестирования, а в процессе использования, обходится значительно дороже. Если ошибка найдена в программе для не-встраиваемых систем, то можно выпустить обновленную версию программы с исправлениями, стоимость таких обновлений, как правило, сравнительно невысокая. Если же ошибка найдена во встроенной системе, то для ее исправления необходим возврат и модификация самих устройств с этой системой. Стоимость такого возврата может достигать астрономических величин, и стать причиной разорения компаний.

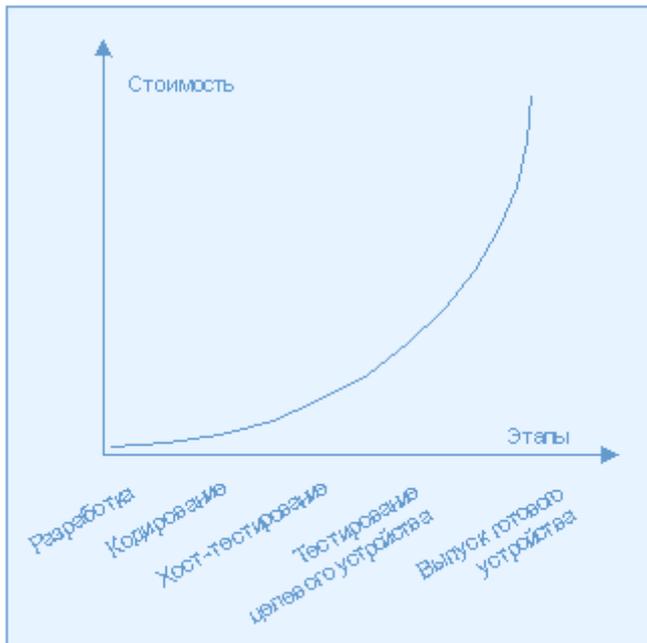


Рис. 1. Стоимость устранения ошибок во встраиваемых системах

На мой взгляд, сроки и затраты на выявление и устранение ошибок для встраиваемых систем приблизительно удваиваются (из-за описанных выше трудностей). В свете таких немислимых затрат любой метод, который изначально будет препятствовать появлению ошибок, имеет неоценимое значение. К счастью для разработчиков встраиваемых систем, для предотвращения ошибок можно использовать некоторые из новых технологий программной разработки. Наиболее рекомендуемые две из них: стандарты программирования и блочное тестирование.

Правда, оба этих метода сегодня не столько применяются, сколько прославляются. Практически каждый разработчик программного обеспечения согласен с их высокой ценностью, но пользуются ими единицы. Подобная непоследовательность объясняется в большинстве случаев двумя причинами. Прежде всего, многие считают следование стандартам программирования и блочное тестирование весьма утомительным делом. Учитывая, сколько времени и сил эти подходы позволяют сэкономить в будущем, разработчикам следовало бы немножко потерпеть и избежать огромных трудозатрат (и возможного отказа от проекта) впоследствии.

Разработчикам систем реального времени еще труднее они в дополнение ко всему должны решать проблемы, связанные с соблюдением различных временных зависимостей. В конце статьи мы рассмотрим трудности, возникающие при отладке систем реального времени, и познакомимся с некоторыми методами отладки, которые рассчитаны на преодоление этих трудностей и которые также могут быть использованы при разработке любого программного обеспечения.

**Способы отладки программ** Отладка программ заключается в проверке правильности работы программы и аппаратуры. Программа, не содержащая синтаксических ошибок тем не менее может содержать логические ошибки, не позволяющие программе выполнять заложенные в ней функции. Логические ошибки могут быть связаны с алгоритмом программы или с неправильным пониманием работы аппаратуры, подключенной к портам микроконтроллера. Встроенный в состав интегрированной среды программирования отладчик позволяет отладить те участки кода программы, которые не зависят от работы аппаратуры, не входящей в состав микросхемы микроконтроллера. Обычно это относится к вычислению математических выражений или преобразованию форматов представления данных. Для отладки программ обычно применяют три способа:

1. Пошаговая отладка программ с заходом в подпрограммы;
2. Пошаговая отладка программ с выполнением подпрограммы как одного оператора;
3. Выполнение программы до точки останова.

Пошаговая отладка программ заключается в том, что выполняется один оператор программы и, затем контролируются те переменные, на которые должен был воздействовать данный оператор. Если в программе имеются уже отлаженные подпрограммы, то подпрограмму можно рассматривать, как один оператор программы и воспользоваться вторым способом отладки программ. Если в программе существует достаточно большой участок программы, уже отлаженный ранее, то его можно выполнить, не контролируя переменные, на которые он воздействует. Использование точек останова позволяет пропускать уже отлаженную часть программы. Точка останова устанавливается в местах, где необходимо проверить содержимое переменных или просто проконтролировать, передается ли управление данному оператору. Практически во всех отладчиках поддерживается это свойство (а также выполнение программы до курсора и выход из подпрограммы). Затем отладка программы продолжается в пошаговом режиме с контролем локальных и глобальных переменных, а также внутренних регистров микроконтроллера и напряжений на выводах этой микросхемы.

#### **Следуйте стандартам программирования!**

Самый лучший способ повысить качество ПО это стараться не допускать ошибок в процессе ввода исходного текста.

Первый шаг на пути предотвращения ошибок это осознание того, что ошибки действительно можно предотвратить. Больше всего препятствует контролю над ошибками распространенное убеждение в том, что ошибки неизбежны. Это заблуждение! Ошибки сами по себе не появляются их вносит в текст разработчик. Человеку свойственно ошибаться, так что даже самые лучшие программисты время от времени допускают ошибки, если у них есть такая возможность. Поэтому чтобы уменьшить число ошибок, надо сократить возможности их появления. Один из лучших способов здесь следование стандартам программирования, что ликвидирует благодатную почву для возникновения ошибок на первых этапах.

Стандарты программирования это специфичные для языка "правила", которые, если их соблюдать, значительно снижают вероятность внесения ошибок в процессе разработки приложения. Следовать стандартам программирования нужно на этапе написания программ, до их переноса в целевые платформы, при этом стандартизация должна существовать для всех языков. Поскольку большая часть разработ-

чиков встраиваемых систем пользуется языком C, больше внимания уделим именно стандартам программирования на C, хотя такие же стандарты существуют и для других языков, включая C++ и Java.

Как правило, стандарты программирования делятся на две категории:

- отраслевые стандарты программирования: правила, принятые всеми программистами на данном языке (например, запрет входа в цикл не через его заголовок).
- специальные стандарты программирования: правила, соблюдаемые конкретной группой разработчиков, в рамках конкретного проекта, или даже единственным программистом. Существует три типа специальных стандартов, которыми может воспользоваться разработчик встраиваемой программной системы: внутренние стандарты, персональные стандарты и стандарты, определяемые целевой платформой.

Внутренние стандарты программирования это правила, которые специфичны для вашей организации или группы разработчиков. Так, уникальные для организации правила присвоения имен это пример внутренних стандартов программирования.

Персональные стандарты это правила, которые помогут вам избежать ваших наиболее частых ошибок. Каждый раз при появлении какой-либо ошибки программист должен проанализировать причину ее появления и выработать собственное правило, препятствующее повторному ее возникновению. Например, если в операторе условия вы часто пишете знак присваивания вместо знака проверки на равенство (т.е. "if (a=b)" вместо "if (a==b)"), то вам необходимо создать для себя следующий стандарт: "Остерегаться применения знака присваивания в операторе проверки условия".

Стандарты, определяемые целевой платформой, это правила, нарушение которых в данной платформе может привести к появлению определенных проблем. Например, такими стандартами могут быть ограничения на использование памяти или размер переменных, налагаемые целевой платформой.

Чтобы лучше разобраться в том, что такое стандарты программирования и как они работают, познакомимся с ними на конкретных примерах. Рассмотрим следующую запись на языке C:

```
char *substring (char string[80], int start_pos, int length)
{
.
.
.
}
```

Здесь размер одномерного массива декларируется в списке аргументов функции. Это опасная конструкция, поскольку в языке C аргумент-массив передается как указатель на его первый элемент, и в разных обращениях к функции в числе ее фактических аргументов могут указываться массивы с разной размерностью. Создав такую конструкцию, вы предполагаете пользоваться буфером фиксированного размера на 80 элементов, считая, что именно такой буфер и будет передаваться функции, а это может привести к разрушению памяти. Если бы автор этого оператора следовал стандарту программирования "не объявлять размер одномерного массива в числе аргументов функции" (взятому из набора стандартов программирования на языке C одной из ведущих телекоммуникационных компаний), то этот текст выглядел бы следующим образом и проблем с разрушением памяти удалось бы избежать:

```
char *substring (char string[], int start_pos, int length)
{
.
.
.
}
```

Стандарты программирования позволяют также избегать проблем, которые до момента портирования кода на другую платформу могут не проявляться. Например, следующий кусок кода будет исправно работать на одних платформах и породить ошибки после переноса его на другие платформы:

```
#include
void test(char c) {
if(a <= c && c <= z) { // Неправильно
}
if(islower(c)) { // Правильно
}
while (A <= c && c <= Z) { // Неправильно
}
while (isupper(c)) { // Правильно
}
}
```

Проблемы портации могут быть связаны с символьными тестами, в которых не используется функции ctype.h (isalnum, isalpha, iscntrl,

isdigit, isgraph, islower, isprint, ispunct, isspace, isupper, isxdigit, tolower, toupper). Функции ctype.h для символьной проверки и преобразования прописных букв в строчные и наоборот работают с самыми разными наборами символов, обычно очень эффективны и гарантируют международную применимость программного продукта.

Лучший способ внедрить эти и другие стандарты программирования это обеспечить их автоматическое применение в составе какой-либо технологии программирования, вместе с набором целенаправленных отраслевых стандартов и механизмами создания и поддержки стандартов программирования, ориентированных на конкретную систему. При выборе подобной технологии необходимо сначала найти ответы на вопросы, среди которых следующие:

- Применима ли она к данной программе и/или компилятору?
- Содержит ли она набор отраслевых стандартов программирования?
- Позволяет ли она создавать и поддерживать специальные стандарты программирования (включая стандарты, определяемые целевой платформой)?
- Легко ли структурировать отчеты в соответствии с вашими групповыми и проектными приоритетами?
- Насколько легко она интегрируется в существующий процесс разработки?

### Блочное тестирование

Зачастую, слыша о блочном тестировании, разработчики воспринимают его как синоним модульного тестирования. Другими словами, проверяя отдельный модуль или подпрограмму более крупной программной единицы, разработчики считают, что выполняют блочное тестирование. Конечно, модульное тестирование имеет очень большое значение и, безусловно, должно проводиться, но это не тот метод, на котором я бы хотел остановиться. Говоря о "блочном тестировании", я имею в виду тестирование на еще более низком уровне: тестирование самых минимально возможных программных единиц, из которых состоит прикладная программа, всё ещё находясь в инструментальной среде (хост-системе) в случае языка C, это будут функции, которые проверяются сразу же после их компиляции.

Блочное тестирование значительно повышает качество программного обеспечения и эффективность процесса разработки. При тестировании на уровне объектов вы гораздо ближе к этой методике и обладаете гораздо большими возможностями построения входных наборов, выявляющих ошибки со стопроцентным покрытием (рис. 2). Кроме того, тестируя блок кода сразу после того, как он был написан, вы тем самым избегаете необходимости "продираться" через наложения ошибок, чтобы найти и исправить единственную исходную в данном случае вы сразу ее устраняете, и вся проблема решена. Это существенно ускоряет и облегчает процесс разработки, поскольку на поиск и устранение ошибок тратится значительно меньше материальных и временных ресурсов.

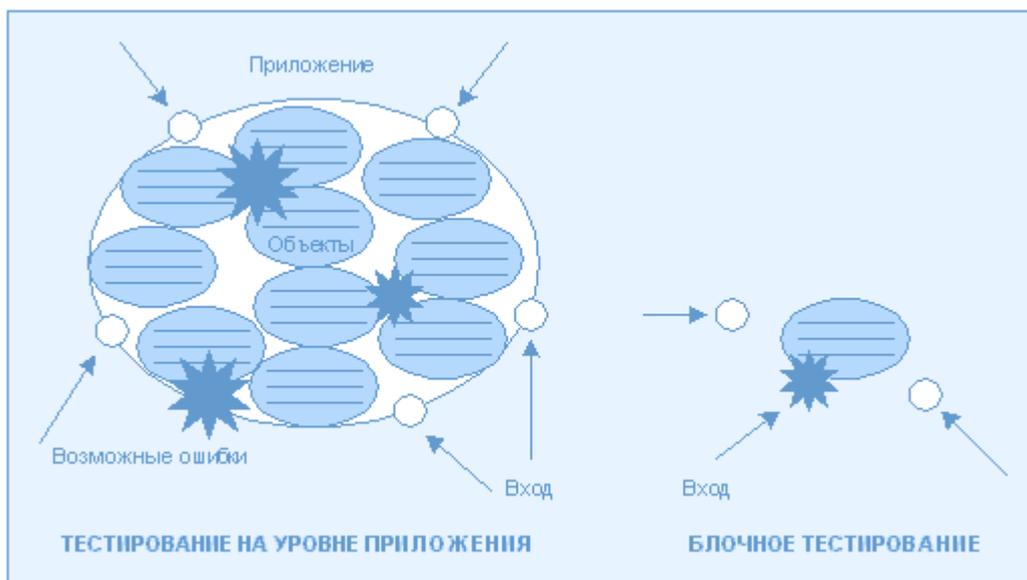


Рис. 2. Простота нахождения ошибок при блочном тестировании

блочном тестировании

Блочное тестирование можно разделить минимум на два отдельных процесса. Первый это тестирование "черного ящика", или процесс определения функциональных проблем. На уровне отдельных блоков тестирование "черного ящика" заключается в проверке функциональных характеристик посредством определения степени соответствия параметров открытого интерфейса функции ее спецификации; подобная проверка выполняется без учета способов реализации. Результатом тестирования "черного ящика" на блочном уровне является уверенность в том, что данная функция ведет себя в полном соответствии с определением и что незначительная функциональная ошибка не приведет к лавине трудноразрешимых проблем.

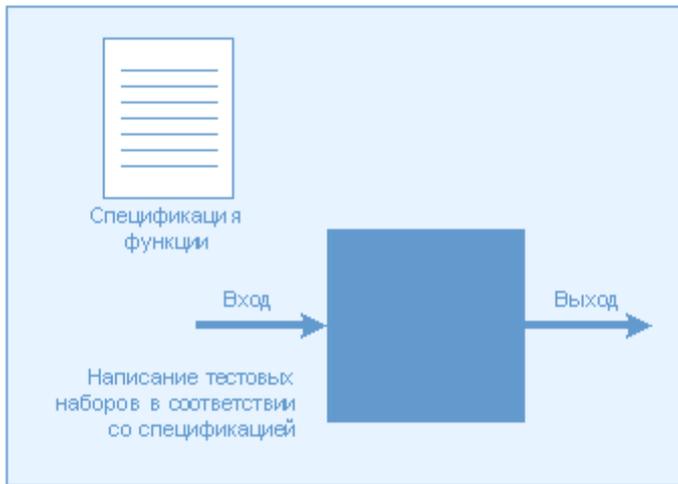


Рис. 3. Тестирование "черного ящика"

Второй процесс называется тестированием "белого ящика" и предназначен для выявления конструктивных недостатков. На уровне отдельных блоков проверяется, не произойдет ли крах всей программы при передаче в функцию неожиданных ею параметров. Этот вид тестирования должен проводиться специалистом, имеющим полное представление о способе реализации проверяемой функции. После такой проверки можно быть уверенным в том, что приводящих к краху системы ошибок нет и что функция будет устойчиво работать в любых условиях (т.е. выдавать предсказуемые результаты даже при вводе непредвиденных входных параметров).

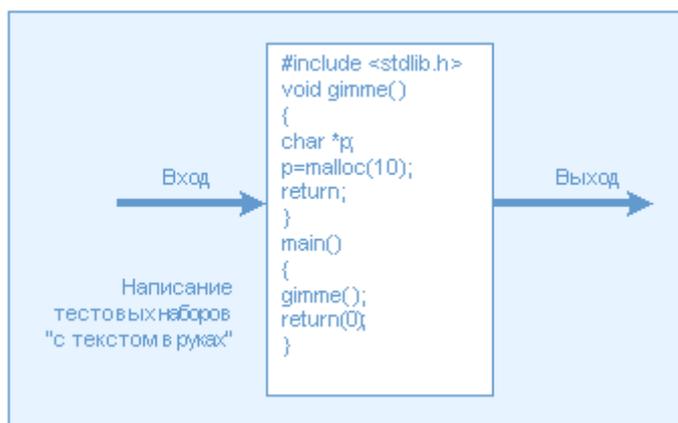


Рис. 4. Тестирование "белого ящика"

Оба вышеописанных процесса могут служить основой третьего, регрессивного тестирования. Сохранив тестовые наборы для "черного" и "белого" ящиков, можно использовать их для регрессивного тестирования на уровне блоков и контролировать целостность кода по мере того, как вы его модифицируете. Концепция регрессивного тестирования на этом уровне является новым оригинальным подходом. При выполнении регрессивного тестирования на уровне блоков можно сразу же после изменения вами текста определять, не появились ли новые проблемы, и устранять их немедленно после возникновения, тем самым препятствуя распространению ошибки по вышележащим уровням.

Главная проблема, связанная с блочным тестированием, заключается в том, что если не пользоваться технологиями автоматического блочного тестирования, проводить его трудно, утомительно и слишком долго. Рассмотрим вкратце причины, по которым включать неавтоматизированное блочное тестирование в сегодняшние процессы разработки трудно (если вообще возможно).

Первый этап блочного тестирования ПО встраиваемых систем заключается в создании такой среды, которая позволит запускать и тестировать интересующую функцию в хост-системе. Это требует выполнения следующих двух действий:

- разработка программного текста, который будет запускать функцию,
- написание фиктивных модулей, которые будут возвращать результаты вместо внешних ресурсов, к которым обращается функция и которые в текущий момент отсутствуют или недоступны.

Второй этап разработка тестовых наборов. Для полноты охвата конструктивных и функциональных особенностей функции необходимо создавать тестовые наборы двух типов: для "черного ящика" и для "белого ящика".

Основой разработки тестовых наборов для "черного ящика" должна стать спецификация функции. В частности, для каждой записи в спецификации должен быть создан хотя бы один тестовый набор, при этом желательно, чтобы эти наборы учитывали указанные в спецификации граничные условия. Проверять того, что некоторые входные параметры приводят к ожидаемым результатам, недостаточно; необходимо определить диапазон взаимосвязей входов и выходов, который позволит сделать вывод о корректной реализации указанных функциональных характеристик, и затем создавать тестовые наборы, полностью покрывающие данный диапазон. Можно также тестировать не указанные в спецификации и ошибочные условия.

Цель тестовых наборов для "белого ящика" обнаружить все скрытые дефекты путем всестороннего тестирования функции разнообразными входными параметрами. Эти наборы должны обладать следующими возможностями:

- обеспечивать максимально возможное (100%) покрытие функции: как уже говорилось, такая степень покрытия на уровне блоков возможна, поскольку создавать наборы для тестирования каждой характеристики функции вне приложения гораздо

проще (стопроцентное покрытие во всех случаях невозможно, но это цель, к которой надо стремиться);

- выявлять условия краха функции.

Следует заметить, что самостоятельное создание подобных наборов, не владея технологиями их построения, невероятно тяжелое занятие. Чтобы создать эффективные тестовые наборы для "белого ящика", необходимо сначала получить полное представление о внутренней структуре функции, написать наборы, обеспечивающие максимальное покрытие функции, и найти совокупность входов, приводящих к отказу функции. Получить спектр покрытия, необходимый для высокоэффективного тестирования "белого ящика", возможно лишь при исследовании значительного числа путей прохода по функции. Например, в обычной программе, состоящей из 10000 операторов, имеется приблизительно сто миллионов возможных путей прохода; вручную создать тестовые наборы для проверки всех этих путей невозможно.

После создания тестовых наборов необходимо провести тестирование функции в полном объеме и проанализировать результаты с целью выявления причин ошибок, крахов и слабых мест. Необходимо иметь способ прогона всех тестовых наборов и быстрого определения, какие из них приводят к возникновению проблем. Необходимо также иметь инструмент измерения степени покрытия для оценки полноты тестирования функции и определения необходимости в дополнительных тестовых наборах.

При любых изменениях функции следует проводить регрессивное тестирование, чтобы убедиться в отсутствии новых и/или устранении предыдущих ошибок. Включение блочного регрессивного тестирования в процесс разработки позволит защититься от многих ошибок, они будут обнаружены сразу же после возникновения и не смогут стать причинами распространения ошибок в приложении.

Регрессивное тестирование можно проводить двумя способами. Первый заключается в том, что разработчик или испытатель анализирует каждый тестовый набор и определяет, на работе которого из них может сказаться измененный код. Этот подход характеризуется экономией машинного времени за счет работы, проводимой человеком. Второй, более эффективный, заключается в автоматическом прогоне на компьютере всех тестовых наборов после каждого изменения текста. Данный подход гарантирует большую эффективность труда разработчика, поскольку он не должен тратить время на анализ всей совокупности тестовых наборов, для того чтобы определить, какие наборы следует прогонять, а какие нет.

Если вы сможете автоматизировать процесс блочного тестирования, то не только повысите качество тестирования, но и высвободите для себя значительно больше временных и материальных ресурсов, чем уйдет на этот процесс. Если вы пишете программы на языке C, то для автоматизации блочного тестирования можете воспользоваться существующими технологиями. Чем больше процессов вы сможете автоматизировать, тем больше пользы вы получите.

При выборе технологии блочного тестирования сначала следует ответить на следующие вопросы:

- Подходит ли эта технология для вашего текста и/или компилятора?
- Может ли она автоматически создавать тестовые схемы?
- Может ли она автоматически генерировать тестовые наборы?
- Позволяет ли она вводить создаваемые пользователем тестовые наборы и фиктивные модули?
- Автоматизировано ли регрессивное тестирование?
- Имеется ли в ее составе технология или связь с технологией автоматического распознавания ошибки в процессе прогона?

#### **Средства отладки, не меняющие режим работы программ**

Из-за того, что операционные системы реального времени должны выполнять определенные задачи в условиях заранее определенных временных ограничений, временные соотношения превращаются в важнейший параметр, который разработчики должны учитывать при установке тестового ПО. Обычно в процессе исполнения программ возникает множество различных прерываний, и чрезвычайно необходимо, чтобы в момент возникновения прерывания приложение реагировало корректно. Ситуация еще более усложняется, когда несколько прерываний возникает сразу или когда в системе исполняется несколько приложений с несколькими взаимодействующими друг с другом тредами. По сути, это приложения с несколькими одновременными путями исполнения различные кодовые последовательности как бы исполняются в одно и то же время, даже если в системе всего один центральный процессор. Интересно заметить, что если бы эти приложения исполнялись в нескольких процессорах, то различные треды на практике были бы загружены в разные процессоры.

Если возникающие в приложениях реального времени ошибки проявляются во взаимодействиях между программой и прерываниями, то они будут в значительной мере чувствительны ко времени. В этом случае критически важно регистрировать порядок возникновения ошибок, поскольку это позволит разобраться в причинах и следствиях каждой ошибки. В этом как раз и кроется главная проблема отладки систем реального времени: существует достаточное количество трудновывяемых ошибок, которые проявляются только при определенных временных соотношениях.

Эта проблема усложняется тем, что подобные ошибки не так-то просто воспроизводятся. Очень трудно воссоздать ситуацию с такими же временными соотношениями, что и приведшие к возникновению ошибки в реальной программе. Механизм отладки таких приложений должен быть максимально возможно щадящим. Любое вмешательство в ход исполнения программ может привести к изменению ее временных характеристик и отсутствию условий возникновения ошибок. Конечно, создание условий, при которых ошибки не возникают, это хорошо, но в данном случае это является препятствием отладке программы.

Теоретической основой проблемы отладки систем реального времени может послужить известный всем из курса физики принцип неопределенности немецкого физика Вернера Гейзенберга, согласно которому одновременно определить скорость и местоположение движущейся частицы невозможно. Гейзенберг считал, что, определяя координаты частицы, экспериментатор тем самым изменяет её местоположение, что не позволяет определить её координаты точно. Операция измерения влияет на измеряемый объект и искажает результаты измерения. Принцип неопределенности это одна из аксиом квантовой механики.

Применительно же к нашей теме, этот принцип означает, что отладка системы требует сбора информации о ее состоянии. Однако сбор информации о состоянии системы меняет ее временные характеристики и существенно затрудняет надежное воспроизведение условий возникновения ошибки.

Таким образом, суть этой проблемы в том, что нужно найти способ обнаружения ошибок реального времени и анализа поведения про-

граммы без влияния на существующие временные соотношения. Наверное, вашим первым порывом было бы обращение к отладчику, но отладчики, как правило, прерывают исполнение программы и, соответственно, изменяют ее временные характеристики. Малопригодны и системы моделирования, поскольку они не могут воссоздать временные характеристики реальных технических средств. Еще никто не создал такую систему моделирования, которая могла бы смоделировать режим реального времени; временные параметры можно определить, только загрузив программу в само железо.

Последнее требует наличия специального механизма для упрощенной регистрации состояния системы. Один из возможных и подходящих механизмов запись информации в оперативную память, поскольку такая операция выполняется чрезвычайно быстро. Один из способов применения этого механизма организация где-нибудь в памяти специального буфера и использование в вашей программе указателя на этот буфер. Указатель всегда ссылается на начало буфера. В программу вставляются операции записи в ячейку, определяемую указателем. После каждой операции записи значение указателя меняется соответствующим образом. Иногда полезно пользоваться кольцевым буфером (т.е. когда после записи в последнюю ячейку буфера указатель начинает показывать на начало буфера), что позволяет отслеживать ситуации, приводящие к возникновению проблемы. Необходимо при этом предусмотреть способ сохранения содержимого буфера после нормального или аварийного завершения программы, чтобы впоследствии иметь к нему доступ и проводить так называемую "посмертную отладку". Способ реализации зависит от аппаратных средств, обычно это можно сделать, если не выполнять повторную инициализацию (reset) оборудования.

Теперь вам нужен механизм чтения этой памяти. Здесь можно использовать и отладчик, и другие средства извлечения информации из оперативной памяти. В частности, можно написать простенькую программу, которая будет пересылать эти данные в файл или на принтер. Каким бы средством вы не пользовались, конечным этапом, вероятнее всего, будет ручной анализ содержимого буфера. Если ваш буфер кольцевой, то вам необходимо иметь точные сведения о значении указателя; события, которые стали началом последовательности, будут непосредственно перед указателем, события, которые возникли непосредственно перед крахом, будут сразу же после указателя.



Рис. 5. Последовательность событий в кольцевом буфере

Теперь ваша главная задача попытаться разобраться в последовательности данных, записанных в буфере. Эта задача аналогична исследованию причин катастрофы самолета по показаниям приборов, зарегистрированных "черным ящиком" самолета. Анализ характеристик программы в этом случае проводится после свершившегося события, что, естественно, гораздо меньше влияет на ее исполнение, чем контроль в течение работы.

Иногда бывает очень трудно восстановить приведшие к краху события, и четкого понятия о моменте возникновения ошибки нет. Только на выяснение причины ошибки могут уходить многие месяцы. В таких случаях для поиска ошибочного оператора можно воспользоваться логарифмическим методом отладки. В разных местах отлаживаемого кода расставляются маркеры (например, операторы типа exit), а перед ними операторы записи в память. Затем запускаете программу и ожидаете момента краха. В случае краха вы знаете, между какими маркерами он произошел. Этот метод позволяет выявлять и проблемы согласования по времени, поскольку он позволяет находить сегменты кода, в которых возникают нарушения временных соотношений.

Ещё одно решение это применение в качестве технологии отладки так называемых брандмауэров. Брандмауэр это точка в логическом потоке программы, в которой доказывается справедливость предположений, на которые опирается последующий код. Проверка этих предположений отличается от обычного контроля ошибок. Срабатывание брандмауэра представляет собой сигнал разработчику о том, что внутреннее состояние системы неустойчиво. Это может произойти, например, если ожидающая строго положительного аргумента функция получает нулевую или отрицательную величину. Неискушённым разработчикам большинство брандмауэров кажутся тривиальными и ненужными. Однако опыт разработки крупных проектов показывает, что по мере развития и совершенствования программных систем неявные предположения в отношении среды исполнения нарушаются все чаще и чаще. Во многих случаях даже сам автор затрудняется сформулировать, что представляют собой надлежащие условия исполнения того или иного участка кода.

Реализуемые внутри встраиваемых систем брандмауэры нуждаются в специальных средствах связи для передачи сообщений во внешний мир; обсуждение способа установления таких каналов передачи выходит за рамки настоящей статьи.

### **Заключение**

Рассмотренные выше методы предотвращения и обнаружения ошибок, а также технологии отладки могут значительно повысить качество программного обеспечения встраиваемых систем и уменьшить затрачиваемые на проведение отладки материальные и временные ресурсы. Вышеупомянутые методы без особого труда могут быть использованы в разработке самых разных проектов программного обеспечения встраиваемых систем, причем накопленный опыт полностью сохраняет свою ценность и при реализации иных проектов и целевых технологий. Кроме того, они позволяют гарантировать простоту сопровождения, модификации и портирования созданных программ в устройства новых типов. И говоря коротко, рассматриваемые методы дают возможность не только совершенствовать существующие встроенные приложения и процессы разработки, но и гарантировать, что с распространением новых встраиваемых устройств у вас уже будет накоплен опыт, необходимый для разработки высокоэффективных приложений для этих технологий причем вовремя и в соответствии с выделенным бюджетом.

**по теме:** Выполнение обслуживания информационной системе в соответствии с пользовательской документацией  
**Цель:** выполнить обслуживание информационной системы в соответствии с пользовательской документацией  
**По завершению практического занятия студент должен уметь:** обслуживать информационную систему в соответствии с пользовательской документацией

Продолжительность: 4 аудиторных часа (180 минут)

#### **Необходимые принадлежности**

Персональный компьютер, программное обеспечение: среда PowerDesigner.

#### **Задание**

Документация информационного обеспечения АСУ предназначена для описания проектных решений по информационному обеспечению в документах:

- описание информационного обеспечения АСУ;
- описание организации информационной базы;
- описание системы классификации и кодирования;
- чертеж формы документа (видеограммы);
- описание массива информации;
- перечень входных сигналов и данных;
- перечень выходных сигналов (документов);
- описание технологического процесса обработки данных.

1.2. При разработке документов на части АСУ содержание разделов каждого документа ограничивают рамками соответствующей части.

1.3. В зависимости от назначения и специфических особенностей создаваемых АСУ допускается включать в документы дополнительные разделы и сведения, требования к содержанию которых не установлены настоящим стандартом.

1.4. Отсутствие проектных решений по разделу документа фиксируют в соответствующем разделе с необходимыми пояснениями.

#### **ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТОВ**

##### **Описание информационного обеспечения АСУ**

2.1.1. Документ должен состоять из следующих разделов:

- принципы организации информационного обеспечения;
- организация сбора и передачи информации;
- построение системы классификации и кодирования;
- организация внутримашинной информационной базы;
- организация внешнемашинной информационной базы.

##### **2.1.2. Требования к содержанию разделов**

2.1.2.1. В разделе «Принципы организации информационного обеспечения» должны быть приведены:

- состав, структура и принципы организации информационного обеспечения;
- обоснование выбора носителей данных и принципы распределения информации по типам данных;
- описание принятых видов и методов контроля в маршрутах обработки данных при создании и функционировании внешнемашинной и внутримашинной информационных баз с указанием требований, на соответствие которым проводят контроль;
- описание решений, обеспечивающих информационную совместимость АСУ с другими связанными с ней системами управления по источникам, потребителям информации, по сопряжению применяемых классификаторов (при необходимости), по использованию в АСУ унифицированных систем документации.

2.1.2.2. В разделе «Организация сбора и передачи информации» должны быть приведены перечни источников, носителей информации, оценка интенсивности и объема информации, описание общих требований к организации сбора и передачи информации.

2.1.2.3. В разделе «Построение системы классификации и кодирования» должны быть приведены:

- описание принятых систем классификации объектов;
- методы кодирования объектов классификации;
- перечень применяемых общесоюзных, отраслевых и других зарегистрированных классификаторов.

2.1.2.4. Раздел «Организация внутримашинной информационной базы» должен содержать описание принципов построения базы, характеристики ее состава и объема, структуры базы на уровне баз данных с описанием характера взаимосвязей баз данных и указанием функций АСУ, при реализации которых используют каждую базу данных, характеристики данных, содержащихся в каждой базе данных.

2.1.2.5. Раздел «Организация внешнемашинной информационной базы» должен содержать характеристики состава и объема, принципы построения базы, в том числе основные положения по организации и обслуживанию фонда нормативно-справочной информации во взаимосвязи с автоматизированными функциями управления.

2.1.2.6. В приложениях следует приводить справочные и другие вспомогательные материалы и сведения (систематизированный перечень наименований структурных единиц информации с присвоенными им обозначениями и описаниями их сущности).

#### **Описание системы классификации и кодирования**

Документ должен содержать по каждому классифицируемому объекту описание метода кодирования, структуру и длину кода, указание о системе классификации и другие сведения по усмотрению разработчика.

#### **Чертеж формы документа (видеограммы)**

В документе должно быть приведено изображения формы документа или видеограммы в соответствии с требованиями государственных стандартов унифицированной системы документации и необходимые пояснения.

#### **Описание массива информации**

Документ должен содержать:

- наименование массива;
- обозначение массива;
- наименование носителя данных;
- перечень реквизитов в порядке их следования в записях массива с указанием по каждому реквизиту: обозначения алфавита, длины в знаках, диапазона изменения (при необходимости), логических и семантических связей с другими реквизитами данной записи и другими записями массива;
- оценку объема массива;
- другие характеристики массива (при необходимости).

**Примечание.** Если массив состоит из записей различных типов, то для записи каждого типа приводят все характеристики, перечисленные выше.

#### **Техническая документация**

При создании программы, одного лишь кода, как правило, недостаточно. Должен быть предоставлен некоторый текст, описывающий различные аспекты того, что именно делает код. Такая документация часто включается непосредственно в исходный код или предоставляется вместе с ним.

Подобная документация имеет сильно выраженный технический характер и в основном используется для определения и описания API, структур данных и алгоритмов.

Часто при составлении технической документации используются автоматизированные средства — генераторы документации, такие как Doxygen, javadoc, NDoc и другие. Они получают информацию из специальным образом оформленных комментариев в исходном коде, и создают справочные руководства в каком-либо формате, например, в виде текста или HTML.

Использование генераторов документации и документирующих комментариев многими программистами признаётся удобным средством, по различным причинам. В частности, при таком подходе документация является частью исходного кода, и одни и те же инструменты могут использоваться для сборки программы и одновременной сборки документации к ней. Это также упрощает поддержку документации в актуальном состоянии.

#### **Маркетинговая документация**

Для многих приложений необходимо располагать рядом с ними рекламные материалы, с тем чтобы заинтересовать людей, обратив их внимание на продукт. Такая форма документации имеет целью:

- подогреть интерес к продукту у потенциальных пользователей
- информировать их о том, что именно делает продукт, с тем чтобы их ожидания совпадали с тем, что они получают
- объяснить положение продукта по сравнению с конкурирующими решениями

Одна из хороших маркетинговых практик — предоставление слогана — простой запоминающейся фразы, иллюстрирующей то, что мы хотим донести до пользователя, а также характеризующей *ощущение*, которое создаёт продукт.

Часто бывает так, что коробка продукта и другие маркетинговые материалы дают более ясную картину о возможностях и способах использования программы, чем всё остальное.

#### **ОБЩИЕ ПОЛОЖЕНИЯ**

1.1. Документация по техническому обеспечению АСУ предназначена для описания проектных решений по техническому обеспечению в документах:

- описание комплекса технических средств;
- структурная схема комплекса технических средств;
- план расположения;
- перечень заявок на разработку новых технических средств;
- перечень заданий заказчику АСУ (Генпроектировщику) на проектирование в смежных частях проекта объекта, связанное с созданием АСУ;
- ведомость оборудования и материалов;
- технические требования к технологическому объекту управления;
- задание на проектирование в смежной части проекта объекта, связанное с созданием АСУ;
- проектная оценка надежности комплекса технических средств;
- принципиальная схема;
- схема автоматизации;
- таблица соединений и подключений;
- схема соединений внешних проводок;
- чертеж общего вида;
- схема подключений внешних проводок;
- спецификация оборудования;
- чертеж установки технических средств;
- ведомость потребности в материалах.

**(Измененная редакция, Изм. № 1)**

1.2. При разработке документов на подсистему содержание разделов каждого документа ограничивают рамками данной подсистемы.

1.3. В зависимости от назначения и специфических особенностей создаваемых АСУ допускается включать в документы дополнительные разделы и сведения, требования к содержанию которых не установлены настоящим стандартом.

1.4. Отсутствие проектных решений по разделу документа фиксируют в соответствующем разделе с необходимыми пояснениями.

## **ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТОВ СОСТАВ И СОДЕРЖАНИЕ**

2.1. ТЗ содержит следующие разделы, которые могут быть разделены на подразделы:

- 1) общие сведения;
- 2) назначение и цели создания (развития) системы;
- 3) характеристика объектов;
- 4) требования к системе;
- 5) состав и содержание работ по созданию системы;
- 6) порядок контроля и приемки системы;
- 7) требования к составу и содержанию работ по подготовке объекта разработки к вводу системы в действие;
- 8) требования к документированию;
- 9) источники разработки.

В ТЗ могут включаться приложения.

2.2. В зависимости от вида, назначения, специфических особенностей проекта и условий функционирования системы допускается оформлять разделы ТЗ в виде приложений, вводить дополнительные, исключать или объединять подразделы ТЗ.

В ТЗ на части системы не включают разделы, дублирующие содержание разделов ТЗ в целом.

2.3. В разделе «Общие сведения» указывают:

- 1) полное наименование системы и ее условное обозначение;
- 2) шифр темы или шифр (номер) договора;
- 3) наименование компаний разработчика и заказчика (пользователя) системы и их реквизиты;
- 4) перечень документов, на основании которых создается система, кем и когда утверждены эти документы;
- 5) плановые сроки начала и окончания работы по созданию системы;
- 6) сведения об источниках и порядке финансирования работ;
- 7) порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы.

2.4. Раздел «Назначение и цели создания (развития) системы» состоит из подразделов:

- 1) назначение системы;
- 2) цели создания системы.

2.4.1. В подразделе «Назначение системы» указывают вид деятельности системы (управление, проектирование и т. п.) и перечень объектов информатизации (объектов), на которых предполагается ее использовать.

2.4.2. В подразделе «Цели создания системы» приводят наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта информатизации, которые должны быть достигнуты в результате создания ИС, и указывают критерии оценки достижения целей создания системы.

2.5. В разделе «Характеристики объекта информатизации» приводят:

- 1) краткие сведения об объекте информатизации или ссылки на документы, содержащие такую информацию;
- 2) сведения об условиях эксплуатации объекта автоматизации.

2.6. Раздел «Требования к системе» состоит из следующих подразделов:

- 1) требования к системе в целом;
- 2) требования к функциям (задачам), выполняемым системой;
- 3) требования к видам обеспечения.

Состав требований к системе, включаемых в данный раздел ТЗ на ИС, устанавливают в зависимости от вида, назначения, специфических особенностей и условий функционирования конкретной системы.

2.6.1. В подразделе «Требования к системе в целом» указывают:

- требования к структуре и функционированию системы;
- требования к численности и квалификации персонала системы и режиму его работы;
- показатели назначения;
- требования к надежности;
- требования к безопасности;
- требования к эргономике и технической эстетике;
- требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы;
- требования к защите информации от несанкционированного доступа;
- требования по сохранности информации при авариях;
- требования к защите от влияния внешних воздействий;
- требования к патентной чистоте;

- требования по стандартизации и унификации;
- дополнительные требования.

2.6.1.1. В требованиях к структуре и функционированию системы приводят:

- 1) перечень подсистем, их назначение и основные характеристики, требования к числу уровней иерархии и степени централизации системы;
- 2) требования к способам и средствам связи для информационного обмена между компонентами системы;
- 3) требования к характеристикам взаимосвязей создаваемой системы со смежными системами, требования к ее совместимости, в том числе указания о способах обмена информацией (автоматически, пересылкой документов, по телефону и т. п.);
- 4) требования к режимам функционирования системы;
- 5) требования по диагностированию системы;
- 6) перспективы развития, модернизации системы.

2.6.1.2. В требованиях к численности и квалификации персонала на ИС приводят:

- требования к численности персонала (пользователей) ИС;
- требования к квалификации персонала, порядку его подготовки и контроля знаний и навыков;
- требуемый режим работы персонала ИС.

2.6.1.3. В требованиях к показателям назначения ИС приводят значения параметров, характеризующие степень соответствия системы ее назначению.

2.6.1.4. В требованиях к надежности включают:

- 1) состав и количественные значения показателей надежности для системы в целом или ее подсистем;
- 2) перечень аварийных ситуаций, по которым должны быть регламентированы требования к надежности, и значения соответствующих показателей;
- 3) требования к надежности технических средств и программного обеспечения;
- 4) требования к методам оценки и контроля показателей надежности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами.

2.6.1.5. В требования по безопасности включают требования по обеспечению безопасности при поставке, наладке, эксплуатации и обслуживании системы.

2.6.1.6. В требования по эргономике и технической эстетике включают показатели ИС, задающие необходимое качество взаимодействия человека с машиной и комфортность условий работы персонала.

2.6.1.7. В требования к защите информации от несанкционированного доступа включают требования, установленные действующей в отрасли и информационной среде заказчика.

2.6.1.8. В требованиях по сохранности информации приводят перечень событий: аварий, отказов технических средств (в том числе - потеря питания) и т. п., при которых должна быть обеспечена сохранность информации в системе.

2.6.1.9. В требованиях по патентной чистоте указывают перечень стран, в отношении которых должна быть обеспечена патентная чистота системы и ее частей.

2.6.1.10. В дополнительные требования включают специальные требования по усмотрению разработчика или заказчика системы.

2.6.2. В подразделе «Требование к функциям (задачам)», выполняемым системой, приводят:

- по каждой подсистеме перечень функций, задач или их комплексов (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации;
- при создании системы в две или более очереди - перечень функциональных подсистем, отдельных функций или задач, вводимых в действие в 1-й и последующих очередях;
- временной регламент реализации каждой функции, задачи (или комплекса задач);
- требования к качеству реализации каждой функции (задачи или комплекса задач), к форме представления выходной информации, характеристики необходимой точности и времени выполнения, требования одновременности выполнения группы функций, достоверности выдачи результатов;
- перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

2.6.3. В подразделе «Требования к видам обеспечения» в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

2.6.3.2. Для информационного обеспечения системы приводят требования:

- 1) к составу, структуре и способам организации данных в системе;
- 2) к информационному обмену между компонентами системы;
- 3) к информационной совместимости со смежными системами;
- 4) по применению систем управления базами данных;
- 5) к структуре процесса сбора, обработки, передачи данных в системе и представлению данных;
- 6) к защите данных;
- 7) к контролю, хранению, обновлению и восстановлению данных;

2.6.3.3. Для лингвистического обеспечения системы приводят требования к применению в системе языков программирования высокого уровня, языков взаимодействия пользователей и технических средств системы, а также требования к кодированию и декодированию данных, к языкам ввода-вывода данных, языкам манипулирования данными, средствам описания предметной области, к способам организации диалога.

2.6.3.4. Для программного обеспечения системы приводят перечень покупных программных средств, а также требования:

- 1) к зависимости программных средств от операционной среды;
- 2) к качеству программных средств, а также к способам его обеспечения и контроля;

2.6.3.5. Для технического обеспечения системы приводят требования:

- 1) к видам технических средств, в том числе к видам комплексов технических средств, программно-технических комплексов и других комплектующих изделий, допустимых к использованию в системе;
- 2) к функциональным, конструктивным и эксплуатационным характеристикам средств технического обеспечения системы.

2.6.3.6. В требованиях к метрологическому обеспечению приводят:

- 1) предварительный перечень измерительных каналов;
- 2) требования к точности измерений параметров и (или) к метрологическим характеристикам измерительных каналов;
- 3) требования к метрологической совместимости технических средств системы;
- 4) перечень управляющих и вычислительных каналов системы, для которых необходимо оценивать точностные характеристики;
- 5) требования к метрологическому обеспечению технических и программных средств, входящих в состав измерительных каналов системы, средств, встроенного контроля, метрологической пригодности измерительных каналов и средств измерений, используемых при наладке и испытаниях системы;
- 6) вид метрологической аттестации (государственная или ведомственная) с указанием порядка ее выполнения и организаций, проводящих аттестацию.

2.6.3.7. Для организационного обеспечения приводят требования:

- 1) к структуре и функциям подразделений, участвующих в функционировании системы или обеспечивающих эксплуатацию;
- 2) к организации функционирования системы и порядку взаимодействия персонала ИС и персонала объекта информатизации;
- 3) к защите от ошибочных действий персонала системы.

2.7. Раздел «Состав и содержание работ по созданию (развитию) системы» должен содержать перечень стадий и этапов работ по созданию системы, сроки их выполнения, перечень организаций - исполнителей работ, ссылки на документы, подтверждающие согласие этих организаций на участие в создании системы, или запись, определяющую ответственного (заказчик или разработчик) за проведение этих работ.

В данном разделе также приводят:

- 1) перечень документов предъявляемых по окончании соответствующих стадий и этапов работ;
- 2) вид и порядок проведения экспертизы технической документации (стадия, этап, объем проверяемой документации, организация-эксперт);
- 3) программу работ, направленных на обеспечение требуемого уровня надежности разрабатываемой системы (при необходимости);
- 4) перечень работ по метрологическому обеспечению на всех стадиях создания системы с указанием их сроков выполнения и организаций-исполнителей (при необходимости).

2.8. В разделе «Порядок контроля и приемки системы» указывают:

- 1) виды, состав, объем и методы испытаний системы и ее составных частей;
- 2) общие требования к приемке работ по стадиям, порядок согласования и утверждения приемочной документации;

2.9. В разделе «Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие» необходимо привести перечень основных мероприятий и их исполнителей, которые следует выполнить при подготовке проекта к вводу ИС в действие.

В перечень основных мероприятий включают:

- 1) приведение поступающей в систему информации (в соответствии с требованиями к информационному и лингвистическому обеспечению);
- 2) создание условий функционирования проекта, при которых гарантируется соответствие создаваемой системы требованиям, содержащимся в ТЗ;
- 3) создание необходимых для функционирования системы подразделений и служб;
- 4) сроки и порядок комплектования штатов и обучения персонала.

2.10. В разделе «Требования к документированию» приводят:

- 1) согласованный разработчиком и Заказчиком системы перечень подлежащих разработке комплектов и видов документов; перечень документов, выпускаемых на машинных носителях;
- 2) при отсутствии государственных стандартов, определяющих требования к документированию элементов системы, дополнительно включают требования к составу и содержанию таких документов.

2.11. В разделе «Источники разработки» должны быть перечислены документы и информационные материалы, на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

## Сеть доступа

*Сеть доступа* представляет собой нижний уровень иерархии *телекоммуникационной сети*.

К этой сети подключаются конечные (терминальные) узлы - оборудование, установленное у пользователей (*абонентов*, клиентов) сети. В случае *компьютерной сети* конечными узлами являются компьютеры, телефонной - телефонные аппараты, а *телевизионной* или *радиосети* - соответствующие теле- и радиоприемники.

Основное назначение *сети доступа* - концентрация *информационных потоков*, поступающих по многочисленным каналам связи от оборудования пользователей, в сравнительно небольшом количестве узлов *магистральной сети*. *Сеть доступа*, как и *телекоммуникационная сеть* в целом, может состоять из нескольких уровней (на рисунке показано два). *Коммутаторы*, установленные в узлах нижнего уровня, мультиплексируют информацию, поступающую по многочисленным абонентским каналам (называемым часто абонентскими окончаниями, *local loop*) и передают ее коммутаторам верхнего уровня, чтобы те в свою очередь передали ее коммутаторам магистралей. Количество уровней *сети доступа* зависит от ее размера; небольшая *сеть доступа* может состоять из одного уровня, а крупная - из двух-трех. Следующие уровни осуществляют дальнейшую концентрацию трафика, собирая его и мультиплексируя в более скоростные каналы.

## Магистральная сеть

*Магистральная сеть* объединяет отдельные *сети доступа*, выполняя функции транзита трафика между ними по высокоскоростным каналам. *Коммутаторы магистралей* могут оперировать не только информационными соединениями между отдельными пользователями, но и *агрегированными информационными потоками*, переносящими данные большого количества пользовательских соединений. В результате информация с помощью *магистралей* попадает в *сеть доступа* получателей, демультиплексируется там и коммутируется таким образом, что на входной порт оборудования пользователя поступает только та информация, которая ему адресована.

В том случае, когда *абонент-получатель* подключен к тому же *коммутатору* доступа, что и *абонент-отправитель* (непосредственно или через подчиненные по иерархии связей *коммутаторы*), последний выполняет необходимую операцию *коммутации* самостоятельно.

## Информационные центры

*Информационные центры* /центры управления сервисами - это собственные информационные ресурсы сети, на основе которых осуществляется обслуживание пользователей. В таких *центрах* может храниться информация двух типов:

- пользовательская информация, то есть те данные, которые непосредственно интересуют пользователей сети;
- вспомогательная служебная информация, позволяющая предоставлять пользователям некоторые услуги.

Примером информационных ресурсов первого типа могут служить Web-порталы, на которых расположена разнообразная справочная информация и новости, информация электронных магазинов и т.п. В *телефонных сетях* роль таких центров играют службы экстренного вызова (например, милиции, скорой помощи) и справочные службы различных организаций и предприятий - вокзалов, аэропортов, магазинов и т.п. В *телевизионных сетях* такими центрами являются телестудии, поставляющие "живую" картинку или же воспроизводящие ранее записанные сюжеты или фильмы.

К ресурсам второго типа относятся, например, различные *системы аутентификации* и *авторизации* пользователей, с помощью которых организация, владеющая сетью, проверяет права пользователей на получение тех или иных услуг; системы биллинга, которые в коммерческих сетях подсчитывают плату за предоставленные услуги; базы данных учетной информации пользователей, хранящие имена и пароли, а также перечни услуг, на которые подписан каждый пользователь. В *телефонных сетях* существуют центры управления сервисами (Services Control Point, *SCP*), где установлены компьютеры, на которых хранятся программы нестандартной обработки телефонных вызовов пользователей, например вызовов бесплатных справочных служб коммерческих предприятий (так называемые службы 800) или вызовов при проведении телеголосования. Еще одним из распространенных видов вспомогательного *информационного центра* является централизованная система управления сетью, которая представляет собой программное обеспечение, работающее на одном или нескольких компьютерах.

## Текущий контроль бсеместр

### Практическое занятие форма текущего контроля

**по теме:** Разработка технического задания на сопровождение информационной системы (указать предметную область)

**Цель:** разработка технического задания на сопровождение информационной системы (указать предметную область)

**По завершению практического занятия студент должен уметь:** разрабатывать техническое задание на сопровождение информационной системы (указать предметную область)

Продолжительность: 4 аудиторных часа (180 минут)

## Необходимые принадлежности

Персональный компьютер, программное обеспечение.

## Задание

\_\_\_ В соответствии с проведенным анализом предметной области были сформулированы технические требования к проектируемой информационно-управляющей системе. Технические требования сформулированы в формате, определенном ГОСТами 34 серии [6,7].

### 1 Общие сведения

#### 1.1 Полное наименование системы и ее условное обозначение:

Полное наименование системы: Информационно-управляющая система для проведения олимпиад по информатике "Инфотест"

Краткое наименование системы: ИС "Инфотест", Система

#### 1.2 Порядок оформления и предъявления заказчику результатов работ

Результаты работ по созданию ИС "Инфотест" оформляются и предъявляются поэтапно в соответствии с Календарным планом.

## 2 Назначение и цели создания системы

### 2.1 Назначение системы

Информационно-управляющая система предназначена для поддержки проведения олимпиады по информатике в области выполнения процессов:

- Сбор личных данных участников олимпиады;
- Хранение решений участников, заданий и другой информации;
- Проверка полученных участниками результатов выполненных олимпиадных заданий с минимальными трудозатратами;
- Взаимодействие между участниками олимпиады и членами жюри или администраторами посредством форума;
- Ведение журнала пользователей;
- Создание бэкапов;
- Проведение аналитических исследований на основе статистических данных;
- Обеспечения разделения доступа к данным и предоставления средств защиты информации, уменьшающих риск фальсификаций.

### 2.2 Цели создания системы

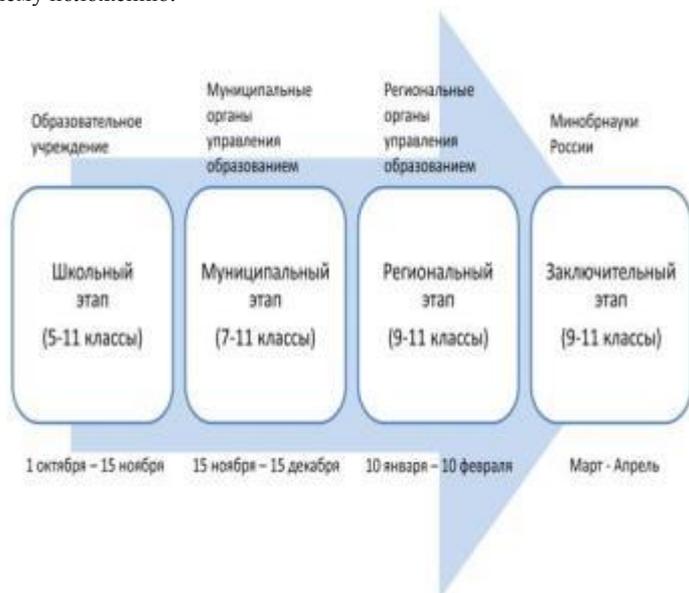
ИС "Инфотест" создается с целью:

- Повышения эффективности учебного процесса;
- Упрощения широкого спектра различных видов деятельности: проверки решений учащихся, организации процесса тестирования.
- Уменьшения рисков возникновения недостоверных данных вследствие человеческого фактора;
- Оперативного получения учащимися информации о выданных им задачах, а так же о статусе проверки их решений;
- Автоматического контроля за соблюдением учащимися сроков выполнения работ;
- Возможности создания единой базы задач для их неоднократного применения в учебном процессе.

## 3 Характеристика объекта автоматизации

Объектом автоматизации выступает Олимпиада по информатике.

На рисунке 2 представлена диаграмма, отображающая этапы проведения Всероссийской олимпиады школьников согласно последнему положению.



**Рисунок 2. Диаграмма этапов Всероссийской олимпиады школьников**

На рисунке 2 отмечены организаторы каждого из этапов олимпиад, возрастные рамки и сроки проведения.

В первую очередь стоит отметить, что проведение школьного, муниципального и регионального этапов олимпиады регламентируется Положением о Всероссийской олимпиаде школьников, утвержденным Приказом Министерства образования и науки Российской Федерации (Минобрнауки России) от 18 ноября 2013 г. № 1252 г. Москва [1].

Первым туром Всероссийской олимпиады является школьный, принять участие в нем может любой учащийся 5-11 класса образовательных организаций, поэтому данный тур является очень массовым.

Основной функцией данного тура является отбор на последующие этапы олимпиады. Для проведения школьного этапа олимпиад создаются оргкомитет, жюри и предметно-методическая комиссия олимпиады. Организатором школьного этапа выступает образовательная организация, как правило, им является школа. За разработку заданий и критериев проверки отвечает школьная предметно-методическая комиссия или школьное жюри на основании методических рекомендаций центральных предметно-методических комиссий.

Вторым туром является муниципальный этап, в нем могут принять участие учащиеся в 7-11 классах образовательных учреждений, победители и призеры школьного тура текущего года либо прошлого учебного года, если продолжили своё обучение. Организатором тура служат органы местного самоуправления муниципальных районов и городских округов в сфере образования.

Муниципальный этап Всероссийской олимпиады играет важную роль в успешности проведения последующих этапов, так как именно по его итогам формируются списки регионального этапа.

Региональный этап проходит в два тура и является заключительным во всей цепочке. Оба тура компьютерные. За его проведение отвечают органы исполнительной власти субъектов Российской Федерации, осуществляющих управление в сфере образования. Разработкой заданий, критериев и методики проверки занимаются центрально-методические комиссии.

На усмотрение организаторов и жюри регионального этапа Олимпиады накануне первого тура для всех участников может быть организован пробный тур, основное назначение которого - знакомство участников с компьютерной техникой и установленным на рабочих местах программным обеспечением. Пробный тур из рекомендательного должен стать обязательным, если во время проведения основных туров участники должны использовать в процессе решения задач специализированную программную среду соревнований, позволяющую осуществлять проверку решений участников в автоматическом режиме.

Для обеспечения одинаковых условий участникам должны быть предоставлены одинаковые или близкие по техническим характеристикам компьютеры. Компьютеры должны быть объединены в локальную вычислительную сеть без доступа к сети Интернет. На компьютерах всех участников должно быть установлено только то программное обеспечение, которое необходимо для решения задач олимпиады. А также на каждом из устройств организаторами олимпиады устанавливается программный продукт, обеспечивающий работу среды. Необходимо сказать, что такая среда не поставляется вместе с материалами центральной предметно-методической комиссии по информатике, и обеспечение регионального этапа такой системой находится в ведении организаторов регионального этапа, региональной предметно-методической комиссии по информатике и жюри [1].

Проанализировав положение о Всероссийской олимпиаде, можно обнаружить некоторые риски, которые могут возникнуть во время организации каждого из этапов олимпиады.

В связи с тем, что отсутствует реальный контроль со стороны оргкомитета школьного этапа, составители могут разработать такие задания, которые не будут удовлетворять методическим рекомендациям. Такое несоответствие заданий олимпиады методическим рекомендациям дает возможность исказить реальные результаты тура. Например, при слишком легком уровне заданий, призеров будет больше, а как следствие и количество участников, прошедших на муниципальный этап олимпиады. Так и наоборот, слишком сложные задания не дадут возможности некоторым участникам пройти в следующий тур.

Существенной проблемой может стать некачественная проверка работ. Это может быть связано с тем, что критерии проверки работ слабо формализованы или жюри недобросовестно и непрофессионально подходит к организации данного процесса. Вследствие чего в какой-то мере опять теряется одна из основных функций поэтапного прохождения олимпиады - функция отбора участников на её следующий тур.

Кроме всего прочего, отсутствует единая информационная система. Вся передаваемая информация из центров проведения олимпиады не систематизирована, отсутствует единый центр обработки информации. При проведении олимпиады каждая учебная организация владеет частью информации, что значительно затрудняет ее сбор и анализ. Данные поступают не одновременно, могут дублироваться. Все это является существенной проблемой при сжатых сроках подведения итогов олимпиад.

#### **4 Требования к системе**

##### **4.1 Требования к системе в целом**

###### **4.1.1 Требования к структуре и функционированию системы**

ИС "Инфотест" может быть централизованной, т.е. все данные должны располагаться в центральном хранилище, а может быть децентрализованной и работать на нескольких площадках с последующим сбором всей информации с них в единый центр.

Перечень подсистем, их назначение и основные характеристики

Система должна иметь трехуровневую архитектуру (клиентская часть - сервер приложений - сервер базы данных).

Система должна включать в себя следующие прикладные функциональные компоненты (функциональные подсистемы и/или АРМы):

- Подсистема управления справочниками и классификаторами;
- АРМ "Администратор";
- АРМ "Учитель";
- АРМ "Участник олимпиады";
- АРМ "Организатор олимпиады";
- АРМ "Методическая комиссия";
- АРМ "Жюри";
- АРМ "Дирекция по проведению Олимпиады";
- Подсистема поддержки сред программирования;
- Подсистема сбора и анализа отзывов участников олимпиады;
- Работа основных функциональных подсистем должны обеспечиваться следующими технологическими подсистемами:
- Операционные системы и системы управления базами данных;
- Локальная вычислительная сеть, компьютерное и специализированное оборудование.

ИС "Инфотест" должна обеспечивать базовые системные сервисы (передача информации, управление пользователями, обеспечение защиты информации и т.п.) для функциональных подсистем.

**4.1.1.1 Настоящее Техническое задание определяет также требования к системному программному обеспечению, оборудованию и используемым каналам связи.**

###### **4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы.**

Использовать протокол TCP/IP в качестве протокола взаимодействия между компонентами системы на транспортно-сетевом уровне. Использовать HTTP для организации доступа пользователей к системе. Использовать SOAP протокол поверх протокола HTTP.

###### **4.1.1.3 Требования к режимам функционирования системы.**

Для ИС "Инфотест" определяются режимы функционирования: основной режим, в котором все подсистемы исправно выполняют все свои функции; аварийный режим, в котором одна или все подсистемы не могут выполнить все или какую-либо одну из своих функций.

В основном режиме функционирования системы: клиентское и серверное программное обеспечение гарантирует возможность круглосуточного функционирования с перерывами на техническое обслуживание; системное, базовое и прикладное программное обеспечение функционирует в полной мере без неполадок.

Выполнение всех требований эксплуатации программного обеспечения и комплекса технических средств обеспечивает функционирование системы в основном режиме.

В аварийном режиме функционирования системы происходит отказ одного и/или нескольких компонент программного/технического обеспечения. Прежде, чем перейти в аварийный режим, система переходит в предаварийный режим, в котором необходимо совершить следующие действия: завершить работу всех открытых приложений с обязательным сохранением данных; выполнить резервное копирование базы данных; отключить все периферийные устройства, если таковые используются. После этого необходимо принять комплекс мер по устранению причины перехода в аварийный режим.

###### **4.1.1.4 Требования по диагностированию системы.**

Специальные требования не предъявляются.

##### **4.1.2 Требования к численности и квалификации персонала системы и режиму его работы**

###### **4.1.2.1 Требования к численности персонала**

Минимальное количество персонала, требуемого для нормального функционирования системы должно составлять не менее двух штатных единиц: ответственный за информационное обслуживание администратор БД и системный администратор.

###### **4.1.2.2 Требования к квалификации персонала**

Ответственный за информационное обслуживание Системы должен обладать практическими навыками работы с пользовательским интерфейсом операционной системы, знать общие принципы организации и функционирования информационных систем, быть компетентным в предметной области Системы. В перечень задач, выполняемых ответственным за информационное обслуживание Системы, должны входить ввод и редактирование информации БД.

Основными обязанностями системного администратора являются:

- 1 Установка, обновление и конфигурирование программного обеспечения технических средств;
- 2 Поддержание программного обеспечения серверов и клиентских станций в работоспособном состоянии;
- 3 Обеспечение своевременного копирования, архивирования и резервирования данных;
- 4 Восстановление нормального функционирования локальной сети при сбоях работы или выходе из строя сетевого оборудования.
- 5 Сопровождение программных средств;
- 6 Обеспечение безопасности сетевых ресурсов, включая контроль доступа в сеть Интернет.

Системный администратор должен иметь высшее профильное образование, опыт обслуживания технических средств и в области настройки и администрирования, применяемых в системе СУБД.

Пользователи системы должны иметь опыт работы в браузерах: Internet Explorer, и/или Opera, и/или Yandex, и/или Mozilla Firefox, и/или Google Chrome.

#### **4.1.2.3 Требования к режимам работы персонала**

Предполагается, что система будет установлена на персональных компьютерах/ноутбуках. Требования к режимам работы персонала устанавливаются с учетом соответствующего типа техники, на котором устанавливается система.

#### **4.1.3 Показатели назначения**

Система должна обеспечивать возможность исторического хранения данных по пользователям до 5 лет. Система должна обеспечивать возможность многопользовательской работы. Время формирования списка задач, пользователей и выполнения любых несложных запросов должно не превышать 15 секунд. Время сбора аналитической информации и формирования таблиц определяется их сложностью и может занимать продолжительное время.

#### **4.1.4 Требования к надежности**

Под надежностью системы понимается её возможность сохранять или восстанавливать работоспособность своего функционала после возникновения внештатных ситуаций: сбоев в системе электроснабжения, появления ошибок в работе аппаратных средств (за исключением носителей и дисков). Защита аппаратуры от больших скачков напряжения должна обеспечиваться сетевыми фильтрами.

#### **4.1.5 Требования к эргономике и технической эстетике**

Под эргономикой и технической эстетикой системы понимается удобство интерфейса и его ориентированность на пользователя. Система должна обеспечивать интерфейс, отвечающий следующим требованиям:

- Взаимодействие между пользователем и системой должно происходить на русском языке;
- Ориентированность на работу с клавиатурой и манипулятором графической информации "мышь";
- Отображение на экране только тех возможностей, которые доступны конкретному пользователю в соответствии с его ролью в системе;
- Отображаемые на информационные элементы должны быть типизированы;
- Во время диалога с пользователем, система должна отображать подсказки и выводить ошибки, в случае их появления, на экран.

#### **4.1.6 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы**

Условия эксплуатации, а также виды и периодичность обслуживания технических средств Системы должны соответствовать требованиям по эксплуатации, техническому обслуживанию, ремонту и хранению, изложенным в документации завода-изготовителя (производителя) на них.

#### **4.1.7 Требования к защите информации от несанкционированного доступа**

##### **4.1.7.1 Требования к информационной безопасности**

Информационная безопасность Системы должна обеспечиваться комплексом программно-технических средств и организационных мер. В тоже время программно-технические средства не должны существенно влиять на функциональность системы. Защита системы должна поддерживаться на всех уровнях обработки информации и во всех режимах работы.

##### **4.1.7.2 Требования к антивирусной защите**

Средства антивирусной защиты должны обеспечивать защиту от вредоносных программ серверов и АРМ пользователей. Средства должны быть установлены на всех рабочих местах пользователей и администраторов Системы.

#### **4.1.8 Требования по сохранности информации при авариях**

В Системе должно быть обеспечено резервное копирование данных.

#### **4.1.9 Требования к защите от влияния внешних воздействий**

Применительно к программно-аппаратному окружению Системы предъявляются следующие требования к защите от влияния внешних воздействий: электромагнитное излучение радиодиапазона, возникающее при работе электробытовых приборов, электрических машин и установок, приёмопередающих устройств, эксплуатируемых на месте размещения Системы, не должны приводить к нарушениям работоспособности подсистем. Система должна иметь возможность функционирования при колебаниях напряжения электропитания в пределах от 155 до 265 В ( $220 \pm 20\% - 30\%$ );

#### **4.1.10 Требования по стандартизации и унификации**

Для работы с базой данных должен использоваться язык запросов SQL в рамках стандарта ANSI SQL-92.

#### **4.1.11 Дополнительные требования**

Специальные требования не предъявляются.

#### **4.1.12 Требования безопасности**

Специальные требования не предъявляются.

#### **4.2 Требования к функциям (задачам), выполняемым системой**

Данная система предназначена для автоматизации процессов организации и проведения олимпиады по информатике.

Функции системы включают в себя функции, обеспечивающие работу с нормативно-справочной информацией, а также функции, обеспечивающие ведение, обработку и анализ текущей информации, относящейся к процессам организации и проведения олимпиады по информатике.

1 Ведение нормативно-справочной информации:

- Добавление НСИ
- Редактирование НСИ
- Удаление НСИ

2 Использование НСИ.

Эта функция предполагает использование нормативно-справочной информации для обеспечения работы других подсистем/АРМов (без права добавления, редактирования, удаления НСИ, но с возможностью просмотра).

Работа с основными функциями системы предполагает ввод, редактирование и удаление текущей информации, ее обработку, формирование первичных документов и аналитических отчетов.

#### **4.2.1 Список функций по работе с НСИ**

- 1 Ведение справочника "Регионы" (АРМ "Администратор");
- 2 Ведение справочника "Региональные площадки" (АРМ "Администратор");
- 3 Ведение справочника "Классы" (АРМ "Администратор");
- 4 Ведение справочника "Страны" (АРМ "Администратор");
- 5 Ведение справочника "Среды программирования" (АРМ "Администратор");
- 6 Ведение справочника "Типы заданий" (АРМ "Администратор");
- 7 Ведение справочника "Роли" (АРМ "Администратор");
- 8 Ведение справочника "Турсы Олимпиады" (АРМ "Администратор");

Пользователи других подсистем/АРМов только используют перечисленные справочники и классификаторы.

#### **4.2.2 Список функций, реализуемых системой**

По завершению разработки система должна предоставлять хранение информации о ходе соревнования, базы участников и их рейтинга, вопросов, заданий, а так же:

Администратору системы:

- Возможность создания ограниченных по времени олимпиад и работы над ними;
- Добавление и редактирование (при необходимости) условий задач и ответов к ним (предполагается создание вопросов с любым контентом);
- Активация, деактивация или удаление олимпиады;
- Возможность одновременного проведения нескольких олимпиад;
- Возможность автоматической проверки введенных ответов на задачу и подсчета результатов;
- Возможность автоматического вычисления статистики по соревнованию (по его завершению соответственно);
- Возможность просмотра списка зарегистрированных пользователей в системе и информации о них (с возможностью записи участника на необходимый турнир, либо его удаления);
- Возможность изменения прав доступа любого пользователя в системе;
- Возможности, предоставляемые любому пользователю системы тестирования, описанные ниже;

Участнику олимпиады:

- Регистрация/авторизация участника (при регистрации участника в АРМе "Участник" на каждого участника заводится "электронная карточка" участника);
- Ознакомление со справкой, техническому руководству и правилам проведения олимпиады (после регистрации в системе, участников следует ознакомить с правилами проведения олимпиады, а также донести до их сведения всю необходимую информацию для успешного прохождения испытания);
- Просмотр заданий (каждый пользователь имеет возможность просмотреть доступные в системе задания и вопросы);
- Просмотр активных и возможных олимпиад;
- Просмотр личных результатов олимпиады для каждого участника (список проверенных решений участника);
- Просмотр конечных результатов (список участников, призеров и победителей);
- Просмотр архива прошедших турниров и их результатов (список участников, призеров и победителей);
- Возможность связи с администратором/учителем/жюри (отправка сообщения)
- Доступ к форуму, на котором в соответствующих разделах можно задавать вопросы и обсуждать конкретные задачи и олимпиады;

Для организаторов олимпиады:

- Публикация Положения об олимпиаде и внесение в него необходимых изменений;
- Формирование списка методической комиссии и жюри олимпиады;
- Публикация правил проведения олимпиады;
- Публикация регламента проведения олимпиады;
- Публикация результатов олимпиады, списка победителей и призеров;
- Рассмотрение апелляций участников;
- Формирование отчета о прошедшей олимпиаде.

Для методической комиссии:

- Предоставление требований для утверждения оргкомитетом спецификации заданий;
- Предоставление требований для проведения олимпиады;
- Рассмотрение апелляций участников;

Для жюри олимпиады:

- Возможность отмены результатов участников, нарушивших регламент проведения олимпиады;
- Проверка результатов олимпиады;
- Предоставление для утверждения результатов олимпиады Организаторам;
- Рассмотрение апелляций участников.

### **4.3 Требования к видам обеспечения**

#### **4.3.1 Требования к математическому обеспечению**

Специальных требований не предъявляется.

#### **4.3.2 Требования к информационному обеспечению**

##### **4.3.2.1 Требования к составу, структуре и способам организации данных в системе.**

Модель данных Системы физически должна быть реализована в СУБД.

##### **4.3.2.2 Требования к информационному обмену между компонентами системы**

Специальных требований не предъявляется.

##### **4.3.2.3 Требования к информационной совместимости со смежными системами**

Специальных требований не предъявляется.

##### **4.3.2.4 Требования по использованию классификаторов, унифицированных документов и классификаторов**

Основные классификаторы и справочники в системе (пользователи, задачи, ответы) должны быть едиными.

##### **4.3.2.5 Требования по применению систем управления базами данных**

Специальных требований не предъявляется.

##### **4.3.2.6 Требования к структуре процесса сбора, обработки, передачи данных в системе и представлению данных**

Процесс сбора, обработки и передачи данных в системе определяются регламентом процессов сбора, преобразования и загрузки данных, разрабатываемом на этапе "Проектирование. Разработка эскизного проекта. Разработка технического проекта".

#### **4.3.2.7 Требования к защите данных от разрушений при авариях и сбоях в электропитании системы**

При аварийной ситуации, связанной со сбоем электропитания, информация в базе данных системы обязана сохраняться.

Желательно использование бесперебойного электропитания, обеспечивающего её нормальное функционирование в течение 15 минут в случае отсутствия внешнего энергоснабжения, и 5 минут дополнительно для корректного завершения всех процессов.

Резервное копирование данных должно осуществляться на регулярной основе, в объемах, достаточных для восстановления информации в подсистеме хранения данных.

#### **4.3.2.8 Требования к контролю, хранению, обновлению и восстановлению данных**

К контролю данных предъявляются следующие требования: сохранение всех событий, связанных с изменением своего информационного наполнения с целью возможного восстановления своего предыдущего состояния в случае сбоя.

#### **4.3.2.9 Требования к процедуре придания юридической силы документам, продуцируемым техническими средствами системы**

Специальных требований не предъявляется.

#### **4.3.3 Требования к лингвистическому обеспечению**

При реализации системы могут применяться следующие языки высокого уровня: PHP, SQL, HTML. Для реализации алгоритмов манипулирования данными в системе необходимо использовать стандартный язык запроса к данным SQL и его процедурное расширение.

#### **4.3.4 Требования к программному обеспечению**

СУБД должна иметь возможность инсталляции на ОС Windows (XP и более поздние версии), Linux (Ubuntu, Solaris).

К обеспечению качества ПС предъявляются следующие требования: выполнение подсистемами их функций обеспечивает функциональность системы; предупреждение или недопущение ошибок обеспечивает надежность; принятие верных и подходящих решений на всех этапах разработки ПС обеспечивает эффективность; техническое руководство и технологическая инструкция должны обеспечивать сопровождение системы.

#### **4.3.5 Требования к техническому обеспечению**

Веб-приложение имеет трехзвенную архитектуру: серверная часть состоит из сервера базы данных и сервера приложений, клиентской частью является браузер пользователя.

##### **Минимальные аппаратные требования:**

Сервер баз данных:

- Процессор - 4 x 3 ГГц;
- Объем оперативной памяти - не менее 2 Гб;
- Объем жесткого диска - не менее 80 Гб;
- Сетевая карта - с поддержкой скорости не менее 1 Гбит/сек.

Сервер приложений.

- Процессор - 4 x 3 ГГц;
- Объем оперативной памяти - не менее 2 Гб;
- Объем жесткого диска - не менее 40 Гб.

##### **Требования к рабочему месту пользователя:**

Процессор Intel-совместимый, тактовая частота не ниже 500 МГц, оперативная память не менее 256 Мб, свободного дискового пространства не менее 100 Мб.

#### **4.3.6 Требования к метрологическому обеспечению**

Специальных требований не предъявляется.

#### **4.3.7 Требования к организационному обеспечению**

Специальных требований не предъявляется.

#### **4.3.8 Требования к методическому обеспечению**

В состав нормативно-правового и методического обеспечения системы должны входить следующие законодательные акты, стандарты и нормативы:

ГОСТ 34.601-90. Автоматизированные системы. Стадии создания.

ГОСТ 34.602-89. Техническое задание на создание автоматизированной системы.

#### **4.3.9 Требования к патентной чистоте**

Специальных требований не предъявляется.

5 Состав и содержание работ по созданию системы

Работы по созданию системы выполняются в три этапа:

- 1 Проектирование. Разработка эскизного проекта. Разработка технического проекта.
- 2 Разработка рабочей документации. Адаптация программы.
- 3 Ввод в эксплуатацию.

Конкретные сроки выполнения стадий и этапов разработки и создания Системы определяются Планом выполнения работ.

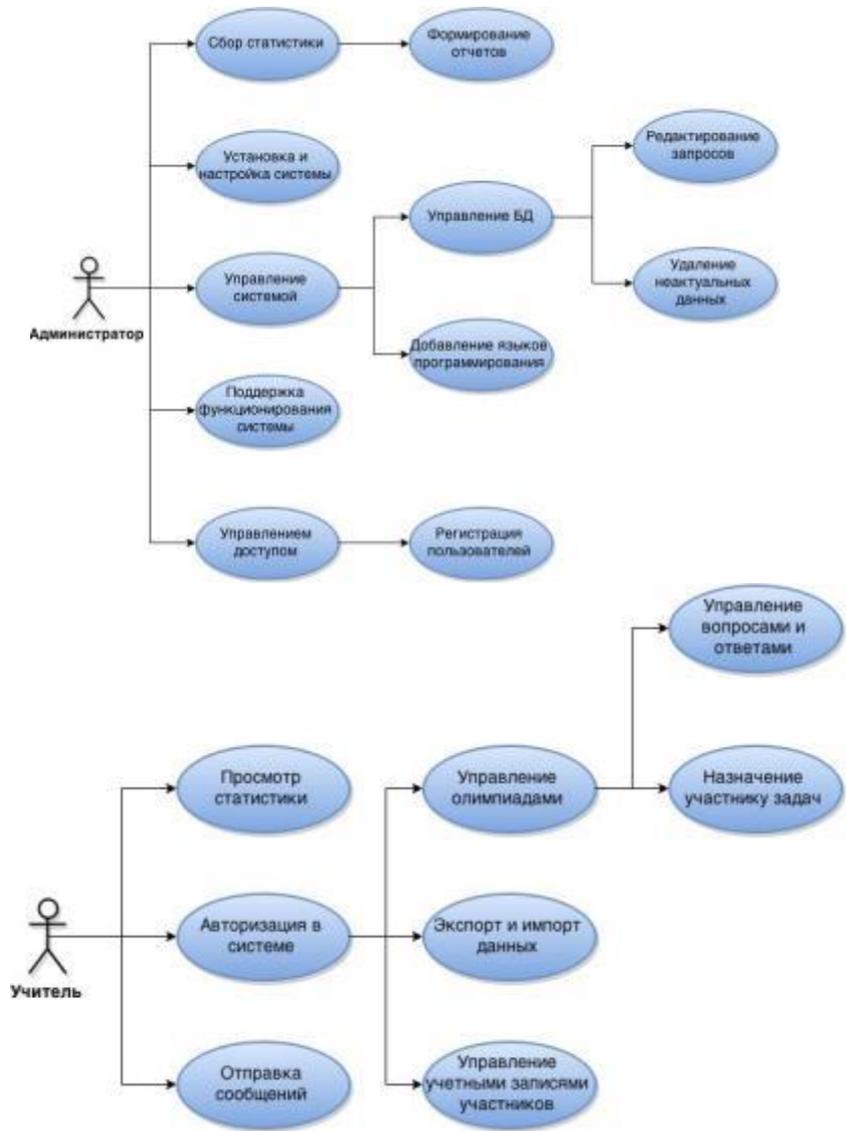
6 Порядок контроля и приёмки системы

6.1 Виды и объем испытаний системы

Система должна пройти три этапа испытаний:

- 1 Предварительные испытания.
- 2 Опытная эксплуатация.
- 3 Приемочные испытания.

Состав, объем и методы предварительных испытаний системы определяются документом "Программа и методика испытаний", разрабатываемым на стадии "Рабочая документация".



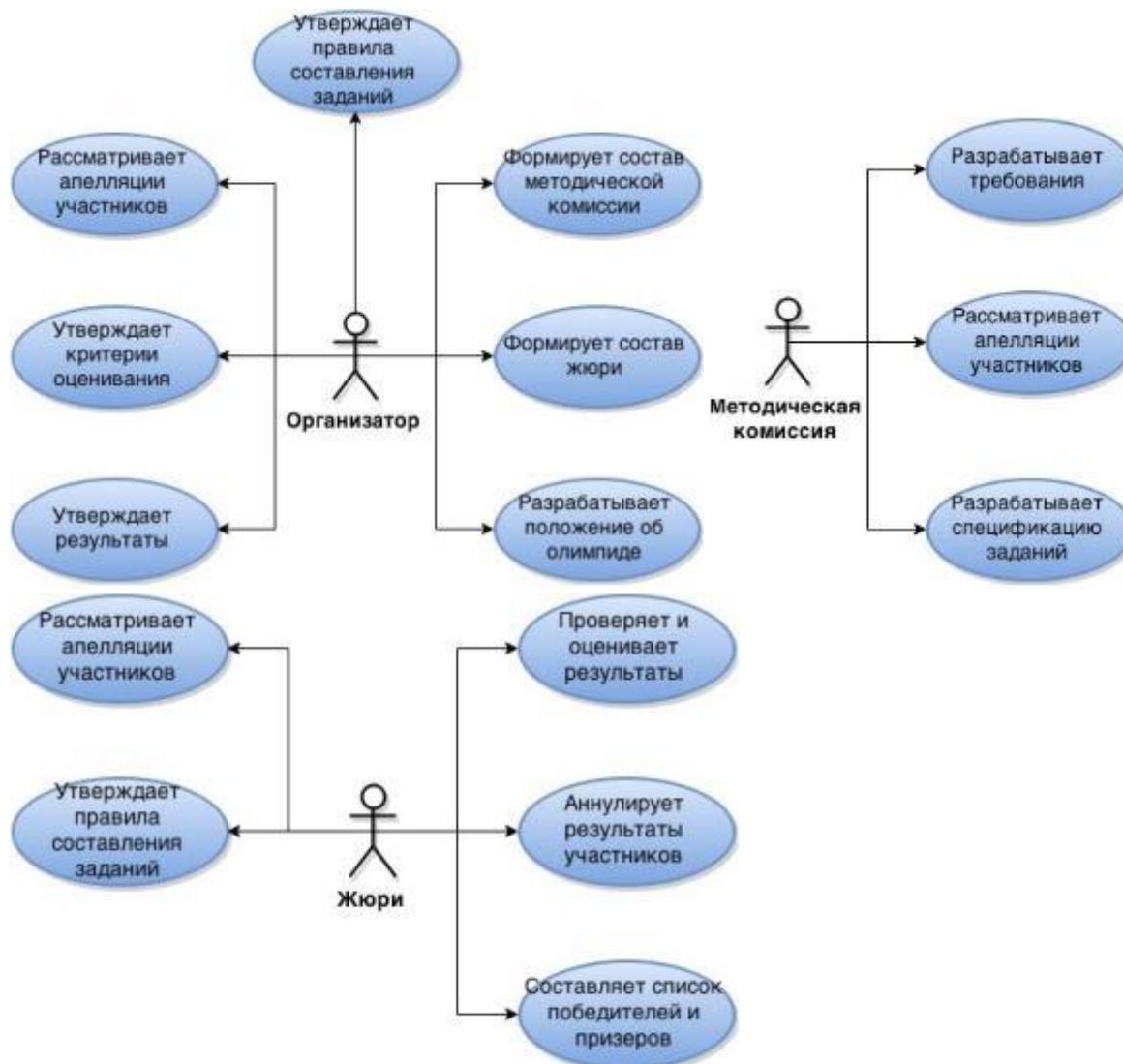


Рисунок 8. Диаграмма прецедентов "Жюри"

**Текущий контроль  
бсеместр  
Практическое занятие  
форма текущего контроля**

**по теме:** Формирование предложений о расширении информационной системы  
**Цель:** сформировать предложения о расширении информационной системы

**По завершению практического занятия студент должен уметь:** формировать предложения о расширении информационной системы

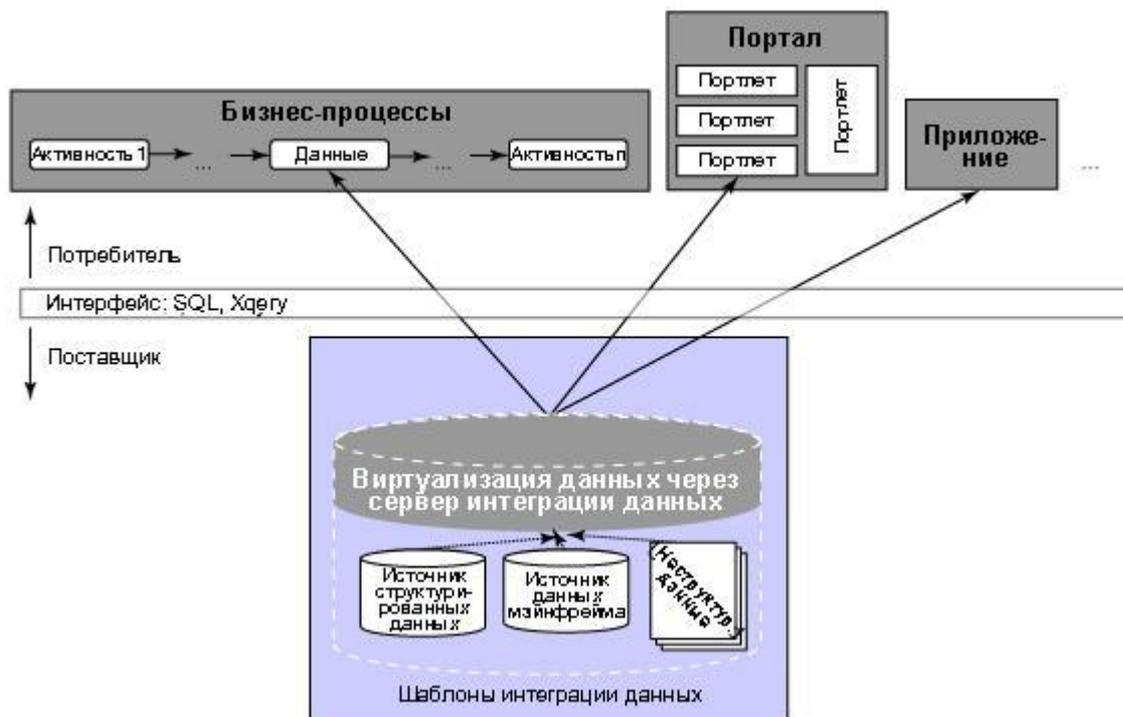
Продолжительность: 3 аудиторных часа (135 минут)

**Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

**Задание**

Одной из главных проблем интеграции данных является обилие форматов и типов (неструктурированные, частично-структурированные, жёстко-структурированные) данных, а также лавинообразное нарастание их объёмов. Циркулирование разнородных массивов данных и информации в сетях различных служб предприятия создает множество проблем с их сбором, структурированием, обработкой, анализом, хранением, архивированием и передачей пользователю для принятия делового решения. На рисунке 3.4 показана традиционная схема интеграции данных.



**Рис. 3.4.** Традиционная схема интеграции данных

Для их интеграции в настоящее время обычно используют стандартные интерфейсы и протоколы, например, SQL и JDBC/ODBC, применяют различные инструменты реляционных баз данных (Relational Database — RD), сквозных репозиториев — баз данных с "настройкой", содержащей информацию об артефактах и объектах проектирования, надмножество словарей метаданных (Transparent Repository — TR) и современных хранилищ и фабрик данных (Data Warehouse, Data Factory — DW, DF).

Последний вид технологий интеграции применяется, как правило, в крупных компаниях и производственных объединениях. Такие технологии создают удобную для пользователя единую среду для хранения и использования данных. Ниже будет подробнее рассказано о системах коллективного использования информации.

#### **Интеграция на уровне физических, программных и пользовательских интерфейсов**

Этот вид интеграции начинался как один из видов "лоскутной интеграции", когда предпринимались попытки объединить разрозненные программные приложения, написанные в разное время разными разработчиками, в подобие единого целого. Приложения объединялись по принципу "каждый с каждым", что, в конечном счёте, усложняло их взаимодействие и создавало массу проблем. Кроме того, всё сложнее становилось использовать унаследованные (Legacy Software) и встроенные (Embedded System) системы.

Такой подход хорош для небольшого количества приложений. При большом их числе он практически не работает и не позволяет строить качественно новые запросы к агрегированным данным, т.е. существенного выигрыша от объединения данных нет. В настоящее время проблема интеграции на уровне интерфейсов решается на базе использования информационных подсистем, реализованных стандартными программными приложениями с открытыми интерфейсами (Open Application Programming Interface).

Подобные унифицированные интерфейсы разрабатываются, например, на базе семейства международных стандартов POSIX. В этом случае степень интегрируемости можно характеризовать некоторым числовым показателем (метрикой) который можно, условно говоря, вычислить, перемножив показатель "качества" и "показатель открытости" программного интерфейса. Показателем качества могут выступать такие характеристики, как "совместимость", "надёжность", "переносимость", "понятность", "удобство использования" и пр. В результате мы получим индекс, который (в известной степени) характеризует способность приложения быть частью какого-то другого, глобального композитного приложения.

В настоящее время всё чаще применяется следующий алгоритм: отделяют слой обработки данных от привязанных к ним форм визуализации и реализуют прикладную бизнес-логику на одном из языков третьего поколения (3GL), оформив программный доступ к прикладным функциям в виде хорошо документированного программного интерфейса (рис. 3.5).

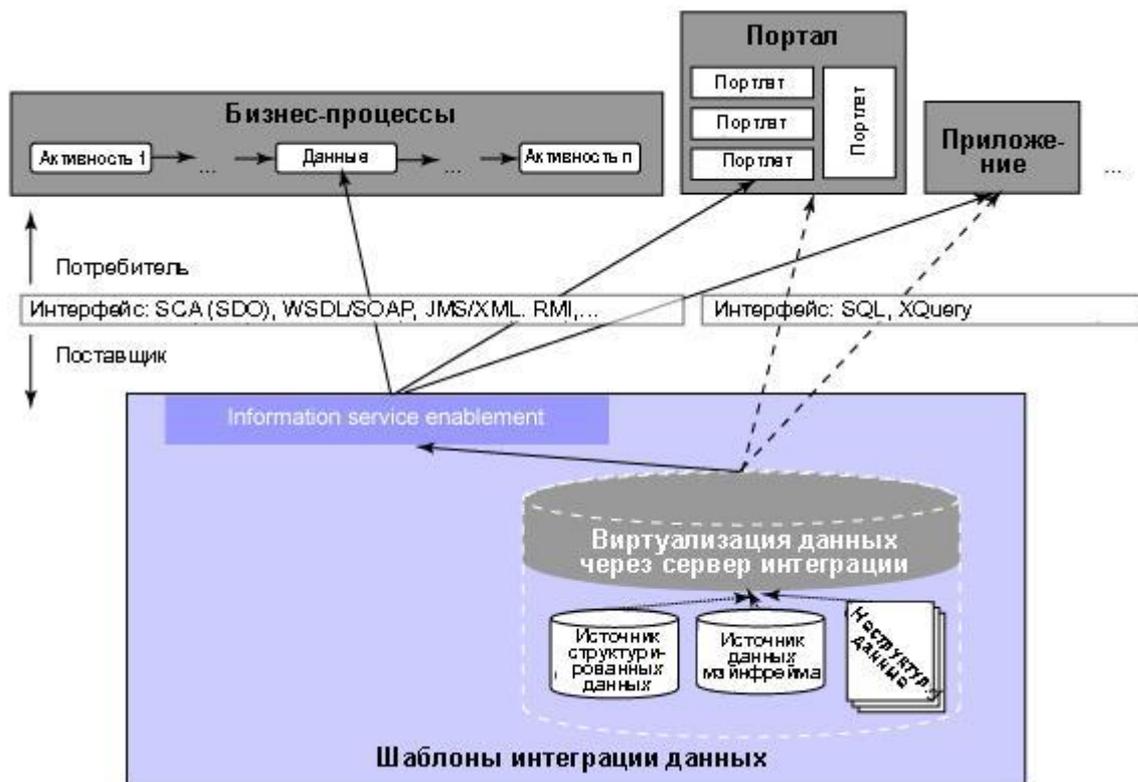


Рис. 3.5. Организация доступа к интегрированным данным через открытые интерфейсы

#### Интеграция на функционально-прикладном и организационном уровнях

Этот вид интеграции предполагает объединение ряда однотипных или схожих функций в макрофункции с перераспределением потоков данных и управления, а также ресурсов и механизмов для исполнения. Это часто влечёт за собой перестройку организационных структур, бизнес-процессов и, соответственно, схему их информационного и документационного обеспечения.

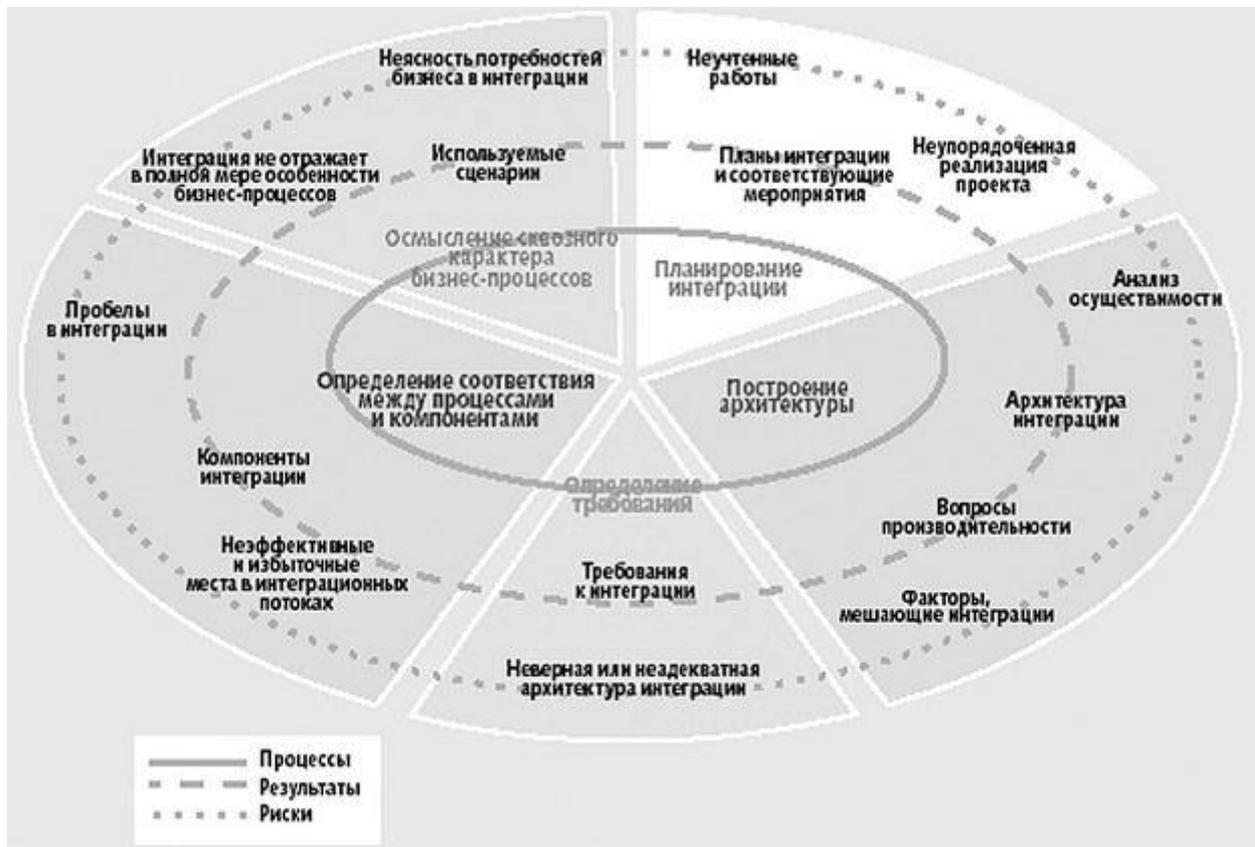
Выгоды от такой интеграции очевидны — процессы становятся более прозрачными, управляемыми, менее затратными, уменьшается количество обслуживающего персонала, число ошибок при формировании документов и т.д. Однако интеграция такого вида влечёт за собой существенную перестройку или полный реинжиниринг сети процессов, что связано с крупными рисками. Чаще всего такая интеграция проводится в том случае, когда предприятие готовится к внедрению КИС на базе известного решения, которое требует привести бизнес-процессы к требуемому стандарту, или перестраивает свою деятельность в связи со сменой устремлений, открытием филиалов в других странах, освоением новых сегментов рынка и т.д.

#### Интеграция на уровне корпоративных программных приложений

Интеграция на уровне приложений (Enterprise Application Integration — EAI), подразумевает совместное использование исполняемого кода, а не только внутренних данных интегрируемых приложений. Программы разбиваются на компоненты, которые интегрируются с помощью стандартизованных программных интерфейсов и специального связующего ПО.

При таком подходе из этих компонентов создается универсальное программное ядро или платформа, с помощью которых используют все приложения. Для каждого приложения создается только один интерфейс для связи с этим ядром, что существенно облегчает задачу интеграции. Полученную в результате систему легче поддерживать и расширять. Повторное использование функций в рамках имеющейся среды позволяет значительно снизить время и стоимость разработки приложений. В этом случае анализ внутренней конструкции приложений — обязательный этап в оценке степени интегрируемости тех приложений, которые предполагается связывать в рамках того или иного проекта. Этот анализ усложняется тем, что обычно разработчики приложений, являющихся законченными программными продуктами, как правило, не показывают деталей внутренней конструкции приложений.

В связи с этим технология интеграции в настоящее время рассматривает не просто интеграцию приложений, но их интеграцию на базе интеграции бизнес-процессов — в этом случае следует говорить об интеграции на уровне всего предприятия (Enterprise Integration Methodology — EIM). Схема такой объединенной методологии показана на рисунке 3.6 [138].



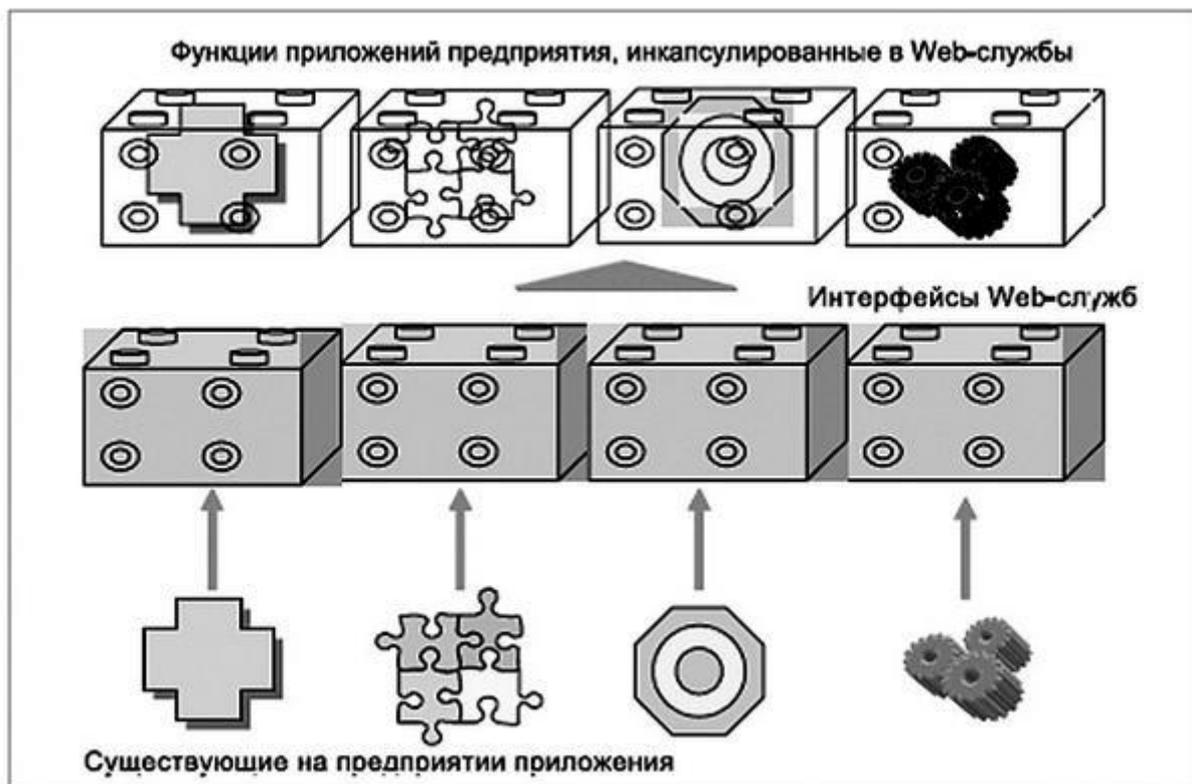
увеличить изображение

**Рис. 3.6.** Схема применения методологии EIM

Методология EIM реализуется современными технологиями и инструментами, среди которых можно, например, указать рассмотренную выше технологию интеграции на базе сервис-ориентированных архитектур (SOA). Архитектура ИС в таком случае строится из набора гетерогенных слабосвязанных компонентов (сервисов) и понимается как парадигма организации и использования распределенного множества функций, которые могут контролироваться различными владельцами. Базовыми понятиями в такой архитектуре являются "информационная услуга" и "компонитное приложение".

#### Интеграция при помощи Web-сервисов

Самый современный и быстро развивающийся подход к интеграции приложений. Он основан на обеспечении стандартного для Web-служб интерфейса доступа к приложениям и данным (рис.3.3.5).



**Рис. 3.7.** Схема доступа с использованием Web-служб

Например, используя стандартный протокол доступа к объектам SOAP (Simple Object Access Protocol), браузер пользователя может сравнить данные на нескольких сайтах и представить клиенту сравнительный отчет. Другой пример — сотрудники территориально распределенного предприятия могут одновременно использовать корпоративные приложения, доступ к которым осуществляется через соответствующие Web-сервисы (портальное решение).

Web-сервисы напоминают подход EAI, но с одним важным отличием — в большинстве случаев EAI-решения разрабатываются как частные для связи конкретных продуктов. Соответственно, подключить к существующему EAI-решению еще одну систему — достаточно трудная и долговременная задача. Web-сервисы существенно более унифицированы и стандартизованы. Поскольку Web-сервисы основаны на общих для W3C-консорциума стандартах, они могут работать всюду, где используется всемирная паутина (WWW). Результаты построения КИС на основе Web-интеграции:

- возможность осуществлять оперативное управление распределенной компанией и ведение консолидированного управленческого учета по нескольким филиалам;
- возможность осуществлять планомерное развитие общекорпоративной информационной системы, интегрируя в нее функциональные компоненты, исходя из приоритетов развития бизнеса компании и потребностей функциональных подразделений, т.е. возможность синхронизировать развитие системы с развитием бизнеса;
- возможность при необходимости заменить любой функциональный компонент другим, более соответствующим текущим бизнес-потребностям;
- возможность инвестировать в развитие информационных технологий не сразу, а поэтапно, на каждом этапе соотнося вложенные средства с полученным бизнес-эффектом, а также снижать общую стоимость автоматизированного рабочего места, включая затраты на создание системы, поддержку рабочих мест и обучение пользователей;
- резкое снижение времени сбора информации, необходимой для принятия управленческих и деловых решений, сокращение времени и трудозатрат на ведение учетных операций, на формирование промежуточных отчетов, на сверку информации между подразделениями и ликвидация противоречивости и несовместимости данных от различных служб;
- сохранение инвестиций в имеющиеся системы и оборудование, в обучение персонала.

В настоящее время крупные разработчики программных продуктов предлагают консолидированные решения, которые содержат не только конкретные инструменты для разработки и внедрения изначально интегрированных корпоративных приложений, но и реализуют интегрированную среду разработки таких приложений. Примером такого решения может служить программный продукт IBM WebSphere (рис. 3.8).

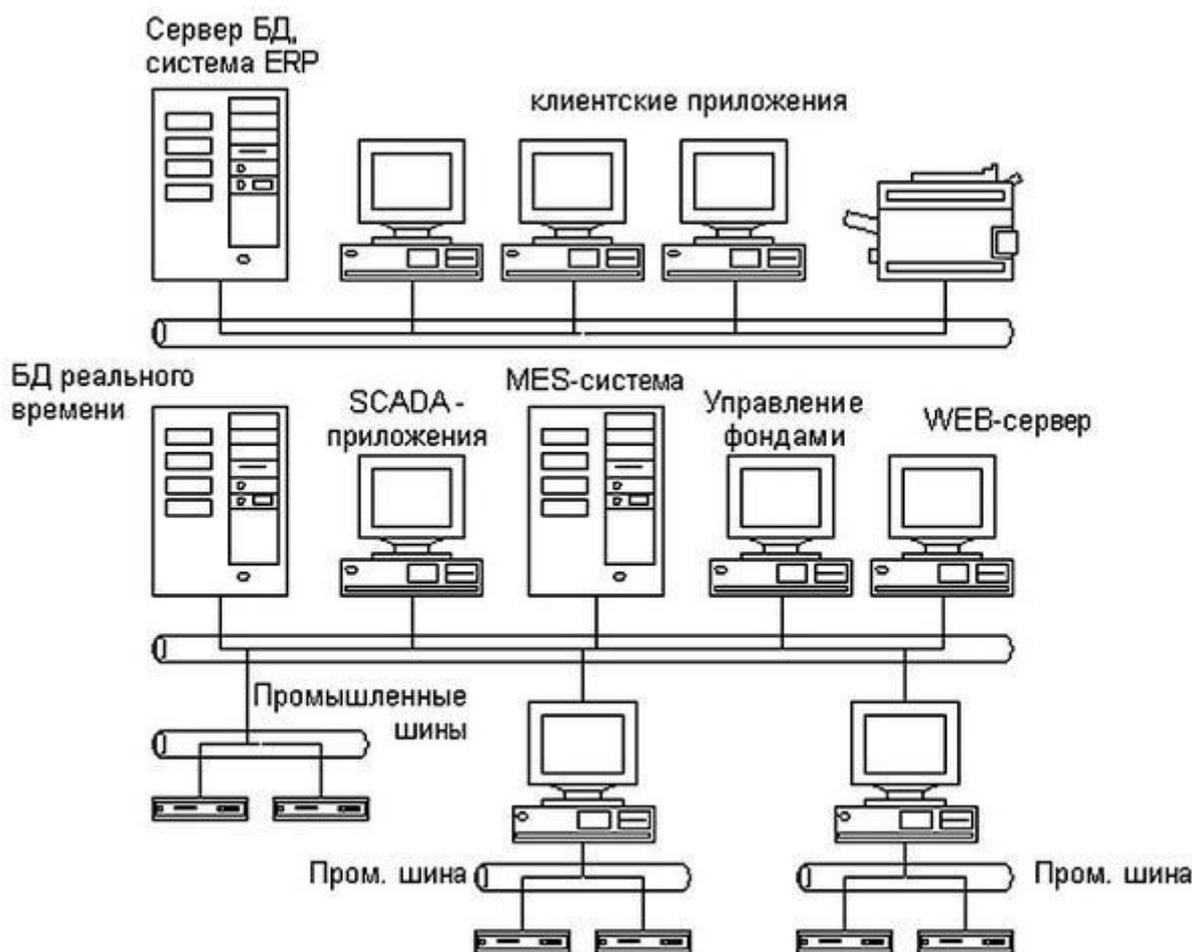


Рис. 3.8. Архитектурная модель WebSphere Application Server

**Текущий контроль**  
**бсеместр**  
**Лабораторная работа**  
*форма текущего контроля*

**по теме:** Обслуживание системы отображения информации актового зала  
**Цель:** Обслуживать системы отображения информации актового зала

**По завершению практического занятия студент должен уметь:** разрабатывать техническое задание на сопровождение информационной системы (указать предметную область)

Продолжительность: 4 аудиторных часа (180 минут)

### Необходимые принадлежности

Персональный компьютер, программное обеспечение.

### Задание

Разработка рабочего проекта и сметной документации по организации конференц-зала административного здания, расположенного по адресу: .

#### Этапы работ:

4.1. Этап №1. Разработка технологического решения.

4.2. Этап №2. Разработка рабочего проекта и сметной документации **по организации** конференц-зала.

#### 5. Состав работы:

5.1 В рамках первого этапа разрабатывается технологическое решение, которое должно включать в себя:

- Общую планировку 3-го этажа юго-восточного крыла здания Правительства Амурской области, где предполагается размещение конференц-зала.

- Планировочное решение в помещениях №№ 000а, 346б, 351, 351а, 353, 353а с учетом размещения мебели, основного оборудования, конференц-системы, кабель-трасс, систем вентиляции, кондиционирования, электропитания, отопления, освещения, пожаротушения.

- 3-х мерную визуализацию интерьера конференц-зала (включая отдельное помещение серверной) с учетом текстуры отделки, "одежды" стен, планов потолка, планов пола, типов отделок, штор, мебели, основного оборудования, мебели.

- Предоставление визуализации и точных планировочных решений по каждой из стен (развертка).

5.2. Рабочий проект разрабатывается только по утвержденному заказчиком технологическому решению. В рамках второго этапа, кроме обязательных разделов проекта согласно Постановлению Правительства РФ от 01.01.2001г. №87, рабочий проект должен включать в себя следующие специальные разделы:

- Общестроительные работы (АС);

- Оборудование (ТХ);

- Система кабельной канализации (СКК);

- Информационная кабельная система (ИКС);

- Локально-вычислительная сеть (ЛВС);

- Сеть охранной сигнализации (ОС);

- Сеть пожарной сигнализации (ПС);

- Система пожаротушения (ПТ);

- Сеть тревожной сигнализации (ТС);

- Контрольно-поисковая система (КПС);

- Система оповещения (ОП);

- Система IP-АТС (АТС);

- Конференц-система (КЦС);

- Система видеоконтроля и визуализации (ВН);

- Система вентиляции (ОВ);

- Система отопления (ОВ);

- Система мульти-зонального кондиционирования (ВК);

- Сеть внешнего энергоснабжения (ВЭО);

- Сеть энергоснабжения системы кондиционирования (ЭОК);

- Сеть энергоснабжения общего назначения (ЭОР);

- Система освещения (ЭО);

- Сеть бесперебойного электроснабжения (СБЭ).

6. Общие требования к выполняемым работам:

6.1 Требования к технологическому решению.

6.1.1 Технологическое решение должно учитывать все уже реализованные заказчиком решения в административном здании по адресу: ул. Ленина, 135.

6.1.2 Технологическое решение должно включать размещение оборудования и рабочих мест, мебели с учётом статуса и специфики конференц-зала Правительства Амурской области. Заказчику должна быть представлена визуализация 3-х мерной точной математической модели конференц-зала в вариантах естественного и искусственного освещения, выполненных с четырёх углов комнат и со средних точек комнат с высот 170 см и 120 см.

6.1.3 Технологическое решение должно отображать текстуру и цвет материала стен, потолка, пола, окон. При этом должны учитываться типы планируемых отделок, шторы, установленная мебель, расположение светильников, оборудования. Кроме того, визуализация выполняется с учетом расположения и объемов будущих вентиляционных систем и систем отопления и кондиционирования.

6.1.4 Технологическое решение должно учитывать внешние отводы кабельных трасс в общую систему, отводы трасс в систему кондиционирования, отводы трасс в компьютерные системы, отводы трасс вентиляции.

6.1.5 Качество предоставляемых изображений должно быть фотореалистичным, с учётом вторичных отражений, переотражений, преломлений, физически корректных свойств материалов и источников света.

6.1.6 Технологическое решение предоставляется заказчику для утверждения в виде презентационного материала с качеством изображений не менее 1920x1080 точек и на бумажных носителях на глянцевой или матовой фотобумаге формата А4.

6.2 Требования к Рабочему проекту.

6.2.1 Рабочий проект выполняется только после утверждения технологического решения Заказчиком.

6.2.2 Рабочий проект должен быть составлен с обязательным учётом требований согласно:

- Постановление Правительства РФ от 01.01.2001г. №87

- СНиП "Общественные здания административного назначения",

- СНиП 4-14-84 "Строительные нормы и правила",

- СНиП 3.04.01-87 "Изоляционные и отделочные покрытия"

- СНиП 3.05.06.-85 "Электротехнические устройства"

- Правила устройства электроустановок;
- СНиП 2.01.02-85 "Противопожарные нормы";
- Стандарты и требования фирм-изготовителей оборудования;
- ГОСТ Р 21. "Правила выполнения рабочей документации проводных средств связи";
- ГОСТ 21.101-97 "Основные требования к проектной и рабочей документации";
- СНиП "Естественное и искусственное освещение";
- СНиП 2.04.01-85 "Внутренний водопровод и канализация зданий";
- СНиП "Отопление, вентиляция, кондиционирование".
- НПБ 88-2001 "Установки пожаротушения и сигнализации. Нормы и правила проектирования".

6.3. Общие требования к общестроительным работам.

6.3.1. Результатом проектных работ должно быть получение архитектурного и планировочного вида конференц-зала в полном соответствии с утвержденным Технологическим решением.

6.3.2. Все проектируемые материалы должны соответствовать требованиям, принятым в РФ и должны иметь соответствующие сертификаты безопасности и качества.

6.4 Общие требования к разделу оборудования (ТХ).

6.4.1 В раздел оборудования включить все оборудование, не вошедшее в специальные и специализированные системы проекта, мебель, инвентарь на основании утвержденного технологического решения.

6.4.2 В спецификации раздела оборудования обязательным является указание производителя оборудования и описание полных качественных характеристик. Оборудование должно соответствовать требованиям, принятым в РФ и должно иметь соответствующие сертификаты безопасности и качества.

6.5 Общие требования к конференц-системе (КЦС).

6.5.1 Конференц - система должна обеспечивать проведение совещаний с количеством участников до 90 человек с зонированием зала на две части. Первая часть с количеством не менее 37 мест на едином рабочем столе. Вторая часть – общий зал для приглашенных. Конференц-система должна быть выполнена с учетом минимального количества прокладываемых кабелей, с учетом согласованности имеющегося в Правительстве Амурской области оборудования, с возможностью включения в систему видеоконференц-связи, с учетом использования не менее 12 мест для синхронного перевода и обеспечения каждого участника индивидуальным аудиокomплектом от необходимого переводчика, обеспечивать отображение выступающего на хорошо просматриваемых с каждой точки зала панелях отображения, иметь качественную систему озвучивания и должна соответствовать требованиям:

Постановление Правительства РФ от 01.01.2001г. №87

НПБ 88-2001 – «Установки пожаротушения и сигнализации.

СНиП 3.05.06.-85 – «Электротехнические устройства»,

СНиП 3.05.07.-85 – «Системы автоматизации»,

СНиП 2.01.02.-85 – «Противопожарные нормы»,

СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,

СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,

ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,

ГОСТ 21.613-88 – «Силовое электрооборудование»,

ГОСТ-21.614-88 – «Система проектной документации для строительства»,

ГОСТ– «Строительство. Электробезопасность. Общие требования»,

ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,

ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,

СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,

СЕИ ИЕС -950, а также Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.5.2. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.6 Требования к системе кабельной канализации (СКК).

6.6.1. Сеть кабельной канализации предназначена для прокладки всех слаботочных сетей, всех кабельных сетей, сетей электропитания и монтажа розеточных блоков.

6.6.2. СКК должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87

-ПУЭ,

-НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,

-СНиП 3.05.06.-85 – «Электротехнические устройства»,

-СНиП 3.05.07.-85 – «Системы автоматизации»,

-СНиП 2.01.02.-85 – «Противопожарные нормы»,

-СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,

-СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,

-ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,

-ГОСТ 21.613-88 – «Силовое электрооборудование»,

-ГОСТ-21.614-88 – «Система проектной документации для строительства»,

-ГОСТ– «Строительство. Электробезопасность. Общие требования»,

-ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,

-ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,

-СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,

-СЕИ ИЕС -950, а также Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.6.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

6.7 Требования к информационной кабельной системе (ИКС).

6.7.1 ИКС должна включать в себя медные кабели, кроссовое оборудование, корды, защитные панели, кабельные тестеры, крепёжные и установочные изделия.

6.7.2. Проектируемая информационная кабельная сеть (ИКС) должна соответствовать требованиям следующих Международных и Европейских стандартов на СКС 5 категории и выше:

- Постановление Правительства РФ от 01.01.2001г. №87
  - ISO/IEC 11801. Информационная технология - Универсальная кабельная система для зданий и территории Заказчика;
  - EIA/TIA-568A. Стандарт по телекоммуникационным кабельным системам в коммерческих зданиях;
  - EIA/TIA-569. Стандарт по телекоммуникационным кабельным трассам и помещениям в коммерческих зданиях;
  - EIA/TIA-606. Стандарт по администрированию телекоммуникационных инфраструктур;
- 6.7.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.
- 6.7.4 Требования к IP-АТС (АТС).
- 6.7.5. АТС должна обеспечивать цифровой связью не менее десяти точек в конференц-зале и быть интегрирована в общую IP-систему Правительства Амурской области.
- 6.7.6. Проектируемая IP-АТС (АТС) должна соответствовать требованиям следующих Международных и Европейских стандартов выше:
- Постановление Правительства РФ от 01.01.2001г. №87
  - ISO/IEC 11801. Информационная технология - Универсальная кабельная система для зданий и территории Заказчика;
  - EIA/TIA-568A. Стандарт по телекоммуникационным кабельным системам в коммерческих зданиях;
  - EIA/TIA-569. Стандарт по телекоммуникационным кабельным трассам и помещениям в коммерческих зданиях;
  - EIA/TIA-606. Стандарт по администрированию телекоммуникационных инфраструктур;
- 6.7.7. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.
- 6.8 Требования к локально-вычислительной сети (ЛВС)
- 6.8.1. Локальная вычислительная сеть предназначена для объединения информационно-вычислительных ресурсов помещения Заказчика.
- 6.8.2. При этом сеть должна обеспечить:
- Высокую степень защиты, надежности и производительности своей работы;
  - Мобильность пользователей;
  - Минимизацию времени устранения аварийных ситуаций.
- 6.8.3. ЛВС должна функционировать в круглосуточном режиме, обеспечивая работу не менее 37 пользователей, поддерживать возможность увеличения их количества на 30% по отношению к первоначальному и включать в себя активное сетевое оборудование и систему управления сетью.
- 6.8.4. Должны быть разработаны возможности по расширению и модернизации сети, развитию ее функциональных возможностей.
- 6.8.5. В проектируемой ЛВС необходимо учитывать характеристики существующих у Заказчика аппаратно-программных комплексов для обеспечения их интеграции с новой сетевой инфраструктурой.
- 6.8.6. На этапе проектирования по отдельному поручению Заказчика может быть построена программная модель сети для анализа ее работы под нагрузкой для моделирования возникновения аварийных ситуаций, а также возможностей по ее расширению и повышению пропускной способности. С использованием результатов моделирования могут быть произведены:
- Расчет пропускной способности ЛВС (исходные данные для расчета уточняются на этапе проектирования);
  - Разработка возможных сценариев возникновения аварийных ситуаций работы сети и их устранения;
  - Обоснование предлагаемых решений по выбору архитектуры сети, активного сетевого оборудования и его характеристик. В ходе проектирования должно быть произведено обследование объекта для уточнения исходных данных для построения ЛВС. Проект не должен включать в себя предложения и расходы по обучению специалистов заказчика по эксплуатации активного сетевого оборудования, системы управления и ЛВС в целом.
  - Интегрированную передачу голосовых, видео - и цифровых данных;
  - Построение виртуальных сетей;
  - Соглашения об уровнях сервисов;
  - Учет используемых ресурсов;
- 6.8.7. Проектируемая ЛВС должна обладать следующими основными свойствами:
- Надежность;
  - Защищенность;
  - Производительность;
  - Управляемость;
  - Масштабируемость.
- 6.8.8. При выработке решения по построению ЛВС должны также учитываться стоимостные показатели. ЛВС может быть построена по иерархическому и модульному принципу с обеспечением минимального влияния возникновения неисправностей на нижних уровнях иерархии как друг на друга, так и на верхние уровни. В ЛВС должны использоваться технологии коммутации, маршрутизации, построения виртуальных сетей и приоритизации передачи трафика.
- 6.8.9. Должны быть предусмотрены средства защиты ЛВС от несанкционированного доступа к сетевому оборудованию, сетевой среде и системе управления.
- 6.8.10. В проектируемой ЛВС могут использоваться следующие отказоустойчивые технические решения (отсутствие единой точки отказа): дублирование и резервирование сетевых устройств и их элементов, сетевой магистрали, автоматическое реконфигурирование сетевого оборудования в случае возникновения неисправностей в работе сети
- 6.8.11. ЛВС должна быть спроектирована с расчетом на централизованное подключение серверного парка и других информационно-вычислительных ресурсов общего пользования.
- 6.8.12. Активное сетевое оборудование должно удовлетворять следующим требованиям:
- Поддержка стандартов: IEEE 802.3 (Ethernet), IEEE 802.3u (Fast Ethernet), IEEE 802.3z (Gigabit Ethernet), IEEE 802.1Q (виртуальные сети), IEEE 802.1D (связующее дерево), SMNP (удаленный мониторинг);
  - Наличие встроенных средств самодиагностики и поддержка функций удаленного управления;
  - Возможность обновления микропрограммного обеспечения;
  - Поддержка режимов балансировки нагрузки по параллельным каналам.
  - Возможность перекоммутации кабельной системы «телефон-компьютер» на одном рабочем месте.
- 6.8.13. Условия эксплуатации сетевого оборудования должны соответствовать его техническим спецификациям.
- 6.8.14. Система управления сетью должна представлять собой комплекс приложений для управления сетевой инфраструктурой и обеспечивать:
- Реализацию клиент-серверной архитектуры платформы сетевого управления;
  - Наглядное графическое представление топологии, физической и организационной структуры сети;
  - Управление конфигурациями сетевого оборудования;
  - Быстрый поиск и обнаружение неисправностей в работе сети;

- Ведение журнала о происходящих в сети событиях и функционировании сетевого оборудования;
- Наличие средств обработки сетевых сообщений;
- Составление отчетов о работе сети;
- Автоматическое обнаружение сетевых устройств и построение карты сети;
- Возможность мониторинга работы рабочих станций, серверов и других информационно-вычислительных ресурсов;
- Возможность контроля за работой системы электропитания сетевого оборудования, доступом в коммутационные шкафы для размещения сетевого оборудования и параметрами окружающей среды в них;
- Выдачу предупреждений о наступлении критических событий в сети в соответствии с настраиваемыми пороговыми значениями;
- Управление виртуальными сетями;
- Возможность управления уровнем сервиса;
- Управление существующим у заказчика активным сетевым оборудованием, состав которого определяется на этапе проектирования.

6.8.15. Система управления сетью должна быть построена по модульному принципу и иметь возможности функционального расширения, а также иметь открытую архитектуру и средства собственной настройки. Должны быть предусмотрены следующие возможности функционального расширения системы управления:

- Средства инвентаризации программно-аппаратной конфигурации рабочих станций и серверов, их управления и автоматизированного развертывания на них программного обеспечения;
- Мониторинг ЛВС и информационно-вычислительных ресурсов.

6.8.16. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

## 6.9 Требования к сети охранной сигнализации (ОС).

6.9.1. Для обеспечения помещений Заказчика системой охранной сигнализации (ОС) предусмотреть распределительную сеть охранной сигнализации со своим коммутационно - распределительным и контрольным оборудованием.

6.9.2. ОС должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,
- СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.9.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.9.4. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

## 6.10 Требования к сети тревожной сигнализации (ТС).

6.10.1. Для обеспечения помещений Заказчика системой тревожной сигнализации (ТС) предусмотреть распределительную сеть тревожной сигнализации со своим коммутационно - распределительным и контрольным оборудованием.

6.10.2. ТС должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,
- СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.10.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

## 6.11 Требования к контрольно-поисковой системе (КПС).

6.11.1. Для обеспечения помещений Заказчика контрольно-поисковой системой (КПС) предусмотреть распределительную сеть контрольно-поисковой системы со своим коммутационно - распределительным и контрольным оборудованием для подачи аудиообъявлений и связи с определенным кругом должностных лиц.

6.11.2. КПС должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,

-СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,  
-СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,  
-ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,  
-ГОСТ 21.613-88 – «Силовое электрооборудование»,  
-ГОСТ-21.614-88 – «Система проектной документации для строительства»,  
-ГОСТ– «Строительство. Электробезопасность. Общие требования»,  
-ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,  
-ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,  
-СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,  
-СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.11.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.12 Требования к сети пожарной сигнализации (ПС).

6.12.1. Для обеспечения помещений Заказчика системой пожарной сигнализации предусмотреть распределительную сеть пожарной сигнализации со своим коммутационно - распределительным и контрольным оборудованием.

6.12.2. ПС должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,
- СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.12.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

6.13 Требования к системе пожаротушения (ПТ).

6.13.1. Предусмотреть систему пожаротушения для помещения серверной.

6.13.2. ПТ должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,
- СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.13.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

6.14 Требования к системе оповещения (ОП).

6.14.1. Предусмотреть локальную систему оповещения с подключением к общей системе оповещения здания. 6.8.17. ОП должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,

-СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.14.2. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию сторон.

6.15 Требования к системе видеоконтроля и визуализации (ВН).

6.15.1. Система ВН служит для регистрации визуальных событий и их дальнейшего просмотра.

6.15.2. ВН должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- ПУЭ, НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,
- СНиП 3.05.06.-85 – «Электротехнические устройства»,
- СНиП 3.05.07.-85 – «Системы автоматизации»,
- СНиП 2.01.02.-85 – «Противопожарные нормы»,
- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,
- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,
- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,
- ГОСТ 21.613-88 – «Силовое электрооборудование»,
- ГОСТ-21.614-88 – «Система проектной документации для строительства»,
- ГОСТ– «Строительство. Электробезопасность. Общие требования»,
- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,
- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,
- СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,
- СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.15.3.Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.16. Требования к системе вентиляции (ОВ).

6.16.1. Система вентиляции конференцзала здания Правительства Амурской области в городе Благовещенске должна обеспечивать комфортные условия для нахождения в помещениях сотрудников и посетителей.

6.16.2. Необходимо предусмотреть механическую приточно-вытяжную вентиляцию.

6.16.3. Система вентиляции должна обеспечивать влажность внутреннего воздуха в пределах 30-45 %.

6.16.4. Система вентиляции должна быть выполнена спирально-навивными круглыми воздуховодами. Вертикальные коллекторы выполнить прямоугольными оцинкованными воздуховодами.

6.16.5. Система вентиляции должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87
- ГОСТ 12.1.003-83 ССБТ «Шум. Общие требования безопасности»,
- ГОСТ 12.1.005-88 ССБТ «Общие санитарно-гигиенические требования к воздуху рабочей зоны»,
- ГОСТ «Здания жилые и общественные. Параметры микроклимата в помещениях»,
- ГОСТ «Решётки вентиляционные пластмассовые»,
- СНиП 2.08.02-89 «Общественные здания и сооружения»,
- СНиП «Пожарная безопасность зданий и сооружений»,
- СНиП «Строительная климатология»,
- СНиП «Защита от шума»,
- СНиП «Общественные здания административного назначения»,
- СНиП «Отопление, вентиляция, кондиционирование»,
- НПБ 105-03 «Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности»,
- НПБ 239-97 «Воздуховоды»,
- НПБ 241-97 «Клапаны противопожарные вентиляционных систем»,
- ПУЭ «Правила устройства электроустановок».

6.16.6.Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.17 Требования к системе отопления (ОВ).

6.17.1. Отопление конференцзала здания Правительства Амурской области в городе Благовещенске реализовать от системы централизованного водяного отопления.

6.17.2. Систему отопления выполнить двухтрубной с нижней разводкой магистральных трубопроводов, с учетом существующего теплового пункта.

6.17.3. В качестве горизонтальных трубопроводов магистральной разводки использовать металлические трубы марки ВГП. Стойки системы отопления и подводу к отопительным приборам выполнить металлопластиковыми трубами, обладающими кислородопроницаемостью не более 0,1 г(м<sup>3</sup>\*сут) и расчетным сроком службы не менее 40 лет, с замоноличиванием в стены в соответствии со СНиП «Отопление, вентиляция, кондиционирование».

6.17.4. В комплекте с полимерными трубами следует применять соединительные детали и изделия одного производителя.

6.17.5. В качестве нагревательных приборов предусмотреть современные алюминиевые, либо биметаллические секционные радиаторы увеличенной поверхности теплообмена с номинальным тепловым потоком одной секции не менее 195 Вт.

6.17.6. Для регулирования теплоотдачи установить регулирующие краны повышенного гидравлического сопротивления у отопительных приборов.

6.17.7. В целях удаления воздуха из системы отопления установить воздушные краны в верхних пробках крайних радиаторов радиаторных узлов на каждом этаже.

6.17.8. Система отопления конференц-зала здания Правительства Амурской области должна обеспечивать комфортные условия для нахождения в помещениях сотрудников и посетителей.

6.17.9.Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.18 Требования к системе мультizonального кондиционирования (ОВ).

6.18.1. Так как в помещении будет установлено оборудование, необходимо предусмотреть системы кондиционирования, обеспечивающие температуру от +18° С до +25°С и относительную влажность не более 30-45% без конденсата, используя настенные кондиционеры. Окружающая температура и влажность должны рассчитываться на расстоянии 1,5м от уровня пола, при включенном оборудовании. Кондиционеры должны обеспечивать необходимые условия микроклимата в течении всего рабочего дня. Необходимо учитывать, что тепловыделение от оборудования в расчётной конфигурации может достигать не менее 1,5кВт. Наружный блок системы кондиционирования не размещать на наружном фасаде здания.

6.18.2. Система кондиционирования должна соответствовать требованиям:

- Постановление Правительства РФ от 01.01.2001г. №87  
- ГОСТ «Здания жилые и общественные. Параметры микроклимата в помещениях»,  
- СНиП «Общественные здания административного назначения»,  
- СНиП «Отопление, вентиляция, кондиционирование»

6.18.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.19. Требования к сети электроснабжения общего назначения (ЭОР).

6.19.1. Для обеспечения электропитания бытового и иного оборудования, не запитываемого от сети бесперебойного электропитания, спроектировать распределительную электросеть общего назначения (розетки), со своим коммутационно - распределительным оборудованием.

6.19.2. ЭОР должна соответствовать требованиям:  
Постановление Правительства РФ от 01.01.2001г. №87  
НПБ 88-2001 – «Установки пожаротушения и сигнализации»  
СНиП 3.05.06.-85 – «Электротехнические устройства»,  
СНиП 3.05.07.-85 – «Системы автоматизации»,  
СНиП 2.01.02.-85 – «Противопожарные нормы»,  
СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,  
СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,  
ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,  
ГОСТ 21.613-88 – «Силовое электрооборудование»,  
ГОСТ-21.614-88 – «Система проектной документации для строительства»,  
ГОСТ– «Строительство. Электробезопасность. Общие требования»,  
ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,  
ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,  
СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,  
СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.19.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.20. Требования к сети внешнего энергоснабжения (ВЭО).

6.20.1. Сеть внешнего энергоснабжения предусмотреть для обеспечения транспорта электроэнергии от ГРЩ здания до распределительных устройств конференц-зала.

6.20.2. ВЭО должна соответствовать требованиям:  
Постановление Правительства РФ от 01.01.2001г. №87  
НПБ 88-2001 – «Установки пожаротушения и сигнализации»  
СНиП 3.05.06.-85 – «Электротехнические устройства»,  
СНиП 3.05.07.-85 – «Системы автоматизации»,  
СНиП 2.01.02.-85 – «Противопожарные нормы»,  
СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,  
СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,  
ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,  
ГОСТ 21.613-88 – «Силовое электрооборудование»,  
ГОСТ-21.614-88 – «Система проектной документации для строительства»,  
ГОСТ– «Строительство. Электробезопасность. Общие требования»,  
ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,  
ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,  
СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,  
СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.20.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.21. Требования к сети энергоснабжения системы кондиционирования (ЭОК).

6.21.1. Сеть энергоснабжения системы кондиционирования предусмотреть для обеспечения отдельного электропитания системы кондиционирования.

6.21.2. ЭОК должна соответствовать требованиям:  
Постановление Правительства РФ от 01.01.2001г. №87  
НПБ 88-2001 – «Установки пожаротушения и сигнализации»  
СНиП 3.05.06.-85 – «Электротехнические устройства»,  
СНиП 3.05.07.-85 – «Системы автоматизации»,  
СНиП 2.01.02.-85 – «Противопожарные нормы»,  
СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,  
СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,  
ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,  
ГОСТ 21.613-88 – «Силовое электрооборудование»,  
ГОСТ-21.614-88 – «Система проектной документации для строительства»,  
ГОСТ– «Строительство. Электробезопасность. Общие требования»,  
ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,  
ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,  
СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,  
СЕИ ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.21.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.22. Требования к сети освещения (ЭО)

6.22.1. Сеть освещения предусмотреть для обеспечения соответствия размещения светильников согласно технологическому решению.

6.22.2. ЭО должна соответствовать требованиям:

Постановление Правительства РФ от 01.01.2001г. №87

НПБ 88-2001 – «Установки пожаротушения и сигнализации.

СНиП 3.05.06.-85 – «Электротехнические устройства»,

СНиП 3.05.07.-85 – «Системы автоматизации»,

СНиП 2.01.02.-85 – «Противопожарные нормы»,

СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,

СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,

ВСН 59-88 – «Электрооборудование жилых и общественных зданий»,

ГОСТ 21.613-88 – «Силовое электрооборудование»,

ГОСТ-21.614-88 – «Система проектной документации для строительства»,

ГОСТ– «Строительство. Электробезопасность. Общие требования»,

ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,

ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,

СНиП – «о порядке разработки, согласования, утверждения и составе проектной документации на строительство предприятий, зданий и сооружений»,

СЕН ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.22.3. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

6.23. Требования к сети бесперебойного электроснабжения

6.23.1. Для обеспечения электропитания активного оборудования ЛВС, IP-АТС, оборудования безопасности, серверов и рабочих станций спроектировать отдельную от сетей общего назначения и освещения распределительную электросеть со своим коммутационно - распределительным оборудованием и источниками бесперебойного электроснабжения мощностью до 20 кВА с учетом 30% резерва мощности, в количестве не менее 1 шт.

6.23.2. СБЭ должна создаваться по схеме централизованных, не резервируемых ИБЭ (UPS) с возможностью параллельной работы на общую нагрузку.

6.23.3. СБЭ должна соответствовать требованиям

- Постановление Правительства РФ от 01.01.2001г. №87

- ПУЭ,

- НПБ 88-2001 – «Установки пожаротушения и сигнализации. Нормы и правила проектирования»,

- СНиП 3.05.06.-85 – «Электротехнические устройства»,

- СНиП 3.05.07.-85 – «Системы автоматизации»,

- СНиП 2.01.02.-85 – «Противопожарные нормы»,

- СНиП 3.01.04.-87 – «Приемка в эксплуатацию законченных строительством объектов. Основные положения»,

- СН 512-78 – «Инструкция по проектированию зданий и помещений для электронно-вычислительных машин»,

- ВСН 59-88 – «Электрооборудование жилых и общественных зданий»

- ГОСТ 21.613-88 – «Силовое электрооборудование»,

- ГОСТ-21.614-88 – «Система проектной документации для строительства»,

- ГОСТ– «Строительство. Электробезопасность. Общие требования»,

- ГОСТ 12.1.030-81 – «Электробезопасность. Защитное заземление. Зануление»,

- ГОСТ – «Требования к качеству электрической энергии в электрических сетях общего назначения»,

- СНиП – «о порядке разработки, согласования, утверждения и состав проектной документации на строительство предприятий, зданий и сооружений»,

- СЕН ИЕС -950, а также - Европейскому руководству по совместной прокладке проводов распределительных систем здания и кабелей электропитания AT&T Bell Laboratoris.

6.23.4. Вышеуказанные требования уточняются по результатам проектирования и взаимному согласованию с Заказчиком.

7. Сметный расчёт.

7.1. Каждой запроектированной системе должна соответствовать отдельная локальная смета. Сводный сметный расчет должен быть составлен с учетом норм, принятых на территории Амурской области для бюджетных организаций.

7.2. Исполнитель обязан предоставить Заказчику сводный сметный расчёт в сумме, не превышающей 30 млн. руб. с учётом всех выполняемых работ по разработанному технологическому решению и проектам.

7.3. Разработка сметной документации должна быть проведена на основе ТЕРов, действующих в южных районах Амурской области с учётом поправочных коэффициентов.

#### **Текущий контроль**

**бсеместр**

#### **Практическое занятие**

*форма текущего контроля*

**по теме:** Разработка технического задания на сопровождение информационной системы (указать предметную область)

**Цель:** разработка технического задания на сопровождение информационной системы (указать предметную область)

**По завершению практического занятия студент должен уметь:** разрабатывать техническое задание на сопровождение информационной системы (указать предметную область)

Продолжительность: 2 аудиторных часа (90 минут)

#### **Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

#### **Задание**

В настоящем регламенте технического обслуживания средств вычислительной техники излагаются порядок и правила выполнения работ по техническому обслуживанию средств вычислительной техники. Требования регламента обязательны для всех организаций, производящих техническое обслуживание средств вычислительной техники.

При выполнении технического обслуживания должны дополнительно использоваться эксплуатационные документы на изделия, входящие в состав средств вычислительной техники.

Техническое обслуживание средств вычислительной техники направлено на обеспечение постоянной готовности оборудования к использованию по прямому назначению и предотвращение преждевременного выхода его из строя.

В состав технического обслуживания средств вычислительной техники входят следующие виды работ:

консультации специалистов заказчика по всем вопросам эксплуатации средств вычислительной техники и системного программного обеспечения;

установка средств вычислительной техники и включение их в работу на рабочих местах заказчика;

передача неисправной техники, находящейся на гарантийном обслуживании, в гарантийный ремонт, получение ее из ремонта и установка на рабочих местах заказчика;

техническое обслуживание по вызову заказчика для оперативного восстановления работоспособности средств вычислительной техники;

ремонт средств вычислительной техники в лабораторных условиях;

плановое месячное техническое обслуживание средств вычислительной техники.

Минимальная периодичность проведения технического обслуживания средств вычислительной техники - один раз в месяц. Срок проведения очередных работ определяется с учетом обслуживания по срочным вызовам пользователя, если это обслуживание включало работы, входящие в состав планового месячного технического обслуживания. В случае выявления дефектов оборудования, которые не могут быть устранены на площадке пользователя, оно (с согласия материально - ответственного лица пользователя) должно быть изъято для ремонта в лабораторных условиях.

Комплекующие и расходные материалы, необходимые для выполнения технического обслуживания средств вычислительной техники, предоставляются заказчиком, и их стоимость не входит в стоимость работ по техническому обслуживанию средств вычислительной техники.

Если в процессе проведения технического обслуживания средств вычислительной техники выявляются факты нарушения правил технической эксплуатации оборудования или правил техники безопасности пользователем, персонал, производящий техническое обслуживание, должен принять соответствующие меры к их немедленному устранению и недопущению впредь, и в установленном порядке проинформировать об этом заказчика.

Техническое обслуживание персонального компьютера:

проверка работоспособности устройств на тестах в ускоренном режиме;

техническая профилактика (чистка и тестирование);

проведение дефрагментации накопителей на жестких магнитных дисках;

полная проверка дисковой памяти на наличие вирусов;

очистка от пыли и грязи внутренних объемов ПЭВМ с разборкой, экранов видеомониторов, печатающих головок матричных и струйных принтеров.

Техническое обслуживание принтера:

проверка работоспособности на тестах;

чистка, смазка, настройка (при необходимости);

замена картриджа, тонера (при необходимости).

Техническое обслуживание сканера:

проверка работоспособности на тестах;

чистка, смазка, настройка (при необходимости).

Сетевое администрирование сервера:

инсталляция и настройка операционной системы;

настройка и проверка прав доступа;

резервное копирование;

установка антивирусных программ и антивирусное сканирование;

восстановление работоспособности при возникновении нештатных ситуаций.

Администрирование локальных вычислительных сетей:

настройка конфигурации активного оборудования и при необходимости ее изменение;

контроль работоспособности активного оборудования программными средствами;

контроль трафика сети;

устранение нештатных ситуаций при работе активного оборудования.

Техническое обслуживание структурированной кабельной системы:

контроль работоспособности сегментов кабельной системы;

устранение выявленных дефектов.

Техническое обслуживание копировального аппарата:

проверка работоспособности на тестах;

профилактика узла ксерографии;

чистка пылесосом тракта подачи транспортировки;

диагностика внутренними программными тестами;

Полугодовое обслуживание включает и ежемесячное обслуживание, а также следующие работы:

очистка от пыли внутренних объемов блоков питания;

очистка от пыли источников бесперебойного питания с последующим их тестированием;

очистка от пыли экранов видеомониторов;

регулировка и настройка, смазка вентиляторов.

Текущий ремонт и обслуживание включает в себя ежемесячное и полугодовое обслуживание, а также следующие работы:

проведение диагностики и локализация неисправных устройств;

заправка печатающих и копировальных устройств расходными материалами Заказчика.

все виды ремонтных работ.

Моделирование работы локальной вычислительной сети

Для моделирования работы локальной вычислительной сети была использована программа Cisco Packet Tracer 6.

```

Packet Tracer PC Command Line 1.0
PC>ping 192.168.0.2

Pinging 192.168.0.2 with 32 bytes of data:

Reply from 192.168.0.2: bytes=32 time=47ms TTL=128
Reply from 192.168.0.2: bytes=32 time=0ms TTL=128
Reply from 192.168.0.2: bytes=32 time=16ms TTL=128
Reply from 192.168.0.2: bytes=32 time=0ms TTL=128

Ping statistics for 192.168.0.2:
 Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
 Approximate round trip times in milli-seconds:
 Minimum = 0ms, Maximum = 47ms, Average = 15ms
PC>

```

Проверка работы локальной вычислительной сети изображена на рисунке 28.

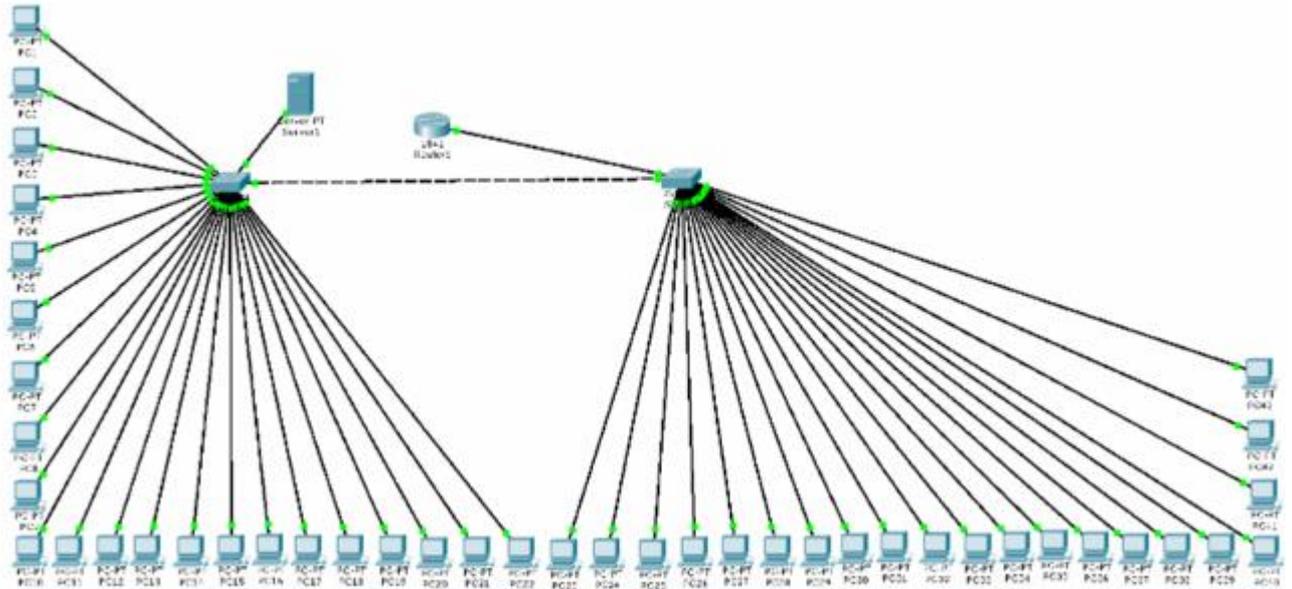


Схема модели локальной вычислительной сети изображена на рисунке 5.

Результаты запросов команды ping изображены в таблице 11.

Таблица 11 - Результаты запросов ping

| Время (сек.) | Начальное устройство | Конечное устройство | Тип  |
|--------------|----------------------|---------------------|------|
| 0,006        | PC1                  | DHCP1               | ICMP |
| 0,008        | PC1                  | Router1             | ICMP |
| 0,012        | PC1                  | PC2                 | ICMP |
| 0,005        | PC1                  | PC3                 | ICMP |
| 0,007        | PC1                  | PC4                 | ICMP |
| 0,006        | PC1                  | PC5                 | ICMP |
| 0,0015       | PC1                  | PC6                 | ICMP |
| 0,0011       | PC1                  | PC7                 | IC   |

|        |     |      |          |
|--------|-----|------|----------|
|        |     |      | MP       |
| 0,006  | PC1 | PC8  | IC<br>MP |
| 0,0012 | PC1 | PC9  | IC<br>MP |
| 0,006  | PC1 | PC10 | IC<br>MP |
| 0,007  | PC1 | PC11 | IC<br>MP |
| 0,006  | PC1 | PC12 | IC<br>MP |
| 0,005  | PC1 | PC13 | IC<br>MP |
| 0,006  | PC1 | PC14 | IC<br>MP |
| 0,009  | PC1 | PC15 | IC<br>MP |
| 0,008  | PC1 | PC16 | IC<br>MP |
| 0,005  | PC1 | PC17 | IC<br>MP |
| 0,009  | PC1 | PC18 | IC<br>MP |
| 0,006  | PC1 | PC19 | IC<br>MP |
| 0,005  | PC1 | PC20 | IC<br>MP |
| 0,007  | PC1 | PC21 | IC<br>MP |
| 0,008  | PC1 | PC22 | IC<br>MP |
| 0,0011 | PC1 | PC23 | IC<br>MP |
| 0,0012 | PC1 | PC24 | IC<br>MP |
| 0,008  | PC1 | PC25 | IC<br>MP |
| 0,009  | PC1 | PC26 | IC<br>MP |

|                       |     |      |          |
|-----------------------|-----|------|----------|
| 0,005                 | PC1 | PC27 | IC<br>MP |
| 0,0010                | PC1 | PC28 | IC<br>MP |
| 0,0010                | PC1 | PC29 | IC<br>MP |
| 0,009                 | PC1 | PC30 | IC<br>MP |
| 0,006                 | PC1 | PC31 | IC<br>MP |
| 0,005                 | PC1 | PC32 | IC<br>MP |
| 0,007                 | PC1 | PC33 | IC<br>MP |
| 0,0010                | PC1 | PC34 | IC<br>MP |
| 0,007                 | PC1 | PC35 | IC<br>MP |
| 0,006                 | PC1 | PC36 | IC<br>MP |
| 0,005                 | PC1 | PC37 | IC<br>MP |
| 0,008                 | PC1 | PC38 | IC<br>MP |
| Продолжение таблицы 1 |     |      |          |
| 0,0012                | PC1 | PC39 | IC<br>MP |
| 0,009                 | PC1 | PC40 | IC<br>MP |
| 0,006                 | PC1 | PC41 | IC<br>MP |
| 0,0011                | PC1 | PC42 | IC<br>MP |
| 0,0010                | PC1 | PC43 | IC<br>MP |

**по теме:** Обслуживание системы видеонаблюдения  
**Цель:** научиться обслуживать системы видеонаблюдения

**По завершению практического занятия студент должен уметь:** обслуживать системы видеонаблюдения

Продолжительность: 3 аудиторных часа (135 минут)

#### **Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

#### **Задание**



Установка оборудования для видеонаблюдения включает в себя такие этапы:

- обследование объекта,
- составление технического задания (основывается на пожеланиях клиента),
- разработка предварительного проекта,
- проведение согласований с заказчиком и утверждение проекта, в случае отсутствия замечаний,
- утверждение проектно-сметной документации,
- подготовка и подписание договора и утверждение,
- работы по монтажу системы, настройка и проверка системы, сдача объекта заказчику.

В стандартный регламент по техническому обслуживанию входят такие услуги, как:

#### 1 периодический осмотр

- внешний осмотр системы (кабельные линии, крепежи камер, разъемы, сервер, сами камеры),
- визуальный контроль качества транслируемого с камер наблюдения видео,
- проверка правильности работы ПО (программного обеспечения),
- проверка работоспособности системы в общем.

Частота такой проверки зависит от самой системы и от пожеланий клиента. Если система крупная, то есть включает большое количество камер и сопутствующего оборудования, то визит мастера может выполняться хоть ежемесячно. При этом время профилактического осмотра подбирается таким образом, чтобы было удобно клиенту.

#### 2 проведение консультаций

Планный осмотр это, конечно, хорошо, но частенько у клиентов возникают различные вопросы. Техническое обслуживание также включает в себя и консультационную поддержку. Клиент с любым вопросом может обратиться в поддержку и должен получить развернутый ответ.

#### 3 устранение неполадок

Техническое обслуживание в обязательном порядке должно включать услугу по устранению неполадок и решению различного рода проблем, связанных с правильным функционированием системы видеонаблюдения. Среди таких проблем может быть пропадание или же ухудшение качества изображения, транслируемого камерой/камерами, нарушение режима записи или сохранения информации, нарушение в работе программного обеспечения или же механическое повреждение оборудования. При этом, в регламенте должны быть четко прописаны сроки устранения неполадок. Определенных сроков нет, у каждой компании они могут быть различными.

В зависимости от компании и типа обслуживания (эконом, VIP), регламент может быть дополнен какими-либо дополнительными услугами.

Вот пример расценок одной из компаний, осуществляющей подобного рода услуги:

**Эконом - 1 профилактический выезд за 3 месяца, но не более 2 выездов в месяц (в том числе аварийных).**

**Стандарт - 2 профилактических выезда за 3 месяца, но не более 4 выездов в месяц (в том числе аварийных).**

**Люкс - 3 профилактических выезда за 3 месяца, не более 5 выездов в месяц (в том числе аварийных).**

| Количество каналов | Эконом | Стандарт | Люкс   |
|--------------------|--------|----------|--------|
| менее 4-х каналов  | 2 500  | 4 000    | 5 500  |
| 4-8 каналов        | 3 500  | 5 000    | 7 000  |
| 8-16 каналов       | 5 000  | 7 000    | 9 000  |
| 16-24 каналов      | 7 500  | 10 000   | 12 500 |
| более 24 каналов   | 9 500  | 12 000   | 15 000 |

ТО систем видео-наблюдения (техническое обслуживание), может быть ежемесячным (ТО-1) может быть ежеквартальным (раз в 3-и месяца, ТО-2).

Регламент работ разный.

При ТО-1 это:

Внешний осмотр технических устройств.

Чистка корпусов (от различных загрязнений), проверка на наличие трещин.



Проверка технического состояния резервного блока питания.

Проверка исправности органов управления .

Проверка правильности установки видеокамер и т.п.

ТО-2, плюс к прочему (см. выше, все перечисленные работы), проверка работоспособности резервного блока питания, переключают систему на резервное питание и обратно.

Контроль правильности программирования, режимов работы системы.

Вот это общее, могут быть и индивидуальные моменты связанные с конкретной системой видео-наблюдения.

На регулярное обслуживание системы, лучше заключить договор на ТО, причём лучше с той же компанией которая устанавливала систему видео-наблюдения.

**Текущий контроль**  
**бсеместр**  
**Практическое занятие**  
*форма текущего контроля*

**по теме:** Определение показателей безотказности системы

**Цель:** определять показатели безотказности системы

**По завершению практического занятия студент должен уметь:** определять показатели безотказности системы

Продолжительность: 3 аудиторных часа (135 минут)

**Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

**Задание**

Вероятность безотказной работы- вероятность того, что в пределах заданной наработки  $t$  отказ не возникает

где  $N_p$  - число работоспособных объектов на момент  $t$ ,

$N$  - общее число наблюдаемых объектов,

$n(t)$  - число объектов, отказавших на момент  $t$  от начала испытаний или эксплуатации.

Вероятность безотказной работы уменьшается с увеличением времени работы или наработки объекта. Зависимость вероятности безотказной работы от времени характеризуется кривой убыли ресурса объекта, пример которой приведен на рис. 9.

В начальный момент времени для работоспособного объекта вероятность его безотказной работы равна единице (100%). По мере работы объекта эта вероятность снижается и стремится к нулю.

Вероятность отказа характеризуется вероятностью возникновения отказа на момент времени  $t$

$$Q(t) = 1 - P(t) = \frac{n(t)}{N}$$

, где  $n(t)$  - число объектов, отказавших на момент  $t$  от начала испытаний или эксплуатации,  $N$  - общее число наблюдаемых объектов.



Вероятность возникновения отказа объекта возрастает с увеличением срока эксплуатации или наработки.

Пример зависимости вероятности возникновения отказа от времени показан на рис. 10. Для работоспособного объекта в начальный момент времени вероятность отказа близка к нулю. Для того, чтобы отказ проявился объекту необходимо начать работать, при этом вероятность отказа увеличивается с увеличением времени и стремится к единице. Вероятность отказа может быть также охарактеризована плотностью вероятности отказа

$$f(t) = \frac{dQ}{dt} \quad \text{или} \quad f(t) = \frac{\Delta n(t)}{N \cdot \Delta t}$$

, где  $\Delta n(t)$  - число отказов за промежуток времени  $\Delta t$ ,

$N$  - общее число наблюдаемых объектов.

Пример 1. После 500 часов наработки из 56 агрегатов, поставленных на эксплуатацию, в работоспособном состоянии оказалось 43 агрегата. Определить вероятность безотказной работы агрегата в течение 500 час.

Решение:

Используем формулу для определения вероятности безотказной работы объекта

$$P(500) = \frac{43}{56} = 0,768$$

Вероятность безотказной работы агрегата в течение 500 час составляет 76,8 %.

Пример 2. Для предыдущего примера определить вероятность отказа агрегат за 500 час работы.

Решение:

Используем формулу для вероятности отказа

$$Q(500) = 1 - P(500) = 1 - 0,786 = 0,232$$

или

$$Q(500) = \frac{56 - 43}{56} = 0,232$$

Таким образом, вероятность отказа агрегата за 500 час составляет 23,2%.

При определении вероятности безотказной работы и вероятности отказов широко используются две основных теоремы для определения вероятности случайного события:

1. Вероятность появления одного из двух несовместных событий равна сумме вероятности этих событий

$$P(A + B) = P(A) + P(B)$$

где  $A, B$  - несовместные события.

2. Вероятность совместного появления нескольких независимых событий равна произведению вероятностей этих событий

$$P(A_1 A_2 \dots A_n) = P(A_1) \cdot P(A_2) \cdot \dots \cdot P(A_n)$$

Первая теорема используется для нахождения вероятности отказа при возможности у объекта нескольких видов несовместных отказов. С использованием второй теоремы определяют вероятность безотказной работы объекта, состоящего из многих элементов, вероятность безотказной работы которых известна.

Пример 3. Система состоит их 4-х агрегатов. Надежность каждого агрегата в течение времени характеризуется вероятностью безотказной работы 90 %. Найти вероятность безотказной работы системы в течение времени при условии независимости отказов агрегатов.

Решение:

Используем теорему вероятности совместного появления работоспособного состояния всех агрегатов

$$P_c(t) = \prod_{i=1}^n P_i(t) = \prod_{i=1}^4 0,9 = 0,656$$

Следовательно, вероятность безотказной работы системы в течение времени равна 65,6 %.

Пример 4. В составе агрегата имеются 5 узлов. Вероятность отказа каждого узла в течение времени составляет 5 %. Отказы узлов несовместны. Определить вероятность отказа агрегата.

Решение:

Используем теорему для вероятности хотя бы одного из нескольких несовместных событий

$$Q(t) = \sum_{i=1}^n Q_i(t) = \sum_{i=1}^5 0,05 = 0,25$$

Таким образом, вероятность отказа агрегата в течение времени  $t$  составляет 25 %.

1.2 Интенсивность отказов - характеризует скорость возникновения отказов объекта в различные моменты времени его работы

$$\lambda(t) = \frac{\Delta n(t)}{N_p \cdot \Delta t}$$

где  $\Delta n(t)$  - число отказов за промежуток времени  $\Delta t$ ,

$N_p$  - число работоспособных объектов на момент  $t$ .

Интенсивность отказов может быть найдена теоретически

$$\lambda(t) = \frac{f(t)}{P(t)}$$

где  $f(t)$  - функция плотности вероятности наработки до отказа,

$P(t)$  - вероятность безотказной работы,

$$f(t) = \frac{\Delta n(t)}{N \cdot \Delta t}$$

Плотность распределения  $f(t)$  наработки до отказа может быть также определена по вероятности отказа

$$f(t) = Q'(t) = \frac{dQ(t)}{dt} \quad \text{или} \quad Q(t) = \int_0^t f(t) dt$$

Вероятность безотказной работы связана с интенсивностью отказов одним из основных уравнений теории надежности:

$$P(t) = \exp\left(-\int_0^t \lambda(t) dt\right)$$

В описанных способах оценки безотказности до первого отказа отказы не различаются по тяжести их последствий. В большинстве случаев при разработке объекта необходимо установить критерий отказа изделия по экономическим соображениям, истощению ресурса или другим характеристикам.

Критерием отказа называют признак или совокупность признаков неработоспособного состояния объекта, установленных в нормативно-технической или конструкторской документации.

1.3 Средняя наработка на отказ - это отношение наработки восстанавливаемого объекта к математическому ожиданию числа его отказов в течение этой наработки

$$T_0 = \frac{\sum_{i=1}^N t_i}{\sum_{i=1}^N m_i}$$

где  $N$  - общее число объектов, поставленных на испытания или в эксплуатацию,  $t_i$  - наработка  $i$ -того объекта,  $m_i$  - число отказов  $i$ -того объекта за весь наблюдаемый период.

Средняя наработка на отказ используется для характеристики восстанавливаемых объектов.

1.4 Средняя наработка до отказа - математическое ожидание наработки объекта до первого отказа

$$T_{cp} = \int_0^{\infty} P(t) dt \quad \text{или} \quad T_{cp} = \sum_{i=1}^k \frac{N_{pi}}{N} \Delta t_i$$

где  $N_{pi}$  - число работоспособных объектов на интервале наработки  $t_i - t_{i+1}$ ;  $N$  - общее число наблюдаемых объектов,  $\Delta t = t_{i+1} - t_i$  - интервал времени;

$k$  - число рассматриваемых интервалов наработки.

Среднюю наработку до отказа можно также определить иначе

$$T_{cp} = \frac{1}{n} \sum_{i=1}^n t_i$$

где  $t_i$  - наработка до отказа  $i$ -того объекта,  $n$  - число объектов.

Показатель используется для характеристики надежности восстанавливаемых объектов.

1.5 Средняя наработка между отказами - математическое ожидание наработки объекта от окончания восстановления его работоспособного состояния после отказа до возникновения следующего отказа.

Вычисляется как отношение суммарной наработки объекта между отказами за рассматриваемый период к числу отказов за тот же период

$$T = \frac{1}{m} \sum_{i=1}^m t_i$$

Показатели безотказности определяют на разных стадиях работы объекта с целью его совершенствования и с целью контроля нормируемых значений при эксплуатации.

## 2. Показатели долговечности

2.1 Средний ресурс - математическое ожидание ресурса

$$T_p = \frac{\sum_{i=1}^N T_{pi}}{N}$$

где  $T_{pi}$  - ресурс  $i$ -того объекта,

$N$  - число объектов.

2.2 Гамма - процентный ресурс представляет собой наработку, в течении которой объект не достигает предельного состояния с заданной вероятностью, выраженный в процентах (рис. 11).

Для расчета показателя используется формула вероятности

$$P(T_{py}) = \int_{T_{py}}^{\infty} p(T_p) dT_p = \frac{\gamma}{100}$$

где  $T_{py}$  - наработка до предельного состояния (ресурс).

Гамма - процентный ресурс является основным расчетным показателем для подшипников и других элементов. Существенное достоинство этого показателя - возможность его определения до завершения испытания всех образцов. В большинстве случаев используют 90 % ресурс.

2.3 Назначенный ресурс- суммарная наработка  $T_{рн}$ , при достижении которой применение объекта по назначению должно быть прекращено независимо от его технического состояния.

2.4 Установленный ресурс- технически обоснованная или заданная величина ресурса  $T_{ру}$ , обеспечиваемая конструкцией, технологией и эксплуатацией, в пределах которой объект не должен достигать предельного состояния.

2.5 Средний срок службы- математическое ожидание срока службы.

$$T_{сл} = \frac{\sum_{i=1}^N T_{сли}}{N}$$

где  $T_{сли}$ - срок службы  $i$ -того объекта.

2.6 Гамма - процентный срок службы- календарная продолжительность эксплуатации в течение которой объект не достигает предельного состояния с вероятностью  $\gamma$ , выраженной в процентах

$$P(T_{слу}) = \int_{T_{слу}}^{\infty} p(T_{сл}) dT_{сл} = \frac{\gamma}{100}$$

2.7 Назначенный срок службы- суммарная календарная продолжительность эксплуатации  $T_{сл.н}$ , при достижении которой применение объекта по назначению должно быть прекращено, независимо от его технического состояния.

2.8 Установленный срок службы- технико-экономически обоснованный или заданный срок службы  $T_{сл.у}$ , обеспечиваемый конструкцией, технологией и эксплуатацией, в пределах которого объект не должен достигать предельного состояния.

**Текущий контроль**  
**бсеместр**  
**Практическое занятие**  
*форма текущего контроля*

**по теме:** Определение показателей долговечности системы

**Цель:** определять показатели долговечности системы

**По завершению практического занятия студент должен уметь:** определять показатели долговечности системы

Продолжительность: 2 аудиторных часа (90 минут)

**Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

**Задание**

**Показатели долговечности.**

1)  $t_c$ -срок службы – это календарная продолжительность от начала эксплуатации объекта до перехода его в предельное состояние (величина случайная)

2)  $\bar{T}_c = M[t_c]$  -средний срок службы,

3)  $T_\gamma = P\{t_c > t_\gamma\} = \frac{\gamma}{100} \%$  -гамма-процентный срок службы, где  $t_\gamma$  - календарная продолжительность от начала

эксплуатации объекта, в течении которой объект не достигнет предельного состояния с вероятностью  $\gamma$ .

4) Ресурс-наработка от начала эксплуатации до перехода в предельное состояние.

5) Назначенный срок хранения - календарная продолжительность хранения при достижении которой эксплуатация объекта должна быть прекращена независимо от его технического состояния.

**Текущий контроль**  
**бсеместр**  
**Практическое занятие**  
*форма текущего контроля*

**по теме:** Определение единичных показателей достоверности информации в системе

**Цель:** определять единичные показатели достоверности информации в системе

**По завершению практического занятия студент должен уметь:** определять единичные показатели достоверности информации в системе

Продолжительность: 3 аудиторных часа (135 минут)

**Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

## Задание

В силу специфики информационных систем, которые априори предназначены для преобразования информации, важнейшим их свойством является достоверность функционирования.

**Достоверность функционирования** - это свойство системы, обуславливающее безошибочность производимых ею преобразований информации.

Достоверность функционирования ИС полностью определяется и измеряется достоверностью ее результирующей информации. Для ИС достоверность функционирования является не просто одним из свойств их надежности, но приобретает и самостоятельное значение, поскольку именно достоверность конечной информации обуславливает требования к надежности системы.

Как уже указывалось, надежность ИС - не самоцель, а лишь средство обеспечения оптимальной достоверности ее выходной информации, обуславливающей наивысшую эффективность функционирования системы.

Достоверность информации - это свойство информации отражать реально существующие объекты с необходимой точностью. Достоверность (D) информации измеряется доверительной вероятностью необходимой точности, то есть вероятностью того, что отражаемое информацией значение параметра отличается от истинного значения этого параметра в пределах необходимой точности:

$$D = P\{\}$$

где - реальная точность отображения параметра,  $\Delta$  - диапазон необходимой точности отображения параметра.

Для более полного понимания вышеприведенного определения следует пояснить некоторые присутствующие в нем понятия.

Истинная информация - информация, объективно, точно и правильно отражающая характеристики и признаки какого-либо объекта или явления (адекватная заданному параметру объекта).

Точность информации - это характеристика, показывающая степень близости отображаемого значения параметра и истинного его значения. Необходимая точность определяется функциональным назначением информации и должна обеспечивать правильность принятия управленческих решений.

Таким образом, при оценке истинности информации существуют две основные вероятностные задачи:

? определение точности информации или расчет математического ожидания абсолютной величины отклонения значения показателя от объективно существующего истинного значения отображаемого им параметра;

? определение достоверности информации или вычисление вероятности того, что погрешность показателя не выйдет за пределы допустимых значений.

Адекватность отражения включает в себя понятия и точности, и достоверности, которые не должны смешиваться (что иногда имеет место в определениях достоверности информации, приводимых в ряде книг).

Из сказанного следует, что нарушение надежности ИС, приводящее к ухудшению точности результирующей информации в пределах необходимой точности, не снижает эффективности функционирования системы (коэффициента сохранения эффективности). И если отсутствие информации в положенное время (ее несвоевременность) трактовать в обобщенном виде как наличие недостоверной информации, то единственным показателем качества информации, зависящим от надежности ИС и влияющим на эффективность ее функционирования, является достоверность.

Показатели достоверности информации

Достоверность информации может рассматриваться с разных точек зрения. Поэтому для достоверности правомерно и целесообразно использовать систему показателей.

Единичные показатели достоверности информации

1. Доверительная вероятность необходимой точности (достоверность) -  $D = 1 - P_{гр}$  - вероятность того, что в пределах заданной наработки (информационной совокупности - массива, показателя, реквизита, кодового слова, символа или иного информационного компонента) отсутствуют грубые погрешности, приводящие к нарушению необходимой точности.

2. Средняя наработка информации на ошибку -  $Q = 1/P$ . Отношение объема информации, преобразуемой в системе, к математическому ожиданию количества ошибок, возникающих в информации.

3. Вероятность ошибки (параметр потока ошибок) -  $P_{гр}$  - вероятность появления ошибки в очередной информационной совокупности.

Показатели корректируемости информационных систем

1. Вероятность коррекции в заданное время -  $R_{корр}(t)$  - вероятность того, что время, затрачиваемое на идентификацию и исправление ошибки, не превысит заданного  $t$ .

2. Среднее время коррекции информации -  $T_{и}$  - математическое ожидание времени, затрачиваемого на идентификацию и исправление ошибки.

Комплексные показатели достоверности

1. Коэффициент информационной готовности -

$$K_{из} = \frac{T_{раб} - (T_{\varepsilon} + T_{и})}{T_{раб}}$$

это вероятность того, что информационная система окажется способной к преобразованию информации в произвольный момент времени того периода ( $T^{\wedge}$ ), который планировался для этого преобразования, то есть выполнения условия, что в данный момент времени система не будет находиться в состоянии внепланового обслуживания, вызванного устранением отказа или идентификацией и исправлением ошибки.

2. Коэффициент информационного технического использования -

$$K_{ти} = \frac{T_{раб} - (T_{\varepsilon} + T_{к} + T_{и})}{T_{раб} + T_{нф}}$$

это отношение математического ожидания планируемого времени работы системы на преобразование информации, за вычетом времени восстановления Тв контроля - Тк, идентификации и исправления ошибок - Ти, к сумме планируемого времени работы системы и профилактического обслуживания Тпф.

Наряду с понятием достоверности информации существует понятие достоверности данных, рассматриваемое в синтаксическом аспекте. Под достоверностью данных понимается их безошибочность. Она измеряется вероятностью отсутствия ошибок в данных (в отличие от достоверности информации, к снижению достоверности данных приводят любые погрешности, а не только грубые). Недостоверность данных может не повлиять на объем данных, но может и уменьшить и увеличить его, в отличие от недостоверности информации, всегда уменьшающей ее количество. Наконец, недостоверность данных может не нарушить достоверность информации (например, при наличии в последней необходимой избыточности).

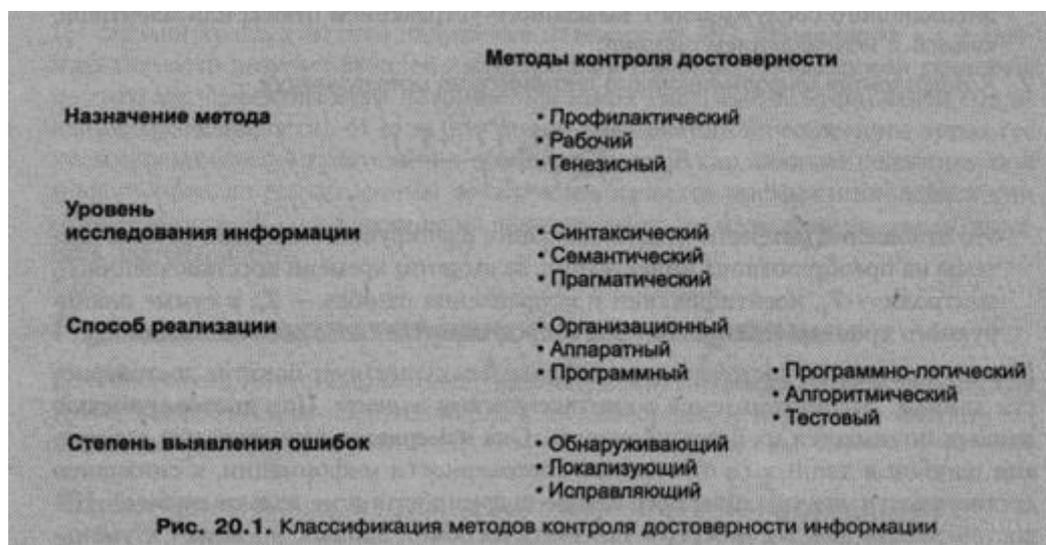
Обеспечение достоверности информации

Одним из наиболее действенных средств обеспечения достоверности информации в ИС является ее контроль. Контроль - процесс получения и обработки информации с целью оценки соответствия фактического состояния объекта предъявляемым к нему требованиям и выработки соответствующего управляющего решения. Объектом контроля в нашем случае является достоверность информации, следовательно, при контроле должно быть выявлено соответствие фактической и необходимой точности представления информации или, с учетом рассмотренной ранее нормы этого соответствия, выявлено наличие или отсутствие ошибок в контролируемой информации. При обнаружении ошибки должны быть приняты меры для ее устранения или, по крайней мере, выработаны соответствующие рекомендации по локализации и идентификации обнаруженной

ошибки и уменьшению последствий ее влияния на функционирование ИС; исправление ошибок в последнем случае выполняется путем выполнения некоторых внешних относительно процедуры контроля операций.

Классификация методов контроля достоверности

**Методы контроля** достоверности информации, применяемые в ИС, весьма разнообразны. Классификация методов контроля может быть выполнена по большому числу признаков, в частности: по назначению, по уровню исследования информации, по способу реализации, по степени выявления и коррекции ошибок.



Классификация методов контроля достоверности по назначению

По назначению следует различать профилактический, рабочий и генезисный контроль.

Профилактический контроль и, например, одна из наиболее распространенных его форм - тестовый контроль, предназначен для выявления состояния системы в целом и отдельных ее звеньев до включения системы в рабочий режим. Целью профилактического контроля, осуществляемого часто в утяжеленном режиме работы системы, является выявление и прогнозирование неисправностей в ее работе с последующим их устранением.

Рабочий контроль, или контроль в рабочем режиме, производится в процессе выполнения системой возложенных на нее функций. Он, в свою очередь, может быть разделен на функциональный контроль и контроль качества продукции. Функциональный контроль может преследовать цель либо только проверки работоспособности (отсутствия неисправностей) системы, либо, кроме того, установления места и причины неисправности (диагностический контроль). Контроль качества продукции в нашем случае как раз и является контролем достоверности информации как одним из важнейших показателей качества продукции выпускаемой ИС.

Генезисный контроль проводится для выяснения технического состояния системы в прошлые моменты времени с целью определения причин сбоев и отказов системы, имевших место ранее, сбора статистических данных об ошибках, их характере, величине и последствиях (экономических потерях) этих ошибок для ИС.

Классификация методов контроля достоверности по уровню исследования информации

По уровню исследования информации контроль может быть синтаксический, семантический и прагматический.

Синтаксический контроль - это, по существу, контроль достоверности данных, не затрагивающий содержательного, смыслового аспекта информации. Предметом синтаксического контроля являются отдельные символы, реквизиты, показатели: допустимость их наличия, допустимость их кодовой структуры, взаимных сочетаний и порядка следования.

Семантический контроль оценивает смысловое содержание информации, ее логичность, непротиворечивость, согласованность, диапазон возможных значений параметров, отражаемых информацией, динамику их изменения.

Прагматический контроль определяет потребительную стоимость (полезность, ценность) информации для управления, своевременность и актуальность информации, ее полноту и доступность.

Классификация методов контроля достоверности по способу реализации

По способу реализации контроль может быть организационным, программным, аппаратным и комбинированным.

Организационный контроль достоверности является одним из основных в ИС. Он представляет собой комплекс мероприятий, предназначенных для выявления ошибок на всех этапах участия эргатического звена в работе системы, причем обязательным элементом этих мероприятий является человек или коллектив людей.

Программный контроль основан на использовании специальных программ и логических методов проверки достоверности информации или правильности работы отдельных компонентов системы и всей системы в целом. Программный контроль, в свою очередь, подразделяется на программно-логический, алгоритмический и тестовый.

Программно-логический контроль базируется на использовании синтаксической или семантической избыточности; алгоритмический контроль использует как основу вспомогательный усеченный алгоритм преобразования информации, логически связанный с основным рабочим алгоритмом (тестовый контроль был рассмотрен чуть выше).

Аппаратный контроль реализуется посредством специально встроенных в систему дополнительных технических схем. Этот вид контроля также подразделяется

на непрерывный и оперативный (аппаратно-логический) контроль достоверности, а также непрерывный контроль работоспособности.

Непрерывный контроль достоверности функционирует непрерывно в процессе работы системы параллельно с процедурами основного технологического процесса преобразования информации. Во время оперативного (аппаратно-логического) контроля достоверности выполнение основных технологических операций над информацией приостанавливается. Непрерывный контроль работоспособности - это уже не контроль достоверности информации, а контроль значений параметров компонентов системы с помощью встроенных в них датчиков.

Классификация методов контроля достоверности по степени выявления и коррекции ошибок

По степени выявления и коррекции ошибок контроль делится на:

? обнаруживающий, фиксирующий только сам факт наличия или отсутствия ошибки;

? локализирующий, позволяющий определить как факт наличия, так и место ошибки (например символ, реквизит и т. д.);

? исправляющий, выполняющий функции обнаружения, и локализации, и исправления ошибки.

Основные показатели качества контроля достоверности

Функциональные показатели качества контроля (показатели его эффективности) должны количественно определять степень приспособленности и выполнения контролем поставленных перед ним задач. В общем случае контроля такими показателями могут служить коэффициенты, численно равные условным вероятностям соответствующих событий при условии наличия ошибки.

Для обнаруживающего и локализирующего контроля такими коэффициентами являются:

? коэффициент обнаружения ошибок -  $K_{обн} = N_{обн} / N_{ош} - R_{обн} / R_{ош}$ ;

? коэффициент необнаружения ошибок -  $K_{но} = N_{но} / N_{ош} = R_{но} / R_{ош}$ ;

? коэффициент локализации ошибок  $K_{лок}$  для большинства методов локализирующего контроля равен коэффициенту обнаружения, то есть  $K_{лок} = K_{обн}$

Методы контроля, исправляющие ошибки, характеризуются следующими коэффициентами:

? исправления ошибок  $K_{испр} = N_{испр} / N_{ош} = R_{испр} / R_{ош}$ ;

? искажения ошибок  $K_{иск} = N_{иск} / N_{ош} = R_{иск} / R_{ош}$ ;

? обнаружения ошибок  $K_{обн} = N_{об} / N_{ош} = R_{обн} / R_{ош}$ ;

? необнаружения ошибок  $K_{но} = N_{но} / N_{ош} = R_{но} / R_{ош}$ .

В этих соотношениях:

? N - число структурных элементов (символов, реквизитов, показателей и т. д.) в информационной совокупности;

?  $N_{но}$ ,  $N_{испр}$ ,  $N_{иск}$ ,  $N_{обн}$ , - число ошибок, которые в процессе контроля, соответственно, не обнаруживаются, правильно исправляются, неверно исправляются (искажаются), только обнаруживаются (факт наличия которых просто устанавливается, а сами они не исправляются);

?  $R_{ош}$ ,  $R_{обн}$ ,  $R_{но}$ ,  $R_{испр}$ ,  $R_{иск}$  ~ вероятности наличия ошибки, обнаружения, необнаружения, исправления и искажения ошибки, соответственно.

Важными показателями качества контроля являются также:

? коэффициент выявления ошибок  $K_{выявл} = N_{выявл} / N_{ош}$ , характеризующий суммарное относительное число выявляемых ( $N_{выявл}$ ) ошибок в контролируемой информационной совокупности;

? коэффициент трансформации ошибок  $K_{тр} = N_{ош.вых} / N_{ош}$ , характеризующий суммарное относительное число необнаруженных и вновь внесенных при контроле ( $N_{ош.вых}$ ) ошибок.

Для контроля с исправлением ошибок:

$K_{выявл} = K_{испр} + K_{иск} + K_{обн}$

$K_{тр} = K_{но} + K_{иск}$

Для контроля с обнаружением ошибок:

$K_{испр} = K_{иск} = 0$

$K_{выявл} = K_{обн}$

$K_{тр} = K_{но}$

В качестве дополнительных функциональных показателей могут быть использованы значения вероятности правильного обнаружения ошибки и ложного обнаружения ошибки, учитывающие надежность работы системы контроля:

?  $R_{пр}$  - вероятность правильного обнаружения ошибки, то есть такого события, когда не вырабатывается информация о наличии ошибки при условии действительного ее отсутствия;

?  $R_{лт}$  - вероятность ложного обнаружения ошибки (ложной тревоги), то есть такого события, когда вырабатывается информация о наличии ошибки при реальном ее отсутствии.

Соответствующие коэффициенты

$K_{пр} = R_{пр} / R_{ош}$ ,  $K_{лт} = R_{лт} / R_{ош}$

могут быть существенно больше 1, поскольку

$K_{пр} + K_{лт} = (1 - R_{ош}) / R_{ош}$

Технико-эксплуатационные показатели контроля:

? алгоритмическая сложность контроля;

? вид и величина используемой избыточности;

? надежность контроля;

? универсальность (возможность использования на различных фазах технологического процесса, при решении различных задач и для различных групп и типов информационных ошибок) и др.

Экономические показатели эффективности контроля - это затраты на контроль:

? единовременные;

? текущие;

? материальные;

? трудовые;

? временные.

**Текущий контроль**  
**6 семестр**  
**Практическое занятие**  
*форма текущего контроля*

**по теме:** Формирование предложений по реинжинирингу информационной системы (указать предметную область)  
**Цель:** формировать предложения по реинжинирингу информационной системы (указать предметную область)

**По завершению практического занятия студент должен уметь:** разрабатывать техническое задание на сопровождение информационной системы (указать предметную область)

Продолжительность: 3 аудиторных часа (135 минут)

**Необходимые принадлежности**

Персональный компьютер, программное обеспечение.

**Задание**

Реинжиниринг определяется как «перепроектирование». Объектом является ТП.

Реинжиниринг- фундаментальное переосмысление и реальное перепроектирование ТП для достижения существенных улучшений показателей затрат, качества, оперативности и надежности.

Главные задачи:

1. Уменьшение себестоимости;
2. Повышение производительности.

Базовые принципы:

1. Горизонтальное сжатие процессов (несколько процедур объединяется в 1);
2. Вертикальное сжатие процесса. Многие процессы требуют принятия квалифицированного решения с последующим исполнением техниками-рабочими. Смысл вертикального сжатия: принятие решения переносится на уровень исполнителя.
3. Обеспечение естественного порядка выполнения живого процесса;
4. Наличие различных вариантов реализации процесса;
5. Уменьшение количества проверок и управляющих воздействий;
6. Устранение излишней интеграции (концентрация процесса в рамках одного структурного подразделения);
7. Минимизация количества согласований по принятию решений;
8. Создание единой точки контактов участников ТП в случае их территориальной распределенности.

Этапы реинжиниринга.

1. Спецификация основных целей. Разработка образа будущего предприятия. (Что бы хотелось иметь?)
2. Создание модели существующего предприятия. (Модель «Как есть»)
3. Перепроектирование ТП. Создание более эффективных рабочих процедур. Определение способов использования ИТ.

Разработка ИС, поддерживающей организацию и ее ТП.

4. Внедрение, интеграция, тестирование, обучение сотрудников.
5. Переход на новую технологию.

Особенности перестроенных предприятий:

- процессами являются все виды работ;
- сокращение управленцев среднего звена;
- группировка работников в соответствии с их областью компетенции.

Переход от «как есть» к «как должно быть» может быть выполнен 2-мя способами:

1. Легкий реинжиниринг- совершенствование процессов на основе оценки их эффективности. Критерии: стоимость и временные затраты, их соотношение с ожидаемой выгодой.
2. Жесткий реинжиниринг- переосмысление целей и задач, радикальное изменение ТП. Имеет место когда есть соответствующее решение вышестоящей организации или в случае банкротства.

Инструментальные средства реинжиниринга:

С информационной точки зрения- это case – система, которая позволяет создание модели ТП предприятия. (Например, AnyLogic)

**Текущий контроль**  
**6 семестр**  
**МДК. 6.04 Интеллектуальные системы и технологии**  
**Практическое занятие**  
*форма текущего контроля*

**по теме:** «Моделирование интеллектуальных систем»

**Цель:** знать виды и классификацию локальных сетей, физические среды передачи данных, научиться устанавливать и настраивать сетевой интерфейс.

**По завершению практического занятия студент должен уметь:** устанавливать и настраивать сетевой интерфейс.

Продолжительность: 4 аудиторных часа (180 минут)

**Необходимые принадлежности**

Компьютеры, программное обеспечение: MS Windows'.

**Основные сведения:**

I. Построение модели в среде AnyLogic.

Пакет AnyLogic – отечественный профессиональный инструмент нового поколения, который предназначен для разработки и исследования имитационных моделей. Разработчик продукта – компания «Экс Джей Текнолоджис» (XJ Technologies), г. Санкт-Петербург; электронный адрес: [www.xjtek.ru](http://www.xjtek.ru).

Графическая среда моделирования поддерживает проектирование, разработку, документирование модели, выполнение компьютерных экспериментов, оптимизацию параметров относительно некоторого критерия.

При разработке модели можно использовать элементы визуальной графики: диаграммы состояний (стейтчарты), сигналы, события (таймеры), порты и т.д.; синхронное и асинхронное планирование событий; библиотеки активных объектов.

#### Пользовательский интерфейс

После запуска AnyLogic открывается рабочее окно, в котором для продолжения работы надо создать новый проект или открыть уже существующий.

Чтобы создать новый проект, щелкните по соответствующей кнопке на панели инструментов или выберите пункт меню **Файл | Создать проект** и затем из выпадающего меню – **Модель**. Откроется диалоговое окно **Новая модель**, где задается имя и местоположение нового проекта. Далее следуйте указаниям *Мастера создания модели*. Можно создавать новую модель «с нуля» или использовать шаблон.

При открытии проекта (нового или существующего) AnyLogic всегда открывает среду разработки проекта – графический редактор модели (рис. 3.1). Рассмотрим основные составляющие этого редактора.

**Окно проекта** обеспечивает легкую навигацию по элементам проекта, таким как пакеты, классы и т.д. Поскольку проект организован иерархически, то он отображается в виде дерева: сам проект образует верхний уровень дерева рабочего проекта, пакеты – следующий уровень, классы активных объектов и сообщений – следующий и т.д. Можно копировать, перемещать и удалять любые элементы дерева объектов, легко управляя рабочим проектом.

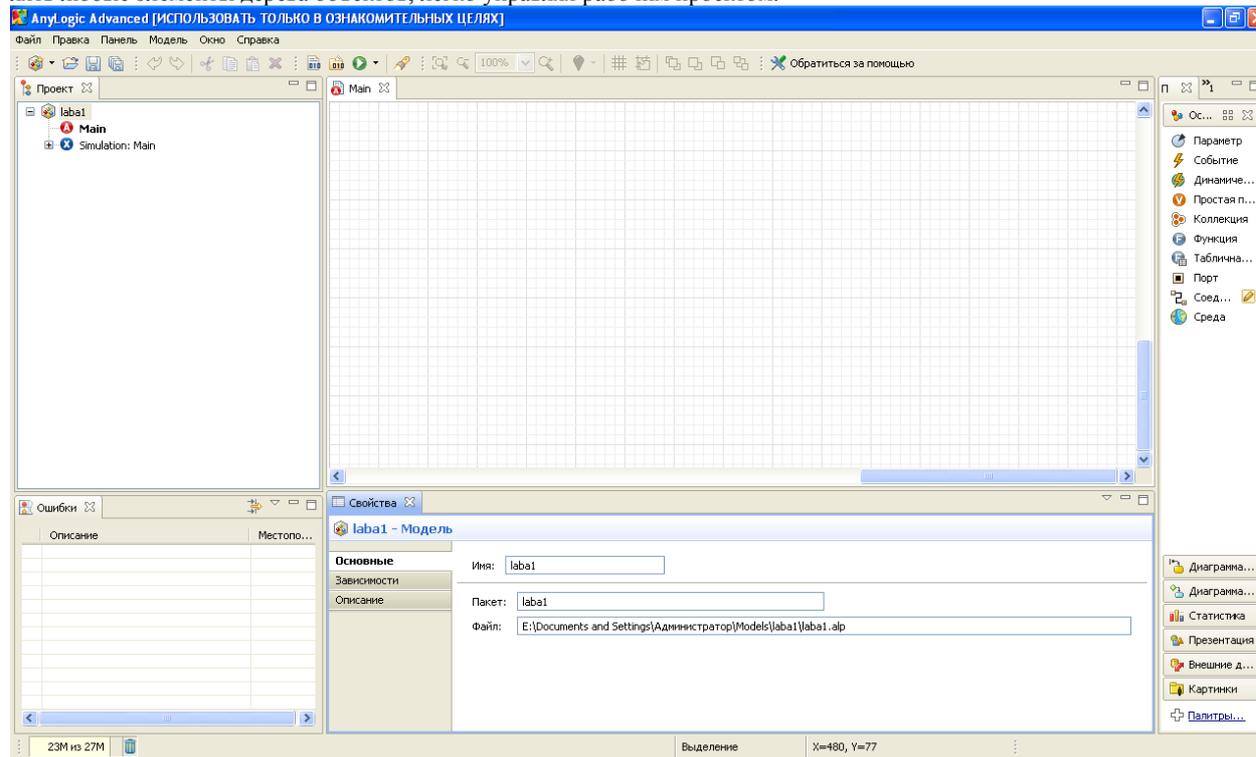


Рис. 3.1 Главное окно проекта.

Одна из ветвей в дереве проекта имеет название *Simulation:Main* (эксперимент). Эксперимент хранит набор настроек, с помощью которых конфигурируют работу модели. Один эксперимент создается автоматически при создании проекта. Это *простой эксперимент* с именем *Simulation*, позволяющий использовать модель с помощью анимации и поддерживающий инструменты для отладки модели. Простой эксперимент используется в большинстве случаев. Поддерживается ещё несколько типов экспериментов для различных задач моделирования.

**Структурная диаграмма.** При построении модели нужно задать ее структуру (т.е. описать, из каких частей состоит модель системы) и поведение отдельных объектов системы. В AnyLogic структурными элементами модели являются так называемые *активные объекты*. Активный объект имеет структуру и поведение. Элементы структуры – это другие активные объекты, включенные как составные элементы данного активного объекта, и связи, которые существуют между включенными активными объектами. Активные объекты могут содержать: события, стейтчарты, переменные, функции, уравнения, параметры. Структура активного объекта задается графически на структурной диаграмме. Поведение задается с помощью стейтчарта и определяет реакции активного объекта на внешние события – логику его действий во времени.

**Диаграмма состояний** (или стейтчарт – *statechart*) – это модифицированные графы переходов конечного автомата. Стейтчарт позволяет графически задать пространство состояний алгоритма поведения объекта, события, которые являются причинами срабатывания переходов из одних состояний в другие, и действия, происходящие при смене состояний. Стейтчарты в AnyLogic поддерживают следующие типы событий: *сигнал* – объект может послать сигнал другому объекту, чтобы уведомить его о чем-то; *таймаут* – в течение заданного промежутка времени в стейтчарте ничего не происходит; *событие* – событие, при котором значение булево выражения становится «истина».

Кроме того, в окне редактора для модели можно построить двумерное или трехмерное анимационное представление, которое помогает понять, что происходит с моделью во времени. Именно в этом окне визуально представляется имитация поведения моделируемой системы. Элементы анимационной картинке имеют свои параметры, которые могут быть связаны с переменными и параметрами модели. Изменение переменных модели во времени ведет к изменению графического образа, что позволяет пользователю наглядно представить динамику моделируемой системы с помощью динамически меняющейся графики.

**Окно свойств.** В редакторе AnyLogic для каждого выделенного элемента модели существует свое окно свойств, в котором указываются свойства (параметры) этого элемента. При выделении какого-либо элемента в окне редактора снизу появляется окно свойств, показывающее параметры данного выделенного элемента. Окно свойств содержит несколько вкладок. Каждая вкладка содержит элементы управления, такие как поля ввода, флажки, переключатели, кнопки и т.д., с помощью которых можно просматривать и изменять свойства элементов модели. Число вкладок и их внешний вид зависит от типа выбранного элемента.

**Окно палитры.** Содержит элементы (графические объекты), которые могут быть добавлены на структурную диаграмму. Элементы разбиты по группам, отображаемым на разных вкладках. Чтобы добавить объект палитры на диаграмму, щелкните сначала по элементу в палитре, а затем щелкните по диаграмме.

**Параметры.** Активный объект может иметь параметры. Параметры обычно используются для задания характеристик объекта. Вы можете задать различные значения параметров для разных объектов одного и того же класса, что требуется в тех случаях, когда объекты имеют одинаковое поведение, но их характеристики разные. Возможно описание параметров любых *Java*-классов.

Чтобы создать параметр класса активного объекта (рис. 3.2), в окне **Проект** щелкните мышью по классу активного объекта. В окне **Свойства** щелкните по кнопке **Новый параметр**. Задайте свойства параметра в открывшемся диалоговом окне **Параметр**.

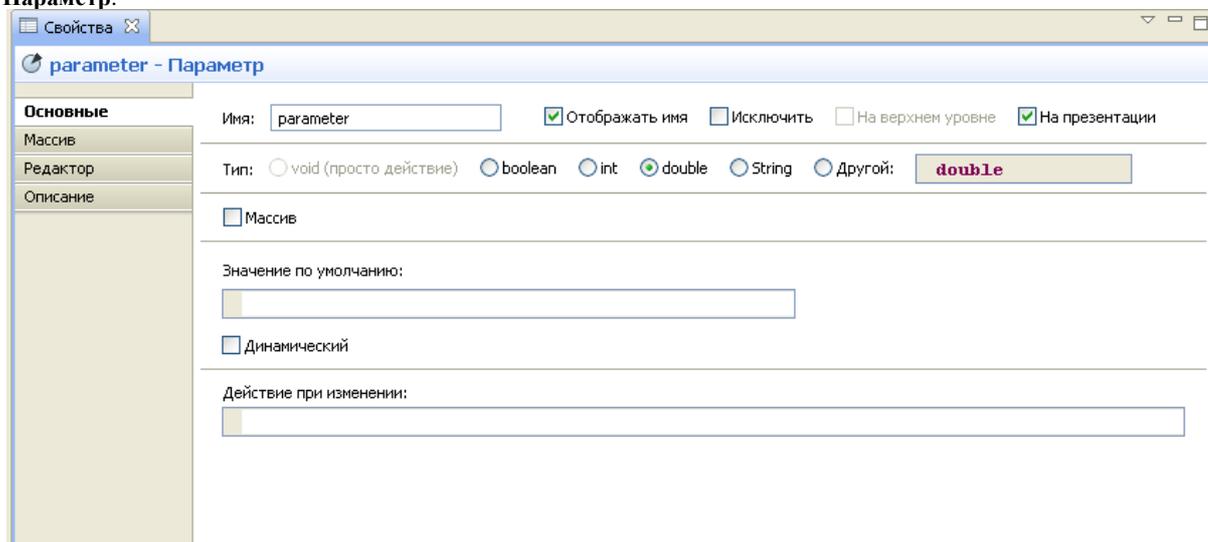


Рис. 3.2. Окно параметров.

**Переменные.** Активный объект может содержать переменные. Переменные могут быть либо внутренними, либо интерфейсными. Активный объект может иметь переменные, моделирующие, меняющиеся во времени величины. Переменные могут быть вынесены в интерфейс активного объекта и связаны с переменными других активных объектов. Тогда при изменении значения одной переменной будет немедленно меняться и значение связанной с ней зависимой переменной другого объекта. Этот механизм обеспечивает непрерывное и/или дискретное взаимодействие объектов.

**Передача сообщений.** AnyLogic позволяет передавать информацию от одного объекта другому путем передачи специальных пакетов данных – сообщений. С помощью передачи сообщений можно реализовать механизм оповещения – сообщения будут представлять команды или сигналы, посылаемые системой управления. *Можно также смоделировать поток заявок*, в этом случае сообщения будут представлять собой заявки – объекты, которые производятся, обрабатываются, обслуживаются или еще каким-нибудь образом подвергаются воздействию моделируемого процесса (документы в модели бизнес-процесса, клиенты в модели системы массового обслуживания, детали в производственных моделях).

Сообщения принимаются и посылаются через специальные элементы активных объектов – *порты*. Обмен сообщениями возможен только между портами, соединенными соединителями – элементами, играющими роль путей движения сообщений. Чтобы соединить порты вложенных объектов, выберите в палитре «**Основная**» инструмент **Соединитель**, перетащите его на диаграмму и подсоедините к портам. Также можно сделать двойной щелчок мышью по одному порту, а затем двойной щелчок мышью по второму порту.

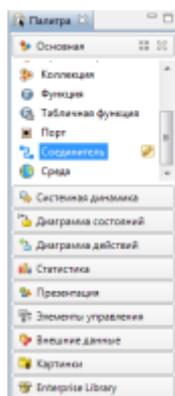


Рис.3.3. Выбор соединителя.

**Запуск и просмотр модели.** Запускать и отлаживать модель можно с помощью меню **Модель** и панели инструментов:



При исполнении модели запустится компилятор, который построит исполняемый код модели в языке *Java*, скомпилирует его и затем запустит модель на исполнение.

Для запуска модели щелкните по кнопке **Выполнить**, затем выберите эксперимент из выпадающего списка. После этого откроется окно презентации, отображающее созданную презентацию для запущенного эксперимента. Щелкните по кнопке, чтобы запустить модель и перейти на презентацию.

В окне презентации можно увидеть: анимированную диаграмму модели, окна инспекта элементов модели, ожившую анимацию, диаграммы состояний, графики статистики.

В ходе выполнения лабораторной работы необходимо научиться создавать дискретно-событийные модели с помощью библиотеки Enterprise Library пакета AnyLogic.

Для создания новой модели щелкните мышью по кнопке **Создать проект**. Появится диалоговое окно, в котором вы должны будете дать имя файлу вашей модели и выбрать каталог, где он будет храниться.

Рассмотрим рабочее окно AnyLogic. В левой части рабочей области находится панель «Проект». Панель «Проект» обеспечивает легкую навигацию по элементам моделей, открытых в текущий момент времени.

В правой рабочей области отображается панель «Палитра», а внизу – панель «Свойства». Панель «Палитра» содержит разделенные по категориям элементы, которые могут быть добавлены на диаграмму класса активного объекта или эксперимента. Панель «Свойства» используется для просмотра и изменения свойств выбранного в данный момент элемента (или элементов) модели.

В центре рабочей области AnyLogic открывается графический редактор диаграммы класса активного объекта *Main*.

Чтобы добавить объект на блок-схему модели, щелкните по объекту в окне палитры **Enterprise Library** и перетащите его мышью на структурную диаграмму. При этом его свойства будут отображены на панели «Свойства». В этом окне вы можете изменять свойства элемента в соответствии с требованиями вашей модели. Позднее для изменения свойств элемента нужно будет сначала щелчком мыши выделить его на диаграмме или в дереве проекта.

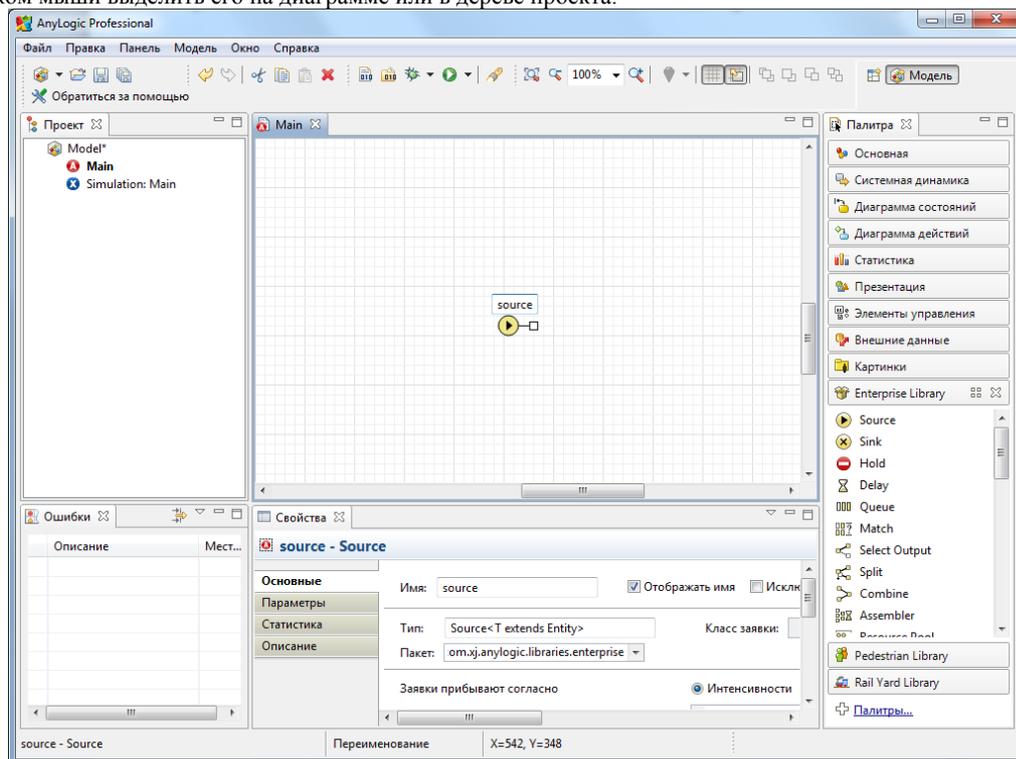


Рис.3.4. Развернутая библиотека Enterprise Library и ее элемент, помещенный на схему.

Объекты должны взаимодействовать между собой, поэтому вы должны будете соединять их друг с другом. Можно соединять объекты с помощью мыши, перетаскиванием порта одного объекта на порт другого или с помощью специального средства «Соединитель». Чтобы соединить порты объектов, щелкните мышью по кнопке панели инструментов **Соединитель**, а затем щелкните мышью поочередно по обоим портам. Для добавления точки изгиба щелкните мышью по кнопке панели инструментов **Редактировать точки**.

Модель выполняется в соответствии с набором конфигурационных установок, называемым *экспериментом*. Вы можете создать несколько экспериментов и изменять рабочую конфигурацию модели, просто меняя текущий эксперимент модели. Один эксперимент, названный *Simulation*, создается автоматически. Выберите его щелчком мыши по элементу дерева и измените настройки модели в окне **Свойства** (рис. 3.5). Окно **Свойства** имеет вкладки: основные, дополнительные, модельное время, презентация, окно, параметры, описание.

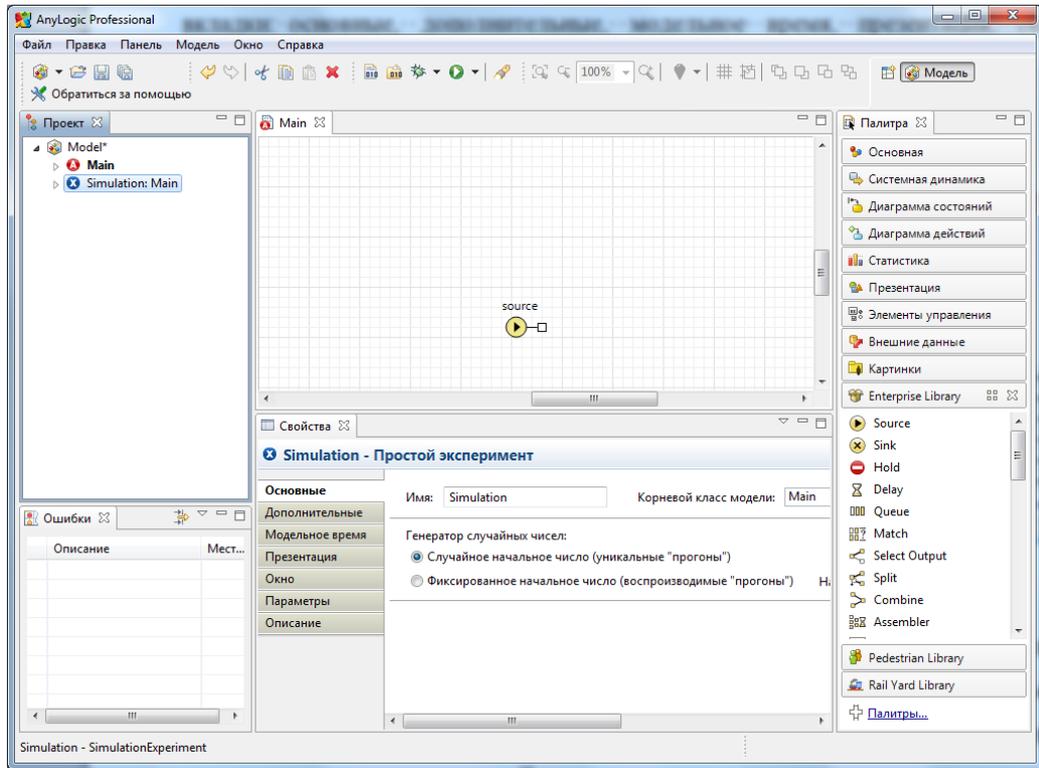


Рис. 3.5 Задание свойств эксперимента.

На вкладке **Основные** можно выбрать класс, который будет запущен при запуске модели. По умолчанию в качестве корневого объекта выбран объект класса *Main*, автоматически создаваемого в каждой модели. Вы можете переименовывать классы модели. Для этого нужно выделить класс щелчком мыши по значку класса в дереве модели и затем изменить его имя в окне **Свойства**.

На вкладке **Модельное время** можно:

1. задать режим моделирования. В режиме реального времени задается связь модельного времени с физическим, т.е. задается количество единиц модельного времени, выполняемых в одну секунду. Режим реального времени лучше всего подходит для показа анимации. В режиме виртуального времени модель выполняется без привязки к физическому времени – она выполняется так быстро, как это возможно. Данный режим лучше всего подходит, когда требуется моделировать работу системы в течение достаточно длительного периода времени;
2. запустить модель так, чтобы она работала бесконечно, но можно и остановить ее в заданный момент времени. Вы можете остановить модель по достижении переменной заданного значения или по выполнении какого-нибудь определенного условия.

Дополнительные свойства эксперимента (вкладка **Дополнительные**) позволяют управлять выполнением модели. Можно задать действие перед и после запуска модели, а также задать численные методы для прогона и точность получаемых значений. На вкладке **Презентация** можно определить вид и скорость выполнения прогона.

### Моделирование одноканальной смо с очередью.

**Постановка задачи.** В банковский офис обращаются клиенты. Офис представляет собой автоматизированный пункт обслуживания, в котором установлен банкомат. Банкомат обслуживает одновременно одного клиента. Клиенты прибывают по экспоненциальному закону с интенсивностью  $\lambda=0,67$ . Одновременно в офисе может находиться не более 15 клиентов. Интервал времени работы банкомата подчиняется треугольному закону распределения с параметрами  $x_{\min}=0,8$ ,  $x_{\max}=1,3$  предпочтительное значение 1.

**Построение модели.** Модель строится с «нуля». Банковский офис представляет собой систему массового обслуживания(СМО). Построение модели такой системы выполняется с помощью элементов библиотеки Enterprise Library Для построения СМО используются элементы:

- Source – источник заявок.
- Queue – очередь ожидающих обслуживания заявок.
- Delay – Элемент моделирующий узел обслуживания.
- Sink – Элемент принимающий отработанные заявки.

Общий вид модели СМО банковского офиса показан на рисунке 3.6.

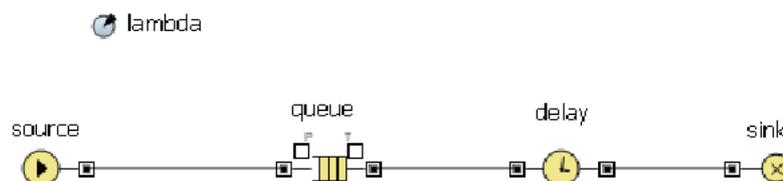


Рис. 3.6. Модель офиса

#### Источник заявок

Заявки – клиенты офиса пребывают с интенсивностью  $\lambda=0,67$ . Источник заявок обладает следующими настройками:

- Заявки пребывают согласно интенсивности.
- Интенсивность прибытия равна  $\lambda$ . **Lambda** – параметр (панель «Основная» пункт параметр). Значение соответствует интенсивности потока клиентов, и равно 0,67.

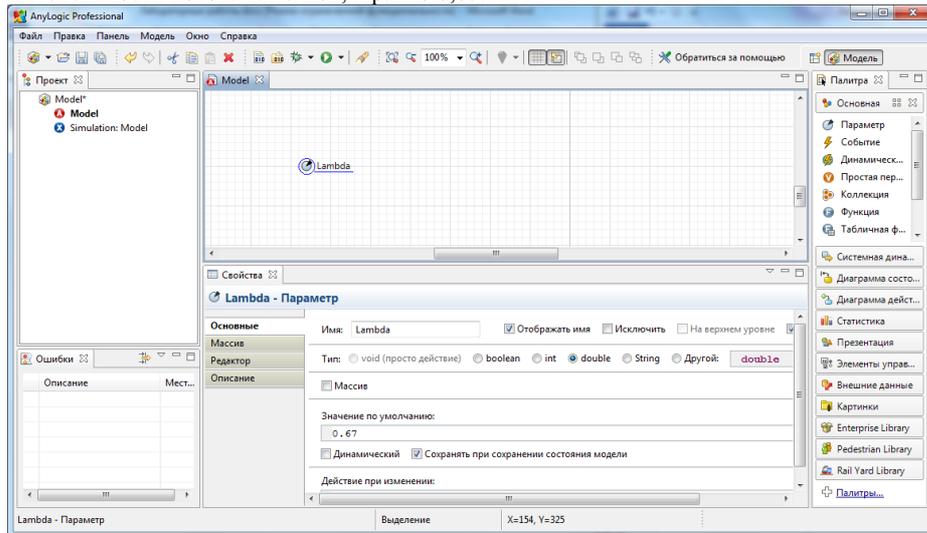


Рис. 3.7. Задание параметра интенсивности заявок.

Закон распределения потока заявок можно задать в свойстве *interarrivalTime* на вкладке *Параметры* для объекта *source*. По умолчанию распределение случайного потока заявок подчиняется экспоненциальному закону. –*exponential()*.

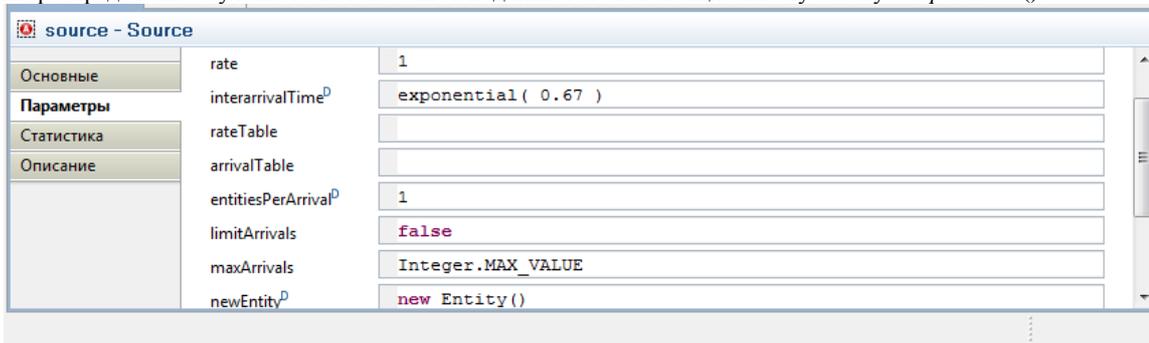


Рис. 3.8. Задание закона распределения потока заявок.

AnyLogic предоставляет функции и других случайных распределений, таких как:

- нормальное с дисперсией  $\sigma$  и мат. ожиданием  $m$  – *normal( $\sigma, m$ )*;
  - равномерное на отрезке  $[a, b]$  – *uniform( $a, b$ )*;
  - треугольное с минимальным значением  $a$ , средним значением  $b$  и максимальным  $c$  – *triangular( $a, b, c$ )*;
- и т.д.
- Количество заявок пребывающих за один раз равно единице.

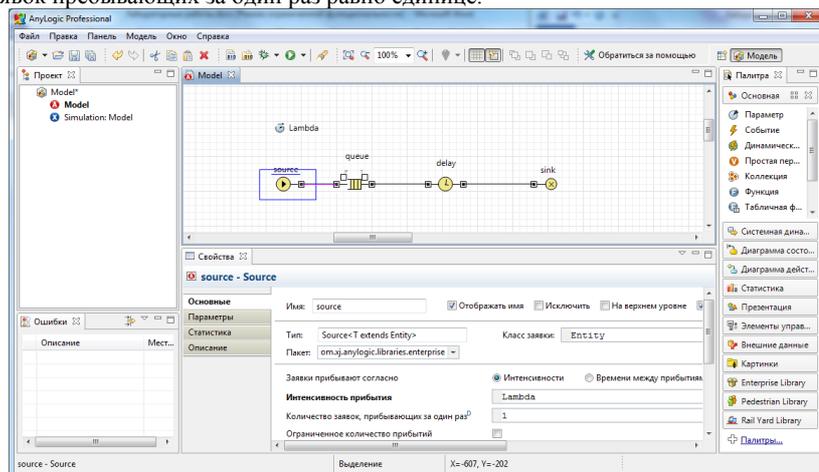


Рис. 3.9. Параметры источника заявок.

### Очередь

Этот элемент характеризуется параметрами:

- Вместимость очереди равна 15.
- Включить сбор статистики – да.

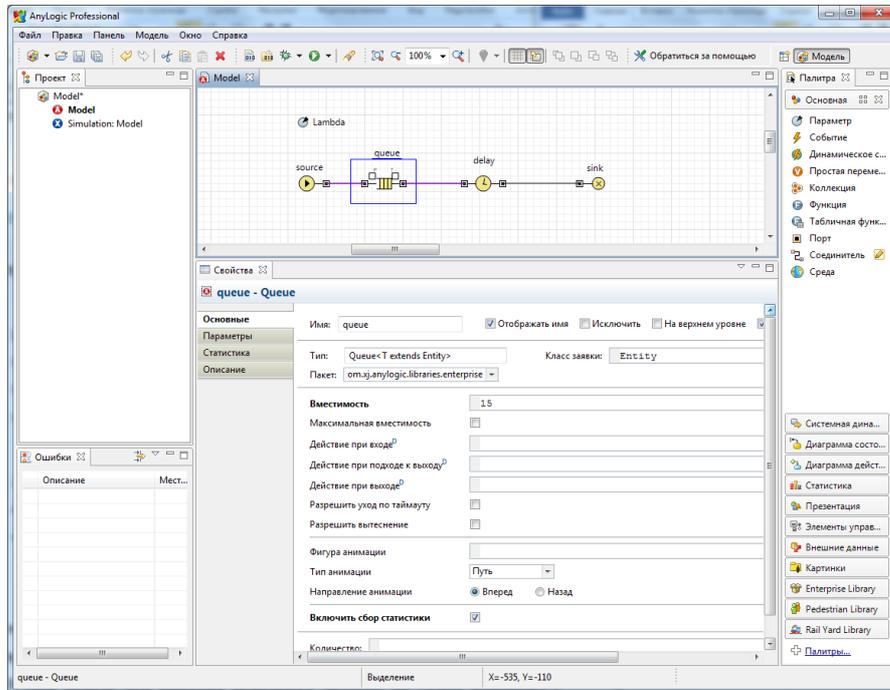


Рис. 3.10. Параметры очереди.

### Узел обслуживания

Параметры элемента:

- Задержка задается явно.
- Время задержки равно:  $\text{triangular}(0.8, 1.3, 1)$ .
- Вместимость узла – один клиент.
- Включить сбор статистики- да.

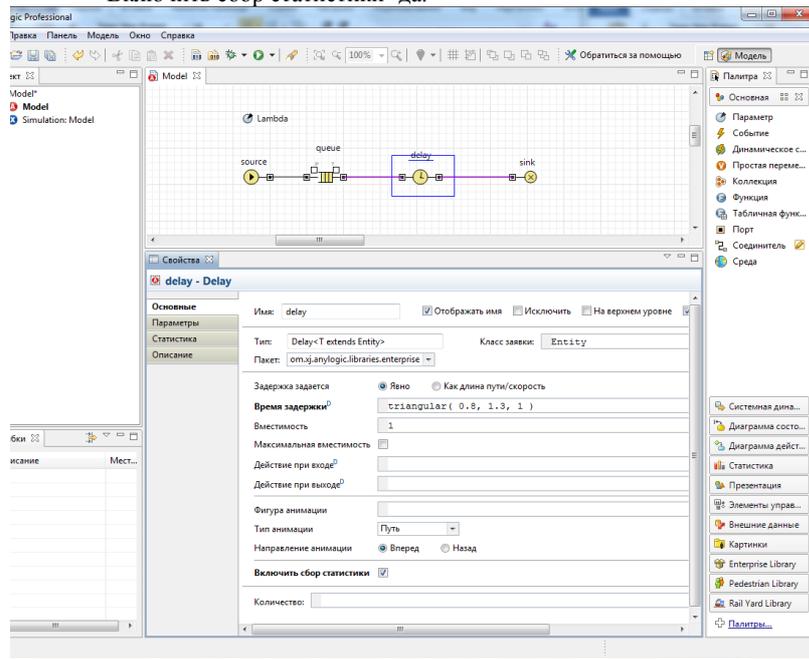


Рис. 3.11. Параметры узла обслуживания.

Элемент, принимающий заявки обладает параметрами настройки по умолчанию.

Настройте эксперимент модели:

- Модельное время – минуты.
- Время остановки модели не задано.
- Задайте Режим выполнения со скоростью 8.

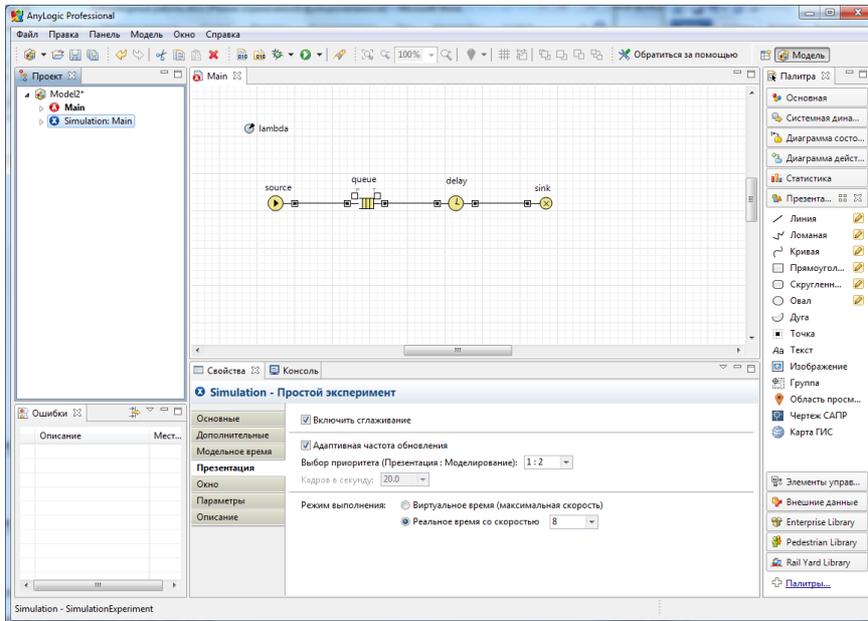


Рис. 3.12. Настройка параметров эксперимента.

Для запуска модели щелкните мышью по кнопке **Запустить**. Откроется окно с презентацией запущенного эксперимента. AnyLogic автоматически помещает на презентацию каждого простого эксперимента заголовок и кнопку, позволяющую запустить модель и перейти на презентацию.

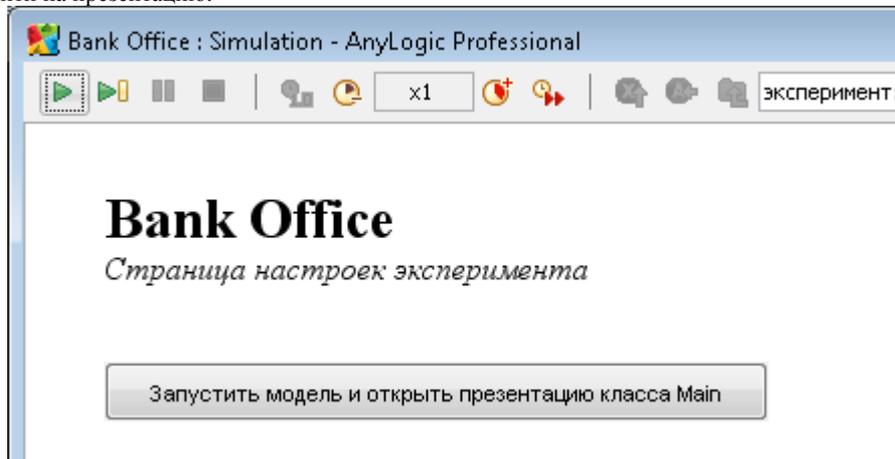


Рис. 3.13. Запуск эксперимента.

Щелкните по этой кнопке. AnyLogic переключится в режим работы модели. С помощью визуализированной блок-схемы вы можете проследить, сколько человек находится в очереди, сколько человек в данный момент обслуживается и т.д.

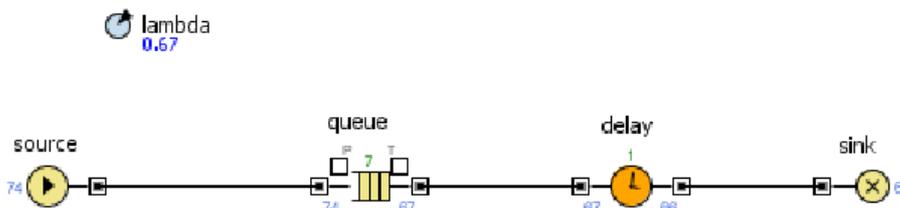


Рис.3.14. Вид работающей модели

### Анимация модели

Покажем процесс обслуживания клиентов в виде анимации очереди, ведущей к банкомату, так как это показано на рисунке 3.15.

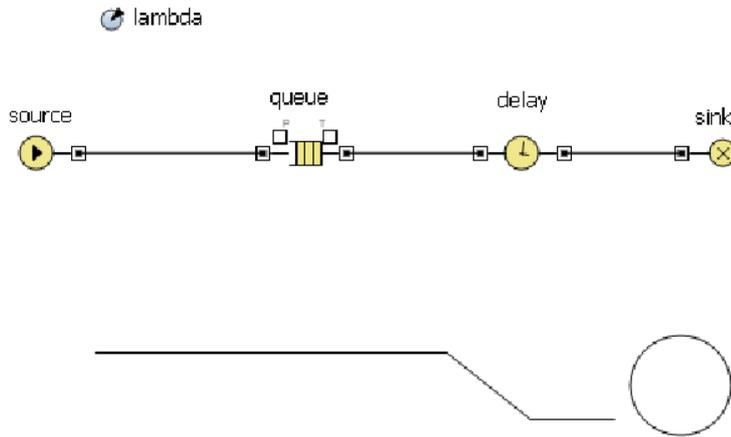


Рис.3.15. Анимация очереди

Банкомат представим в виде окружности. Когда клиент находится в банкомате, окружность будет окрашена в красный цвет, при свободном банкомате окружность закрашивается в зеленый цвет.

С помощью элемента «Овал» палитры «Презентация» разместите окружность и присвойте ей имя ServicePoint. Цвет заливки должен изменяться динамически:

`delay.size()>0 ? Color.red: Color.green`

Здесь `size()` – метод объекта `delay`, который возвращает количество заявок-клиентов в приборе обслуживания.

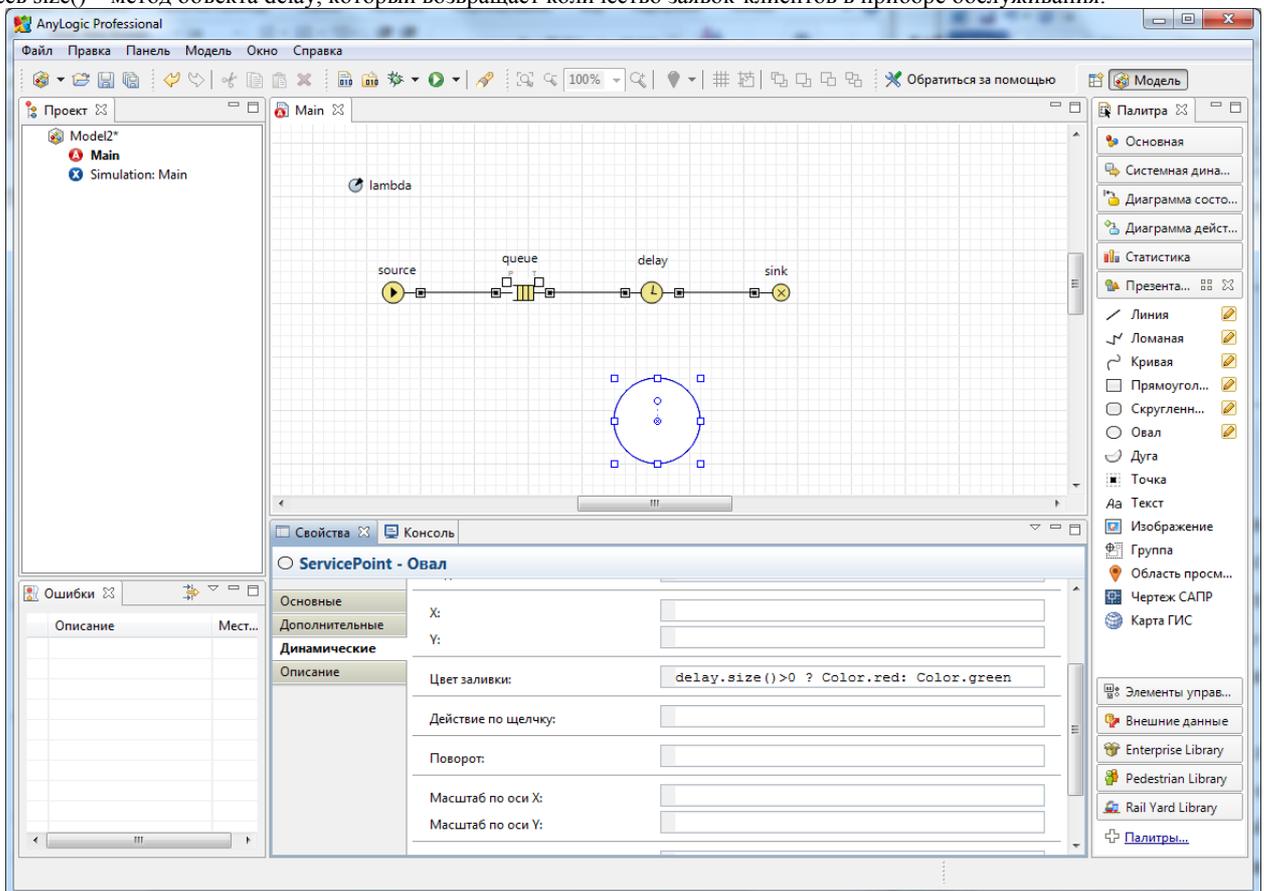


Рис.3.16. Задание свойств окружности – банкомата.

Для отображения очереди следует нарисовать ломаную линию (см. рисунок 3.15), используя элемент «Ломаная» из палитры «Презентация». Режим рисования включается после выполнения двойного щелчка по пиктограмме .

Рисование ломаной нужно выполнять по направлению движения клиентов к банкомату: слева на право. Ломаной присвойте имя `GoToService`.

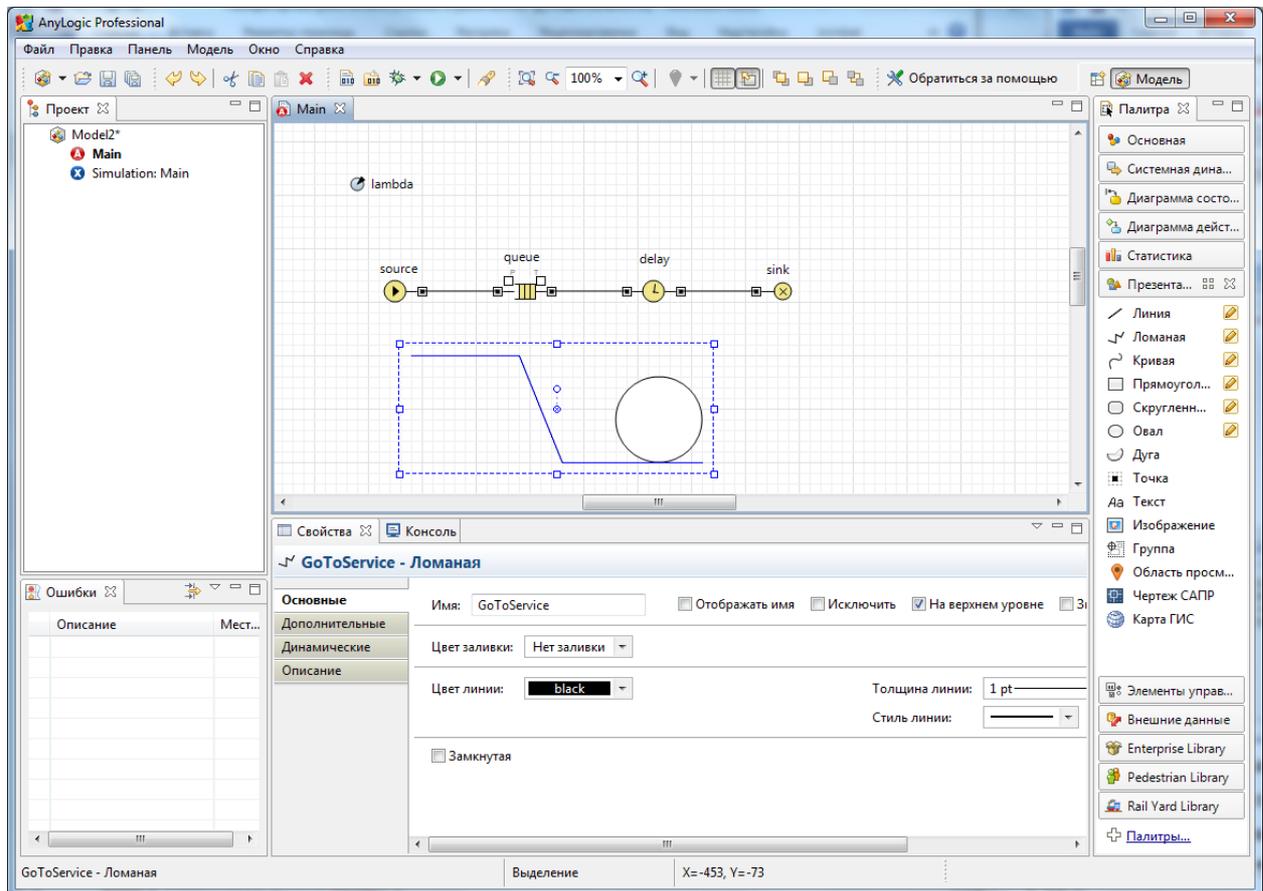


Рис.3.17. Задание свойств ломаной – очереди.

После создания элементов презентации нужно выполнить ряд настроек модели для связи графических элементов с объектами схемы.

Откройте окно свойств элемента очередь (queue) и на вкладке «Основные» задайте настройки так, как это показано на рисунке 3.18.

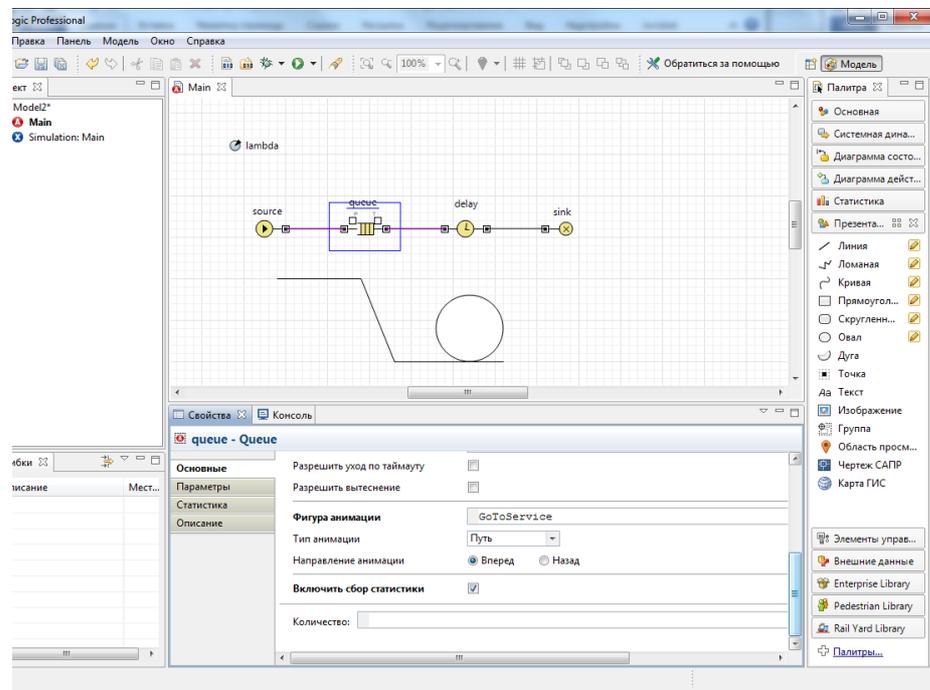


Рис.3.18. Настройка очереди

Откройте прибор обслуживания – элемент delay, и настройте на вкладке «Основные», свойства анимации:

- Фигура анимации: ServicePoint
- Тип анимации: Одиночная

Установите режим скорости исполнения равным 4 и протестируйте модель. На рисунке 3.19. Показан вид работающей модели.

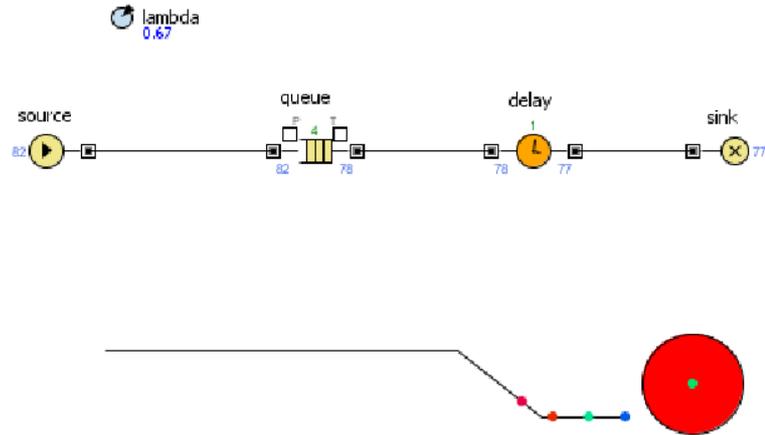


Рис.3.19. Модель с анимацией

**Размещение датчиков.**

Чтобы представить процесс загрузки прибора обслуживания и очереди разместим два датчика – столбчатые диаграммы. Первая диаграмма отображает среднее значение клиентов в очереди, а вторая - среднее значение числа обслуженных клиентов в банкомате (приборе обслуживания).

Для размещения диаграмм нужно использовать палитру «Статистика» и элемент «Столбчатая диаграмма». Разместите две диаграммы (см. рисунок 3.20).

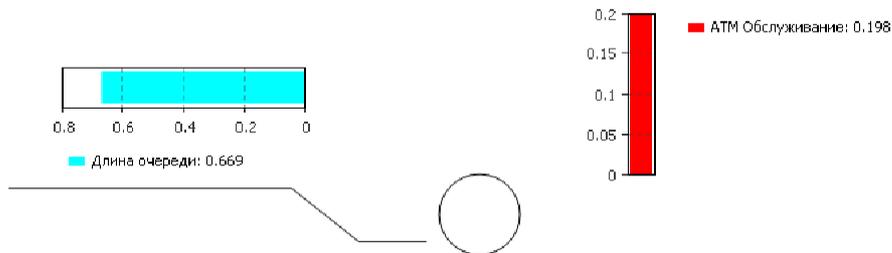


Рис.3.20. Диаграммы загрузки

Первую поместите над очередью. Добавьте элемент данных. Задайте подпись «Длина очереди». Выберите цвет, а затем в качестве значения задайте выражение:

```
queue.statsSize.mean()
```

Здесь метод mean() – возвращает среднюю длину очереди.

При вводе выражения можно использовать помощник AnyLogic. Для этого следует нажать комбинацию клавиш CTRL + SPACE.

После ввода выражения нужно сменить ориентацию диаграммы на горизонтальное. Для этого необходимо открыть вкладку «Внешний вид» и изменить направление столбцов (см. рисунок 3.21).

Столбцы

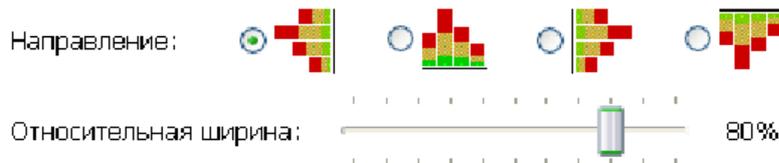


Рис.3.21. Направление столбцов диаграммы

Вторую диаграмму расположите рядом с изображением банкомата. Назовите диаграмму «АТМ Обслуживание», а в качестве значения задайте выражение:

```
delay.statsUtilization.mean()
```

задающее среднее время обслуживания заявки в процессоре. Направление столбцов вертикальное.

Вид работающей модели показан на рисунке 3.22.

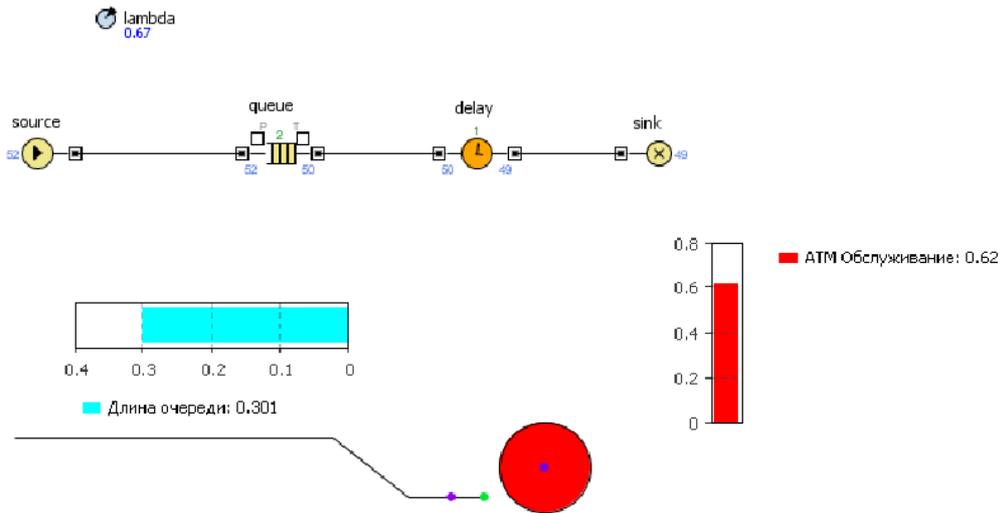


Рис.3.22. Модель с диаграммами

### Моделирование многоканальной смо с очередью.

Усложним модель, добавив в нее банковских кассиров. Можно моделировать число кассиров, как и банкомат, с помощью объектов *delay*. Но куда более удобным представляется моделирование числа кассиров с помощью ресурсов. **Ресурс** – это специальный объект Enterprise Library, который может потребоваться заявке для выполнения какой-то задачи. В нашем примере посетителям банковского отделения (заявкам) необходимо получить помощь у банковских служащих (ресурсов).

Добавьте на диаграмму следующие объекты:

1. *selectOutput* – является блоком принятия решения. В зависимости от заданного вами условия, заявка, поступившая в этот объект, будет поступать на один из двух выходов объекта. Оставьте свойство *selectCondition* – *uniform() < 0.5*, тогда к кассирам и банкомату будет приходить примерно равное количество клиентов;
2. *Service* – моделирует занятие заявкой ресурса на определенное время. С помощью этого объекта мы промоделируем обслуживание клиента кассиром. Задайте следующие свойства объекта: назовите объект *tellerLines* (свойство **Имя**); укажите, что в очереди к кассирам может находиться до 20 человек (свойство *queueCapacity*); задайте время обслуживания (свойство *delayTime*). Будем полагать, что время обслуживания имеет треугольное распределение с минимальным средним значением 2.5, средним – 6 и максимальным – 11 минут;

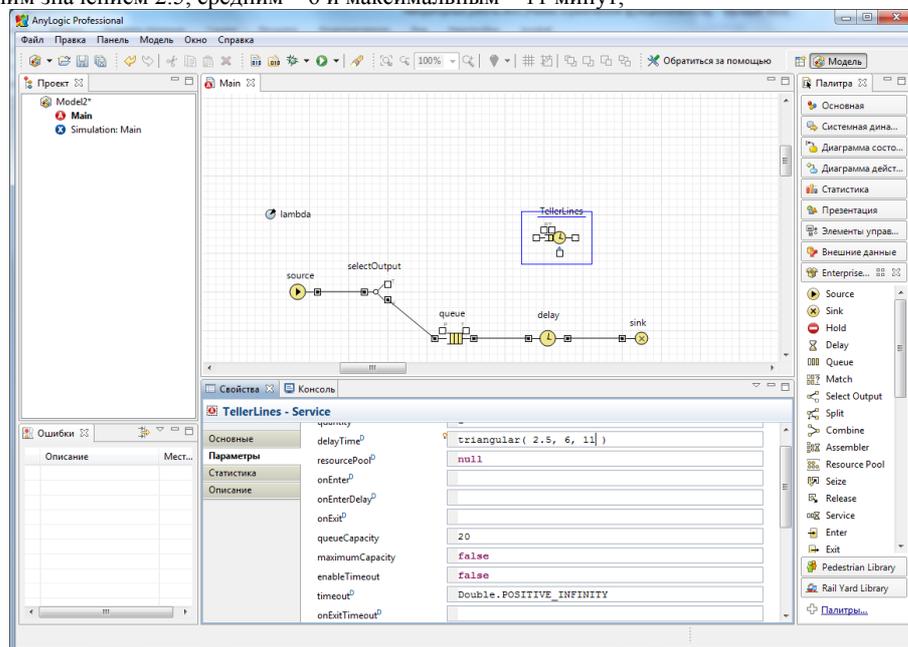


Рис. 3.23. Задание свойств линии касс.

3. *ResourcePool* – задает ресурсы определенного типа. Он должен быть подсоединен к объектам, моделирующим занятие и освобождение ресурсов (в нашем случае это объект *Service*). Задайте следующие свойства объекта: назовите объект *tellers*; задайте число кассиров (свойство *capacity*) – 4.

Измените имя объекта *delay* на *АТМ* (банкомат). Соедините объекты соответствующим образом (рис. 3.24).

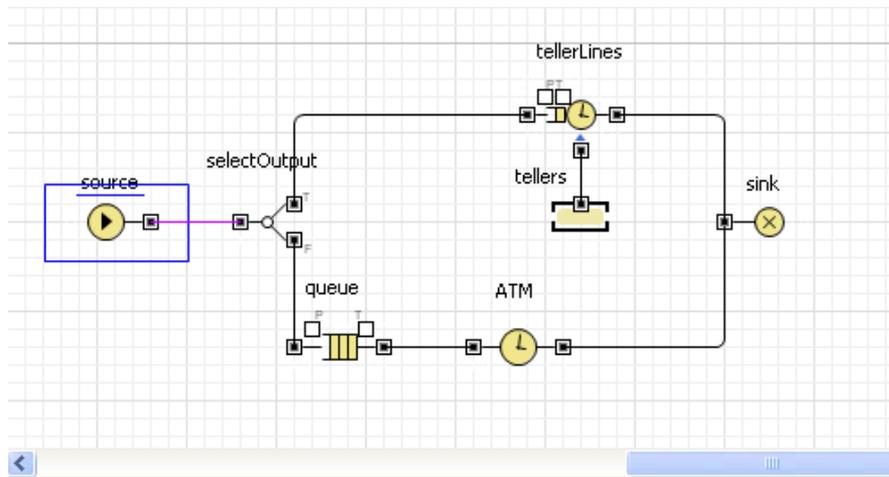


Рис. 3.24. Вид двухканальной СМО.

Запустите модель и изучите ее поведение.

Сбор статистики о времени обслуживания клиента.

Необходимо определить, сколько времени клиент проводит в банковском отделении и сколько времени он теряет, ожидая своей очереди. Соберем эту статистику с помощью специальных объектов сбора данных и отобразим собранную статистику распределения времени обслуживания клиентов с помощью гистограмм.

Создадим класс сообщения *Customer*. Сообщения этого класса будут представлять клиентов банковского отделения. Выберите базовый класс *Entity*(сообщения), добавьте параметры для хранения информации о проведенном времени:

1. в панели **Проект**, щелкните правой кнопкой мыши по элементу модели и выберите **Создать | Java класс** из контекстного меню (рис. 3.25);

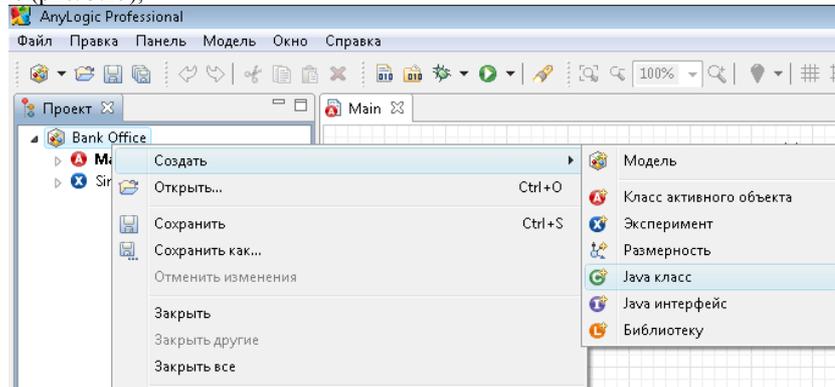


Рис. 3.25 Добавление Java- класса.

2. появится диалоговое окно **Новый Java класс**. В поле **Имя** введите имя нового класса *Customer*;
3. сделайте так, чтобы этот класс наследовался от базового класса заявки *Entity* (рис. 3.26): выберите из выпадающего списка **Базовый класс** полное имя данного класса: *com.xj.anylogic.libraries.enterprise.Entity*;

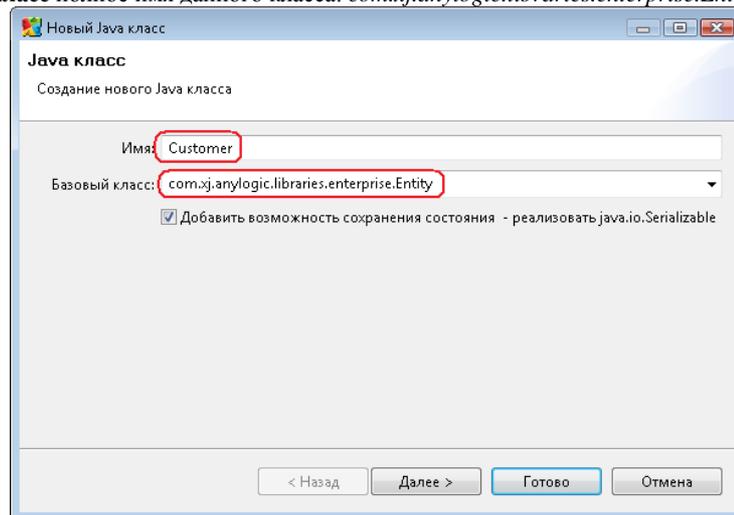


Рис. 3.26 Задание свойств Java- класса.

4. щелкните мышью по кнопке **Далее**. На второй странице Мастера вы можете задать параметры создаваемого Java-класса. Создайте параметры:
    - *enteredSystem* типа *double* для сохранения момента времени, когда клиент пришел в банковское отделение;
    - *startWaiting* типа *double* для сохранения момента времени, когда клиент встал в очередь к банкомату;
- щелкните мышью по кнопке **Готово**. Вы увидите редактор кода созданного класса. Можете закрыть его, щелкнув мышью по крестике в закладке с его названием.

**Те-**

**перь вычислим время, которое тратится персоналом банка на обслуживание клиентов, и время, которое клиенты тратят на ожидание своей очереди.**

Для этого добавьте элементы сбора статистики по времени ожидания клиентов и времени пребывания клиентов в системе. Эти элементы будут запоминать соответствующие значения времени для каждого клиента и предоставят пользователю стандартную статистическую информацию: среднее, минимальное, максимальное из измеренных значений, среднеквадратичное отклонение, доверительный интервал для среднего и т.п.:

- чтобы добавить объект сбора данных гистограммы на диаграмму, перетащите элемент **Данные гистограммы** с палитры **Статистика** на диаграмму активного класса;
- задайте свойства элемента (рис. 3.27).
  - Измените **Имя** на *waitTimeDistr*.
  - Измените **Заголовок** на *Waiting time distribution*.
  - Сделайте **Кол-во интервалов** равным 50.
  - Задайте **Начальный размер интервала**: 0.01;

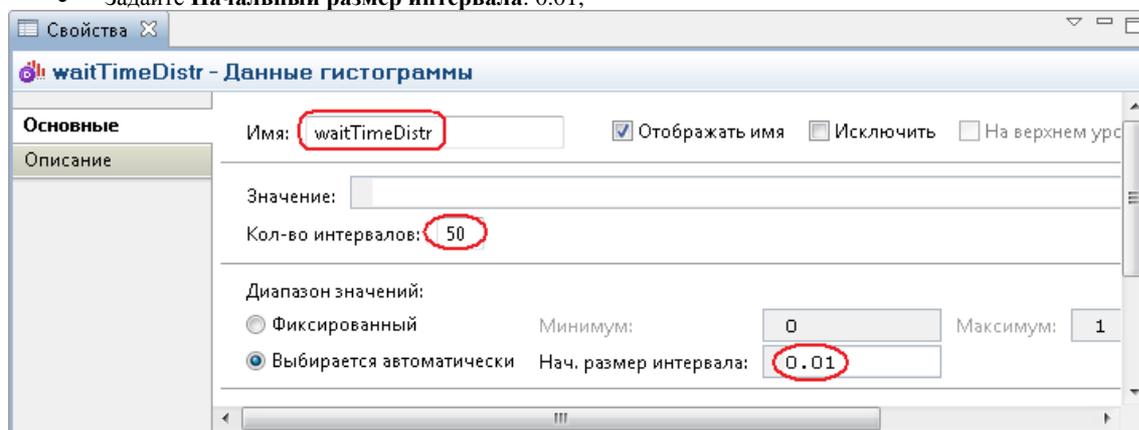


Рис. 3.27 Задание свойств гистограммы.

- создайте еще один элемент сбора данных гистограммы (**Ctrl** + перетащите только что созданный объект данных гистограммы, чтобы создать его копию). Измените **Имя** этого элемента на *timeInSystemDistr*, а **Заголовок** на *Time in system distribution*.

Измените свойства блоков вашей диаграммы процесса. Задайте следующие свойства объектов диаграммы:

- блок *source*, свойство **Новая заявка** – введите `new Customer()`. Введите `Customer` в поле **Класс заявки**. Это позволит напрямую обращаться к полям класса заявки `Customer` в коде динамических параметров этого объекта. Введите `entity.enteredSystem = time()`; в поле **Действие привходе**. Этот код будет сохранять время создания заявки-клиента в переменной `enteredSystem` нашего класса заявки `Customer`. Функция `time()` возвращает текущее значение модельного времени;
- блок *tellerLines* (блок *Service*) – введите `Customer` в поле **Класс заявки**. Добавьте код в поля:

**Действие при входе:** `entity.startWaiting = time()`;

**Действие при выходе:** `waitTimeDistr.add(time() - entity.startWaiting)`;

- блок *queue* – введите `Customer` в поле **Класс заявки**. Добавьте код в поля: **Действие при входе:** `entity.startWaiting = time()`;

**Действие при выходе:** `waitTimeDistr.add(time()-entity.startWaiting)`

Данный код добавляет время, в течение которого клиент ожидал обслуживания в объект сбора данных *waitTimeDistr*;

- блок *ATM* (блок *delay*) – введите `Customer` в поле **Класс заявки**;
- блок *sink* – введите `Customer` в поле **Класс заявки**. Напишите следующий код, чтобы сохранить в наборах данных данные о клиенте, покидающем банковское отделение (**Действие при входе**):

`timeInSystemDistr.add(time()-entity.enteredSystem)`;

Данный код добавляет полное время пребывания клиента в банковском отделении в объект сбора данных гистограммы *timeInSystemDistr*.

Добавьте две гистограммы для отображения распределений времен ожидания клиента и пребывания клиента в системе.

Чтобы добавить гистограмму на диаграмму класса активного объекта, перетащите элемент **Гистограмма** из палитры **Статистика** в то место, куда вы хотите ее поместить. Укажите, какой элемент сбора данных хранит данные, которые хотите отображать на гистограмме: щелкните мышью по кнопке **Добавить данные** и введите в поле **Данные** имя соответствующего элемента – *waitTimeDistr*.

Аналогичным образом добавьте еще одну гистограмму и расположите ее под ранее добавленной. В поле **Данные** введите *timeInSystemDistr*. Измените заголовки отображаемых данных.

Запустите модель. Включите режим виртуального времени и посмотрите, какой вид примет распределение времени ожидания и времени пребывания клиента в системе.

Индивидуальные варианты заданий.

Внесите изменения в модель банковского отделения согласно варианту (параметры законов распределения задайте произвольно).

| Вариант | Распределение вероятности прихода клиентов в банк | Вероятность обращения к кассиру/к банкомату | Время обслуживания клиента кассиром | Количество кассиров |
|---------|---------------------------------------------------|---------------------------------------------|-------------------------------------|---------------------|
| 1       | Экспоненциальное                                  | 1/1                                         | 4±2                                 | 1                   |
| 2       | Экспоненциальное                                  | 1/2                                         | 6±2                                 | 2                   |
| 3       | Экспоненциальное                                  | 2/1                                         | 8±2                                 | 3                   |
| 4       | Экспоненциальное                                  | 1/3                                         | 7±2                                 | 4                   |

|    |                  |     |     |   |
|----|------------------|-----|-----|---|
| 5  | Экспоненциальное | 3/1 | 9±2 | 5 |
| 6  | Нормальное       | 1/1 | 8±2 | 1 |
| 7  | Нормальное       | 1/2 | 7±2 | 2 |
| 8  | Нормальное       | 2/1 | 9±2 | 3 |
| 9  | Нормальное       | 1/3 | 4±2 | 4 |
| 10 | Нормальное       | 3/1 | 6±2 | 5 |
| 11 | Треугольное      | 1/1 | 2±2 | 1 |
| 12 | Треугольное      | 1/2 | 7±2 | 2 |
| 13 | Треугольное      | 2/1 | 9±2 | 3 |
| 14 | Треугольное      | 1/3 | 4±2 | 4 |
| 15 | Треугольное      | 3/1 | 6±2 | 5 |
| 16 | Равномерное      | 1/1 | 4±2 | 1 |
| 17 | Равномерное      | 1/2 | 6±2 | 2 |
| 18 | Равномерное      | 2/1 | 7±2 | 3 |
| 19 | Равномерное      | 1/3 | 8±2 | 4 |
| 20 | Равномерное      | 3/1 | 9±2 | 5 |

Проанализируйте поведение модели. Постройте графики и диаграммы.

**Текущий контроль  
по МДК.06.01 Внедрение информационных систем  
Контрольная работа:**

**Текущий контроль  
по МДК. 06.02 Инженерно-техническая поддержка сопровождения информационных систем  
Контрольная работа:**

**1. Базы данных — это:**

- A) информационные структуры, хранящиеся во внешней памяти;
- B) программные средства, позволяющие организовывать информацию в виде таблиц;
- C) программные средства, обрабатывающие табличные данные;
- D) программные средства, осуществляющие поиск информации.

**2. В коробке меньше 9, но больше 3 шаров. Сколько шаров может быть в коробке?**

- A) 3; B) 9; C) 2; D) 5; E) 10.

**3. Какие атрибуты (признаки) объекта должны быть отражены в информационной модели, описывающей хобби ваших одноклассников, если эта модель позволяет получить ответы на следующие вопросы:**

- Каков возраст всех детей, увлекающихся компьютером?
- Каковы имена девочек, увлекающихся пением?
- Каковы фамилии мальчиков, увлекающихся хоккеем?

- A) имя, пол, хобби;
- B) фамилия, пол, хоккей, пение, возраст;
- C) имя, пол, хобби, возраст;
- D) имя, возраст, хобби;
- E) фамилия, имя, пол, возраст, хобби?

**4. Реляционная база данных задана таблицей:**

|   | Ф.И.О         | Пол | Возраст | Клуб    | Спорт   |
|---|---------------|-----|---------|---------|---------|
| 1 | Панько Л.П.   | жен | 22      | Спартак | футбол  |
| 2 | Арбузов А.А.  | муж | 20      | Динамо  | лыжи    |
| 3 | Жиганова П.Н. | жен | 19      | Ротор   | футбол  |
| 4 | Иванов О.Г.   | муж | 21      | Звезда  | лыжи    |
| 5 | Седова О.Л.   | жен | 18      | Спартак | биатлон |
| 6 | Багаева СИ.   | жен | 23      | Звезда  | лыжи    |

Какие записи будут выбраны по условию: Спорт= "лыжи" И Пол= "жен" ИЛИ Возраст

- A) 2, 3, 4, 5, 6; B) 3, 5, 6; C) 1, 3, 5, 6; D) 2, 3, 5, 6; E) таких записей нет.

**5. Реляционная БД задана таблицей:**

|   | Название     | Категория | Кинотеатр | Начало сеанса |
|---|--------------|-----------|-----------|---------------|
| 1 | Буратино     | х/ф       | Рубин     | 14            |
| 2 | Кортик       | х/ф       | Искра     | 12            |
| 3 | Винни-Пух    | м/ф       | Экран     | 9             |
| 4 | Дюймовочка   | м/ф       | Россия    | 10            |
| 5 | Буратино     | х/ф       | Искра     | 14            |
| 6 | Ну, погоди   | м/ф       | Экран     | 14            |
| 7 | Два капитана | х/ф       | Россия    | 16            |

Выбрать первичный ключ для таблицы (допуская, что в кинотеатре один зал):

- A) Название+Кинотеатр;
- B) Кинотеатр+Начало сеанса;
- C) Название+Начало сеанса;
- D) Кинотеатр;
- E) Начало сеанса.

**6. Структура реляционной базы данных изменяется при:**

- A) удалении любой записи;
- B) удалении любого поля;
- C) изменении любой записи;
- D) добавлении записи;
- E) удалении всех записей.

**7. Реляционная база данных задана таблицей. Записи в таблице пронумерованы.**

|   | Код дистанции | Код соревнований | Дата       | Время спортсмена (с) |
|---|---------------|------------------|------------|----------------------|
| 1 | 101           | Д02              | 11.12.2004 | 56,6                 |
| 2 | 104           | Д01              | 12.10.2005 | 37                   |

|   |     |     |            |       |
|---|-----|-----|------------|-------|
| 3 | 102 | Д02 | 11.12.2005 | 56,1  |
| 4 | 103 | Д05 | 11.12.2005 | 242,8 |
| 5 | 101 | Д04 | 13.01.2005 | 181,1 |
| 6 | 102 | Д01 | 12.10.2005 | 35,45 |

Сформулировать условие поиска, дающее сведения о спортсменах, принимавших участие в соревнованиях на дистанциях с кодами Д01 и Д03 не позднее 10.12.2004.

- А) Код\_дистанции="Д01" и Код\_дистанции="Д03" и Дата\_соревнования10.12.2004  
 В) (Код\_дистанции="Д01" или Код\_дистанции="Д03") и Дата\_соревнования10.12.2004  
 С) Код\_дистанции="Д01" и (Код\_дистанции="Д03" или Дата\_соревнования  
 Д) Код\_дистанции="Д01" и Код\_дистанции="Д03" и Дата\_соревнования  
 Е) (Код\_дистанции="Д01" или Код\_дистанции="Д03") и Дата\_соревнования

8. Дана однотабличная база данных «Автомобилисты»:

|   | Владелец    | Модель | Номер     | Дата регистрации |
|---|-------------|--------|-----------|------------------|
| 1 | Левченко Н. | Волга  | И537ИГ-59 | 15.08.2001       |
| 2 | Сидоров А.  | Жигули | Ф131ФП-59 | 14.02.2000       |
| 3 | Горохов И.  | Форд   | Б171БП-59 | 27.10.2000       |
| 4 | Федоров К.  | Волга  | И138ИП-59 | 20.05.2001       |
| 5 | Сидоров А.  | Жигули | И321ИП-59 | 27.10.2000       |

Отсортировать таблицу в порядке возрастания по двум полям: Модель+Номер.

- А) 1; 4; 2; 5; 3; ; В) 3; 4; 5; 1; 2; С) 4; 1; 5; 2; 3 Д) 3; 5; 2; 4; 1; Е) 2; 1; 5; 4; 3.

9. Поле м реляционной БД является:

- А) строка таблицы; В) корень дерева; С) дерево; Д) столбец таблицы; Е) ветви дерева.

10. Что может служить источником данных при построении запроса (в СУБД Access): (1) таблица, (2) запрос, (3) форма, (4) отчет?

- А) 1, 2; В) только 1; С) только 2; Д) 3; Е) 4.

### Контрольная работа № 3 Технологии использования и разработки информационных систем.

#### Вариант – 2.

1. В реляционной БД информация организована в виде:

- А) сети;  
 В) иерархической структуры;  
 С) файла;  
 Д) дерева;  
 Е) связанных прямоугольных таблиц.

2. БД содержит информацию об учениках школы: фамилия, класс, балл за тест, балл за практическое задание, общее количество баллов. Какого типа должно быть поле «Общее количество баллов»?

- А) текстовое; С) числовое; Е) любого типа.  
 В) логическое; Д) «дата/время»;

3. Реляционная база данных задана таблицей:

|   | Ф.И.О         | Пол | Возраст | Клуб    | Спорт   |
|---|---------------|-----|---------|---------|---------|
| 1 | Панько Л.П.   | жен | 22      | Спартак | футбол  |
| 2 | Арбузов А.А.  | муж | 20      | Динамо  | лыжи    |
| 3 | Жиганова П.Н. | жен | 19      | Ротор   | футбол  |
| 4 | Иванов О.Г.   | муж | 21      | Звезда  | лыжи    |
| 5 | Седова О.Л.   | жен | 18      | Спартак | биатлон |
| 6 | Багаева СИ.   | жен | 23      | Звезда  | лыжи    |

Какие записи будут выбраны по условию: (Клуб= "Спартак" И Клуб= "Ротор") И НЕ (Пол="жен")

- А) 3, 5; Д) 2, 4;  
 В) 1, 3, 5; Е) таких записей нет.  
 С) 2, 3, 4, 5;

4. БД содержит информацию о собаках из клуба собаководства: кличка, порода, дата рождения, пол, количество медалей. Какого типа должны быть поля?

- А) текстовое, текстовое, числовое, текстовое, числовое;  
 В) текстовое, текстовое, дата/время, текстовое, числовое;  
 С) текстовое, текстовое, дата/время, логическое, числовое;  
 Д) текстовое, текстовое, числовое, логическое, числовое;  
 Е) текстовое, текстовое, дата/время, логическое, текстовое.

5. Реляционная БД задана таблицей:

| Название | Категория | Кинотеатр | Начало сеанса |
|----------|-----------|-----------|---------------|
|----------|-----------|-----------|---------------|

|   |              |     |        |    |
|---|--------------|-----|--------|----|
| 1 | Буратино     | х/ф | Рубин  | 14 |
| 2 | Кортик       | х/ф | Искра  | 12 |
| 3 | Винни-Пух    | м/ф | Экран  | 9  |
| 4 | Дюймовочка   | м/ф | Россия | 10 |
| 5 | Буратино     | х/ф | Искра  | 14 |
| 6 | Ну, погоди   | м/ф | Экран  | 14 |
| 7 | Два капитана | х/ф | Россия | 16 |

В каком порядке будут идти записи, если их отсортировать по двум ключам: Название+Кинотеатр в порядке возрастания?

- A) 1, 5, 3, 4, 7, 2, 6; D) 6, 2, 7, 4, 3, 1, 5;  
 B) 5, 1, 3, 7, 4, 2, 6; E) 2, 5, 4, 7, 1, 3, 6.  
 C) 6, 2, 4, 7, 3, 1, 5;

**6. Какие характеристики объекта «Склад» должны быть отражены в структуре реляционной базы данных, если необходимо получить следующую информацию:**

- наименование и количество товара с истекшим сроком хранения (дата окончания срока хранения превысила текущую дату);
- наименование товара с ценой менее 70 руб.;
- наименование всех товаров на общую сумму более 2 000 руб.?

Построенная модель не должна содержать избыточную информацию.

- A) наименование, количество, цена, дата окончания срока хранения, общая сумма;  
 B) наименование, количество, цена, дата окончания срока хранения, текущая дата, общая сумма;  
 C) наименование, количество, цена, дата окончания срока хранения;  
 D) наименование, количество, цена, дата окончания срока хранения, текущая дата;  
 E) наименование, количество, цена, текущая дата, общая сумма.

**7. Дана однотабличная база данных «Автомобилисты»:**

|   | Владелец    | Модель | Номер     | Дата регистрации |
|---|-------------|--------|-----------|------------------|
| 1 | Левченко Н. | Волга  | И537ИГь59 | 15.08.2001       |
| 2 | Сидоров А.  | Жигули | Ф131ФП-59 | 14.02.2000       |
| 3 | Горохов И.  | Форд   | Б171БП-59 | 27.10.2000       |
| 4 | Федоров К.  | Волга  | И138ИП-59 | 20.05.2001       |
| 5 | Сидоров А.  | Жигули | И321ИП-59 | 27.10.2000       |

Какие записи будут удовлетворять условию отбора: Дата регистрации13.02.2000 и Дата регистрации

- A) 4; B) 2; 3; 5; C) 1; 4; D) 1; E) таких записей нет.

**8. Сформулировать условие отбора, позволяющее получить номера Волг и Жигулей, зарегистрированных ранее 01.01.2001:**

- A) Модель="Волга" или Модель="Жигули" и Дата регистрации01.01.2001  
 B) Модель="Волга" или Модель="Жигули" или Дата регистрации01.01.2001  
 C) Модель= "Волга" и Модель="Жигули" и Дата регистрации  
 D) (Модель="Волга" или Модель="Жигули") и Дата регистрации  
 E) Модель="Волга" и Модель="Жигули" или Дата регистрации

**9. Одним из основных типов информационных структур является:**

- A) логическая; B) база данных; C) строковая; D) дерево; E) числовая.

**10. Дано логическое выражение НЕ (а И b), где а и b – логические величины. При выполнении какого из следующих высказываний данное выражение будет ложным?**

- A) а и b имеют значение ИСТИНА;

## ЭКЗАМЕН

за 6 семестр

по МДК. 06.01 Внедрение информационных систем  
 Вопросы для сдачи экзамена

1. Жизненный цикл информационных систем.
2. Классификация информационных систем.
3. Основные методологии разработки информационных систем: MSF, RUP и т.п. ГОСТ Р ИСО/МЭК 12207.
4. Основные процессы и взаимосвязь между документами в информационной системе согласно стандартам.
5. Техническое задание: основные разделы согласно стандартам.
6. Виды внедрения, план внедрения.
7. Макетирование.

8. Пилотный проект.
9. Стратегии, цели и сценарии внедрения.
10. Структура и этапы проектирования информационной системы
11. Локальные акты.
12. Обучение группы внедрения.
13. Обучающая документация.
14. Стандарты ЕСПД.
15. Методы разработки обучающей документации.
16. Порядок внесения и регистрации изменений в документации.
17. Установка, конфигурирование и настройка сетевых и телекоммуникационных средств
18. . Формирование интерфейсов и организация доступа пользователей к информационной системе.
19. Режимы оповещения пользователей.
20. Организация мониторинга процесса внедрения.
21. Оформление результатов внедрения.
22. Оценка качества функционирования информационной системы. CALS-технологии

### **ЭКЗАМЕН**

**за 6 семестр**

**по МДК. 06.02 Инженерно-техническая поддержка сопровождения информационных систем**

**Вопросы для сдачи экзамена**

1. Задачи сопровождения информационной системы.
2. Ролевые функции и организация процесса сопровождения.
3. Сценарий сопровождения.
4. Договор на сопровождение.
5. Анализ исходных программ и компонентов программного средства.
6. Программная инженерия и оценка качества.
7. Реинжиниринг.
8. Цели и регламенты резервного копирования.
9. Сохранение и откат рабочих версий системы.
10. Сохранение и восстановление баз данных.
11. Организация процесса обновления в информационной системе.
12. Регламенты обновления.
13. Обеспечение безопасности функционирования информационной системы.
14. Организация доступа пользователей к информационной системе.
15. Организация сбора данных об ошибках в информационных системах, источники сведений
16. Системы управления производительностью приложений.
17. Мониторинг сетевых ресурсов.
18. Схемы и алгоритмы анализа ошибок, использование баз знаний.
19. Отчет об ошибках системы: содержание, использование информации.
20. Методы и инструменты тестирования приложений.

### **ЭКЗАМЕН**

**за 6 семестр**

**по МДК. 6.03 Устройство и функционирование информационной системы**

**Вопросы для сдачи экзамена**

1. Базовая структура информационной системы.
2. Основное оборудование системной интеграции.
3. Особенности информационного, программного и технического обеспечения различных видов АИС.
4. Особенности сопровождения информационных систем бухгалтерского учета и материально-технического снабжения.

5. Особенности сопровождения информационных систем управления качеством, технической и технологической подготовки производства.
6. Особенности сопровождения информационных систем удаленного управления и контроля объектов.
7. Особенности сопровождения информационных систем реального времени.
8. Структура и этапы проектирования информационной системы.
9. Модели качества информационных систем.
10. Стандарты управления качеством.
11. Надежность информационных систем: основные понятия и определения.
12. Метрики качества.
13. Показатели надежности в соответствии со стандартами.
14. Обеспечение надежности.
15. Методы обеспечения и контроля качества информационных систем.
16. Достоверность информационных систем.
17. Эффективность информационных систем.
18. Безопасность информационных систем.
19. Основные угрозы.
20. Защита от несанкционированного доступа.

## **ЭКЗАМЕН**

### **за 6 семестр**

#### **по МДК. 6.04 Интеллектуальные системы и технологии**

#### **Вопросы для сдачи экзамена**

1. Виды интеллектуальных систем и области их применения.
2. Основные модели интеллектуальных систем
3. Архитектура интеллектуальных информационных систем.
4. Типовая схема функционирования интеллектуальной системы.
5. Примеры интеллектуальных систем
6. Понятие модели представления знаний (МПЗ).
7. Основные МПЗ, их особенности и области применения.
8. Понятие вывода на знаниях.
9. Методы представления знаний в базах данных информационных систем.
10. Формальная грамматика как способ представления знаний в продукционной МПЗ.
11. Понятие и форма записи правил продукции.
12. Синтаксические деревья, задачи разбора и вывода.
13. Составные части экспертной системы: база знаний, механизм вывода, механизмы приобретения и объяснения знаний, интеллектуальный интерфейс.
14. Ограничения, присущие экспертным системам.
15. Особенности экспертных систем экономического анализа.
16. Статические и динамические экспертные системы.
17. Организация процесса приобретения и формализации знаний.
18. Эксперт и инженер по знаниям: формы и порядок взаимодействия.
19. Проблемы неопределенности в экспертных системах.
20. Классификация методов обработки неопределенности знаний.
21. Теория субъективных вероятностей.
22. Байесовское оценивание.
23. Теорема Байеса как основа управления неопределенностью.

## **3.2. Информационное обеспечение реализации программы**

### **3.2.1. Печатные издания**

1. Фуфаев Э.В. Разработка и эксплуатация удаленных баз данных: учебник для студ. учреждений сред. проф. образования/ Э.В.Фуфаев, Д.Э. Фуфаев. – 4-е изд., стер. – М.: Издательский центр «Академия», 2016. – 256 с.
2. Боровская Е. В. Основы искусственного интеллекта - М.: Бином. Лаборатория знаний, 2018

### **3.2.2. Электронные издания (электронные ресурсы)**

1. Система федеральных образовательных порталов информационно -коммуникационные технологии в образовании. [Электронный ресурс] – режим доступа: <http://www.ict.edu.ru> (2003-2017)

### **3.2.3. Дополнительные источники**

1. Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы: учебник / В. А. Гвоздева. - М.: ИД "ФОРУМ-ИНФРА-М, 2017.-544 с.
2. Ясницкий Л.Н. Интеллектуальные системы: учебник – М.: Лаборатория знаний, 2016. – 221 с.
3. Стюарт Рассел, Питер Норвиг. Искусственный интеллект. Современный подход. - М.: Вильямс, 2016

МАКЕТ

ОТЧЕТА ТИТУЛЬНОГО ЛИСТА

---

государственное автономное профессиональное образовательное учреждение  
«Сабинский аграрный колледж»

специальность: 09.02.07 – Информационные системы и программирование (Программист)

ОТЧЕТ  
ПО УЧЕБНОЙ ПРАКТИКЕ

ПМ06. УП06.01 Сопровождение информационных систем

студента группы \_\_\_\_\_

**Иванова Ивана Ивановича**

Начало практики:

\_\_\_\_\_  
(Ф.И.О., и подпись руководителя практики)

Окончание практики

\_\_\_\_\_  
(Ф.И.О., и подпись руководителя практики)

М.П.

Б.Сабы, 20\_\_г

## **Требования к оформлению отчета по производственной практике**

Отчет студента - практиканта по производственной практике (далее - отчет) должен соответствовать следующим требованиям:

### **I. Отчет содержит:**

1. Титульный лист;
2. Содержание отчета (с указанием номеров страниц);
3. Титульный лист дневника;
4. Содержание дневника;
5. Введение (история о предприятии (компании, организации) цель, место, перечень основных работ и заданий, выполняемых в процессе практики);
6. Мероприятия по охране труда и технике безопасности (включая: положение, инструкции и нормативные документы);
7. Описание технологического процесса выполненных работ в период производственной практики;
8. Индивидуальное задание (*темы выданные руководителем практики*);
9. Заключение (*вывод по практике*):
  - необходимо описать навыки и умения, приобретенные за время практики;
  - индивидуальные выводы о практической значимости для себя проведенного вида практики;
10. Характеристика-отзыв практиканта с места практики;
11. Аттестационный лист по практике (*заполняется руководителями практики от техникума и от предприятия, подписывается 2-мя руководителями*).

II. Отчёты выполняются на стандартных листах (ф. А-4(210×297 мм) с одной стороны.

III. Схемы, рисунки, таблицы выполняются с использованием компьютерной техники (Microsoft Office Word 2003, 2007, 2010 гг.).

IV. Текст работы следует печатать, соблюдая следующие принципы:

-заголовки печатаются прописными буквами, с абзацным отступом по середине 1,25 мм.

- текст набирается шрифтом Times New Roman кеглем 14, строчным, без выделения, с выравниванием по ширине;
- текст и таблица не переносятся на следующую страницу;
- абзацный отступ должен быть одинаковым и равен по всему тексту 1,25 см;
- строки разделяются полуторным интервалом;
- поля страницы: верхнее и нижнее - 20 мм, левое - 30 мм, правое - 10 мм;
- полужирный шрифт не применяется;
- в таблицах разрешается набор текста шрифтом Times New Roman кеглем 12;
- разрешается использовать компьютерные возможности акцентирования внимания на определенных формулах, теоремах.

Опечатки, описки и графические неточности, обнаруженные в процессе оформления отчета, допускается исправлять подчисткой или закрашиванием белой краской и нанесением на том же месте исправленного текста (графиков) чернилами черного цвета. Число исправлений на одном листе (странице) не должно превышать 10% от общего объема информации на листе. Повреждения листов тестовых документов, помарки и следы не полностью удаленного прежнего текста (графики) не допускаются.

V. Отчет должен содержать не более 20 листов печатного текста, включая схемы, чертежи, рисунки, скрин фото.

VI. Готовый отчет, собранный в папку, сдаётся в течение 2 дней после окончания практики руководителю практики от техникума.

**ПРИМЕР МАКЕТА ОТЧЕТА ТИТУЛЬНОГО ЛИСТА ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ (приведен ниже).**

МАКЕТ  
ОТЧЕТА ТИТУЛЬНОГО ЛИСТА

---

государственное автономное профессиональное образовательное  
учреждение «Сабинский аграрный колледж»

специальность: 09.02.07 – Информационные системы и программирование

ОТЧЕТ  
ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

ПМ06. УП06.01 Сопровождение информационных систем

(наименование профессионального модуля)

студента группы \_\_\_\_\_

**Иванова Ивана Ивановича**

---

(наименование базового предприятия)

Начало практики:

---

(Ф.И.О., и подпись руководителя практики)

Окончание практики

---

(Ф.И.О., и подпись руководителя практики)

М.П.

МАКЕТ  
ТИТУЛЬНОГО ЛИСТА ДНЕВНИКА

---

государственное автономное профессиональное образовательное  
учреждение «Сабинский аграрный колледж»

специальность: 09.02.07 – Информационные системы и программирование

ДНЕВНИК  
ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

ПМ06. УП06.01 Сопровождение информационных систем  
(наименование профессионального модуля)

студента группы \_\_\_\_\_

**Иванова Ивана Ивановича**

---

(наименование базового предприятия)

Начало практики:

---

(Ф.И.О., и подпись руководителя практики)

Окончание практики

---

(Ф.И.О., и подпись руководителя практики)

М.П.

Б.Сабы, 20\_\_ г

## Содержание дневника

(Количество страниц зависит от продолжительности практики)

| Дата | Рабочее место<br>Структурного подразделения | Краткое содержание выполненных работ                                           | Отметка Руководителей практики от предприятия и техникума (подпись) |
|------|---------------------------------------------|--------------------------------------------------------------------------------|---------------------------------------------------------------------|
|      |                                             | Инструктаж по технике безопасности и охране труда. Распорядок дня предприятия. |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             | Консультация. Выдача индивидуального задания                                   |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             |                                                                                |                                                                     |
|      |                                             | Оформление отчета                                                              |                                                                     |
|      |                                             | Сдача и защита отчета                                                          | 2 подписи<br><b>Без печати</b>                                      |

\*Дневник работ, выполненных на практике (проверяется руководителем практики от предприятия не реже одного раза в неделю и делается отметка в дневнике).

МАКЕТ  
ХАРАКТЕРИСТИКА-ОТЗЫВ

---

Характеристика - Отзыв  
о прохождении производственной практики студента(-ки)  
государственное автономное профессиональное образовательное  
учреждение  
«Сабинский аграрный колледж»

студент (ка) \_\_\_\_\_  
(ФИО студента)

группы \_\_\_\_\_  
(№ группы, специальность)

проходил (ла) практику с «\_\_» \_\_\_\_\_ 20\_\_ г. по «\_\_» \_\_\_\_\_ 20\_\_ г.  
на

(название предприятия)  
в подразделе-  
нии \_\_\_\_\_  
(название подразделения)

Студент соблюдал/не соблюдал (написать нужное) трудовую дисциплину и правила техники безопасности.

Отмечены нарушения трудовой дисциплины и/или правил техники безопасности:

За время прохождения практики \_\_\_\_\_  
(фамилия имя отчество студента)

показал, что умеет/не умеет (написать нужное) планировать и организовывать собственную деятельность, способен/не способен (написать нужное) налаживать взаимоотношения с другими сотрудниками, имеет/не имеет хороший уровень культуры поведения, умеет/не умеет (написать нужное) работать в команде, овладел высокой/низкой (написать нужное) степенью сформированности умений, знаний и практических навыков в профессиональной деятельности.

В отношении выполнения трудовых заданий проявил (а) себя \_\_\_\_\_

1. рамках дальнейшего обучения и прохождения учебной практики студенту можно порекомендовать \_\_\_\_\_

Оценка(\_\_\_\_\_) \_\_\_\_\_  
(прописью)

Должность руководителя практики от

предприя-  
тия/организации \_\_\_\_\_

\_\_\_\_\_  
(Фамилия Имя Отчество руководителя)

\_\_\_\_\_  
(подпись)

М.П.

**Примечание:** Вам предложен макет, содержащий примерные словесные обороты при написании характеристики-отзыва, которая пишется руководителем от предприятия на бланке предприятия/организации в свободной форме.

Прошито, пронумеровано и скреплено печатью

181 листов

Директор ГАПОУ «Сабинский аграрный колледж»

  
Викхаметов М.