Министерство образования Российской Федерации

# Южно-Уральский государственный университет

Кафедра *“Электронные вычислительные машины”*

**Проект проверен**  **Допустить к защите**

Рецензент Зав. кафедрой

И.Л.Кафтанников

“\_\_ ” 2003г*.* *“ “* 2003г*.*

# Программная система «Аттестации ИТ-специалистов»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к дипломному проекту

ЮУрГУ- Д 220100.2003.452.00.ПЗ

## Консультанты: Руководитель проекта:

*Экономическая часть*

\_*И.В. Хлопотова*\_\_\_\_\_\_\_ *И.Л .Надточий*

“\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2003г. “\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2003г.

### Авторы проекта:

Технологическая часть студенты группы\_ПС- 300

\_*Н.С. Колмакова*\_ \_\_\_\_ *Ю.Е Родионов. О.В Калагин*

“\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_2003г. “\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_2003г.

\_*Раздел БЖД*\_\_\_\_\_\_\_\_\_ Нормоконтроль\_\_\_\_\_

\_*В.Ф. Бухтояров\_\_\_\_\_* *А.Н. Пустыгин*\_\_\_\_\_\_

“\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_2003г. “\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_2003г.

Челябинск

2003г.

ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет *Приборостроительный*

Специальность *220100 “Вычислительные машины,комплексы, системы и сети*”

УТВЕРЖДАЮ:

## Зав. кафедрой

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

“\_\_\_”\_\_\_\_\_\_\_2003г.

**З А Д А Н И Е**

по дипломному проекту (работе) студента группы ПС- 300

**Родионов Юрий Евгеньевич, Калагин Олег Владимирович**

**(**фамилия, имя, отчество)

1. Тема проекта (работы)

**Программная система «Аттестации ИT-специалистов»**

утверждена приказом по институту от “\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_2002 г. № \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. Сроки сдачи студентом законченного проекта (работы)

3. Исходные данные к проекту (работе)

1. Технология проведения аттестации ИТ-специалистов

*2. Среда разработки Delphi 7.0*

*3. Требования к системе:*

*а) Защита информации*

*б) Возможность корректировки процесса аттестации*

*в) Возможность удалённого проведения аттестации*

4. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов)

1. *Анализ существующих систем. Обзор литературы*

*2. Архитектура ПС*

*3. Разработка структур баз данных*

*4. Технология аттестации с использованием данной системы*

1. *Разработка компонент программной системы в среде Delphi 7*
2. *Формирование отчетов*
3. *Решение проблемы защиты информаци*
4. *Методика обработки данных ,полученных в результате аттестации*
5. *Инструкция по эксплуатации*
6. *Экономический раздел*
7. *БЖД*

12*. Заключение*

13*. Руководство системного администратора*

5. Перечень графического материала (с точным указанием обязательных чертежей

1. *Среда разработки (1лист)*

*2. Архитектура программного комплекса (2 лист)*

*3. Структура базы данных (1 лист)*

*4. Технология аттестации (1 лист)*

*5. Технология аттестации с использованием программной системы (1 лист)*

*6. Методика анализа результатов аттестации (1 лист)*

*7. Интерфейс пользователя (1 лист)*

*8. Схемы алгоритмов (3 листа)*

9. *Технология защиты информации*  (*1 лист)*

10.*Экономический раздел (1 лист)*

*итого : 14 листов*

6. Консультанты по проекту (работе), с указанием относящихся к ним разделам проекта:

|  |  |  |  |
| --- | --- | --- | --- |
| Раздел | Консультант | Подпись, дата  Задание выдал Задание принял | |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

7. Дата выдачи задания \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2002г.

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Задание принял к исполнению\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| №№ | Наименование этапов дипломного проекта (работы) | Срок выполнения этапов  проекта (работы) | Отметка о  выполнении |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |
| 4. |  |  |  |
| 5. |  |  |  |
| 6. |  |  |  |
| 7. |  |  |  |
| 8. |  |  |  |
| 9. |  |  |  |
| 10. |  |  |  |
| 11. |  |  |  |

# Зав. кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Студент-дипломник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

## Аннотация

# 

Родионов Ю .Е. Калагин О. В Система аттестации ИТ-

специалистов. – Челябинск: ЮУрГУ, ПС, 2003г, 193с.

Библиография литературы –12 наименований,14листов

чертежей ф.А1.

Проанализирован существующий порядок проведения аттестации ИТ – специалистов, принятый на предприятии ОАО «Троицкая ГРЭС». В существующем порядке были обнаружены существенные недостатки. Предложен новый порядок проведения аттестации ИТ-специалистов с использованием программной системы. Произведён расчёт экономической эффективности внедрения программной системы аттестации ИТ – специалистов. Среда разработки системы Delphi 7.0 Inprise и сервер баз данных Interbase.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

220100.2003.452.00.ПЗ

Разраб.

Родионов Ю.Е.

Калагин О.В.

Провер.

Надточий И.Л

Т. Контр

Колмакова Н.С

Н. Контр.

Пустыгин А.Н..

Утверд.

Мельников А.В

Система аттестации ИТ-специалистов

Лит.

Листов

193

ЮУрГУ Кафедра ЭВМ

# СОДЕРЖАНИЕ

[Введение 6](#_Toc29836162)

##### [1 Типовые решения клиент серверных технологий……………………………9](#_Toc29836163)

[1.1 Архитектуры "файл-сервер" и "клиент-сервер" 9](#_Toc29836164)

[1.2 Создание приложений для работы с базами данных 12](#_Toc29836165)

[1.3 Ядро Borland Database Engine (BDE) 15](#_Toc29836166)

[1.4 Пакет Borland SQL Links for Windows 18](#_Toc29836167)

[1.5 Использование SQL 22](#_Toc29836168)

[1.6 Особенности создания систем клиент/сервер 22](#_Toc29836169)

[1.7 Совместимость / эффективность 23](#_Toc29836170)

[1.8 Перенос данных 24](#_Toc29836171)

[1.9 Применение локального сервера InterBase 27](#_Toc29836172)

[1.10 Локальный сервер InterBase 28](#_Toc29836173)

##### [2 Анализ существующей системы. Обзор литературы………………….…….30](#_Toc29836174)

##### [3 Архитектура программной системы………………………………………….34](#_Toc29836175)

###### [4 Разработка структуры баз данных……………………………………………36](#_Toc29836179)

[4.1.Общая характеристика реляционной модели данных 36](#_Toc29836180)

[4.1.Предварительная структура базы данных, нормализация 43](#_Toc29836181)

[4.1.Окончательная структура базы данных 46](#_Toc29836182)

[5 Технология проведения аттестации с использованием ИТ-системы 49](#_Toc29836183)

[5.1.Технология проведения аттестации на ОАО «Троицкая ГРЭС» и ее недостатки. 49](#_Toc29836184)

[5.2.Технология проведения аттестации с использованием системы. 49](#_Toc29836184)

[6. Разработка компонент программной системы в среде 56](#_Toc29836185)

[7.Формирование отчетов 61](#_Toc29836186)

[8 Решение проблемы защиты информации 62](#_Toc29836189)

[9 Методика обработки данных, полученных в результате аттестации. 69](#_Toc29836190)

[10 Инструкция по экплуатации 70](#_Toc29836191)

[10.1.Компонент «ИТА: Аттестация» 70](#_Toc29836192)

[10.1.1.Минимальные системные требования 70](#_Toc29836193)

[10.1.2 Порядок работы 70](#_Toc29836193)

[10.2. Компонент «ИТА: Дизайнер-эксперт» 72](#_Toc29836195)

[10.2.1.Минимальные системные требования 72](#_Toc29836196)

[10.2.2Порядок работы 72](#_Toc29836197)

[10.3.Компонент ИТА: Руководитель 75](#_Toc29836198)

[10.3.1.Минимальные системные требования 75](#_Toc29836199)

[10.3.2 Порядок работы 76](#_Toc29836200)

[11 Экономический раздел 78](#_Toc29836201)

[11.1.Постановка задачи 78](#_Toc29836203)

[11.2.Построение сетевого графика и расчет его параметров 78](#_Toc29836202)

[11.2.1.Построение сетевого графика 78](#_Toc29836203)

[11.2.2.Расчет временных параметров событий сетевого графика 86](#_Toc29836204)

[11.2.3.Расчет временных параметров работ сетевого графика 88](#_Toc29836205)

[11.3. Технико-экономические показатели 92](#_Toc29836206)

[11.3.1.Учет амортизации 92](#_Toc29836207)

[11.3.2.Расходы на заработную плату исполнителей проекта. 93](#_Toc29836208)

[11.3.3.Затраты на разработку программной системы аттестации ИТ-специалистов 94](#_Toc29836209)

[11.4.Целесообразность использования данного программного продукта 95](#_Toc29836210)

[11.4.1.Анализ качественных преимуществ 95](#_Toc29836211)

11.4.2.Оценка эффективности приминения системы ИТ-тестирования

на предприятии…………………………….…………………………96

[11.5.Вывод 97](#_Toc29836213)

[12 Безопасность жизнедеятельности 98](#_Toc29836214)

[12.1.Планировка помещения 98](#_Toc29836215)

[12.2.Требования охраны труда к помещениям 99](#_Toc29836217)

[12.3.Условия труда на рабочем месте 99](#_Toc29836218)

[12.4.Освещение рабочих мест 100](#_Toc29836219)

[12.4.1.Расчет естественного освещения 101](#_Toc29836220)

[12.4.2.Расчет искусственного освещения 101](#_Toc29836221)

[12.2.Анализ воздействия электромагнитных излучений 106](#_Toc29836222)

[12.3.Анализ электробезопасности на рабочем месте 107](#_Toc29836223)

[12.4.Пожарная профилактика. 108](#_Toc29836224)

[12.5.Анализ шума на рабочем месте 110](#_Toc29836225)

[12.6.Эргономические требования 111](#_Toc29836226)

[13 Заключение 113](#_Toc29836227)

[Список литературы 114](#_Toc29836228)

[14 Руководство системного администратора 115](#_Toc29836229)

##### Приложение 1

###### [Исходный текст программного комплекса:…………………………………..119](#_Toc29836230)

# Введение

В связи с бурным развитием информационных технологий и проникновение их в системы управления технологическими и бизнес процессами предприятий и организаций, всё более значимой становится роль специалиста в области информационных технологий – ИТ - специалиста.

Результатом работы над данным дипломным проектом является анализ существующей технологии проведения аттестации ИТ-специалистов на предприятии ОАО «Троицкая ГРЭС», её модификация и автоматизация.

Аттестация персонала становится общепризнанным способом оценки результативности труда и как следствие важным средством для управления и организационного развития.

Во многих компаниях регулярно, раз в год или в полгода, проводятся аттестационные интервью, или собеседования. Обычно их проводит непосредственный начальник аттестуемого работника. Целью такого собеседования является налаживание обратной связи с сотрудником, чтобы ознакомить его с мнением руководства о его деятельности [3].

Аттестационные собеседования носят исключительно оценочный характер, и проводить их бывает довольно трудно. Деликатность ситуации заключается еще и в том, что результаты этого интервью могут сказаться на зарплате сотрудника. Конкретная цель каждого такого собеседования зависит от того, как интервьюируемый справляется со своими служебными обязанностями.

Аргументом для их проведения является то, что они служит ряду важных целей.

Оценка помогает определить, во-первых, какие работники требуют большей подготовки и, во-вторых, результаты программ набора и обучения персонала [4].

Она помогает установлению и укреплению деловых отношений между подчиненными и руководителями через обсуждение результатов оценки и, кроме того, она побуждает руководителей учиться оказывать необходимую помощь.

Оценка для администрации помогает решить, кому следует повысить зарплату, кого — повысить в должности, а кого — уволить.

Оценка побуждает работников действовать более результативно. Наличие соответствующей программы и гласность результатов ее выполнения развивают инициативу, повышают чувство ответственности и стимулируют стремление работать лучше.

Оценка служит юридической основой переводов, продвижений по службе, наград и увольнений. Она дает материал для разработки вопросников по найму. Оценка позволяет получить дополнительную информацию для того, чтобы определить зарплату и другие вознаграждения работника. Она является естественным поводом для продолжительной беседы между руководителем и подчиненным о проблемах работы, в ходе которой они лучше узнают друг друга.

И, наконец, результаты оценки могут быть использованы при разработке средств отбора персонала, например, тестов [4].

Необходимо учитывать, что в процессе аттестации всегда имеется место для случаев предубежденности и ошибок. Так, например, менеджеры, проводящие интервью, могут ориентироваться на проблемы, несущественные для выполнения конкретных работ, или же придавать слишком малое значение важным слагаемым результативности труда работника.

Основным вопросом при этом является: "Что следует оценивать?"

Показатели, по которым оцениваются работники, называют критериями оценки. Сюда относятся, в частности, качество выполняемой работы, ее количество и ценностная оценка результатов. Важным здесь является то, что оцениваться должны не личности, а результативность работы.

Для такой оценки необходимо довольно большое количество критериев. Их отбор — непростая задача. Предпочтения зависят от конкретных целей бизнеса и сложившейся корпоративной культуры. Если задача — повышение результативности труда на рабочих местах, то критерии должны непосредственно относиться к результативности труда. Если для этой или будущих работ необходимы навыки общения и личные качества — необходимо делать упор именно на них.

# 1 Типовые решения клиент серверных технологий

## 1.1 Архитектуры "файл-сервер" и "клиент-сервер"

Базы данных на персональных компьютерах развивались по направлению от настольных (desktop), или локальных приложений, когда реально с БД могло работать одно приложение, до систем коллективного доступа к БД.

Локальное приложение устанавливаются на единичном персональном компьютере; там же располагаются и БД, с которой работает данное приложение. Однако необходимость коллективной работы с одной и той же БД влечет за собой перенос БД на сетевой сервер. Приложение, работающее с БД, располагается также на сервере. Менее характерным стал другой способ, заключающийся в хранении приложения, обращающегося к БД, на конкретном компьютере пользователей ("клиентов"). Были выпущены новые версии локальных СУБД, которые позволяли создавать приложения, одновременно работающие с одной БД на файловом сервере. Основной проблемой стала явная или неявная обработка транзакций и неизбежно встающая при коллективном доступе проблема обеспечения смысловой и ссылочной целостности БД при одновременном изменении одних и тех же данных [3].

В ходе эксплуатации таких систем были выявлены общие недостатки файл серверного подхода при обеспечении многопользовательского доступа к БД. Они состоят в следующем:

* вся тяжесть вычислительной нагрузки при доступе к БД ложится на приложение клиента, что является следствием принципа обработки информации в системах "файл-сервер": при выдаче запроса на выборку информации из таблицы *вся* таблица БД копируется на клиентское место, и выборка осуществляется на клиентском месте;
* локальные СУБД используют так называемый "навигационный подход", ориентированный на работу с отдельными записями;
* не оптимально расходуются ресурсы клиентского компьютера и сети: например, если в результате запроса мы должны получить 2 записи из таблицы объемом 10 000 записей, все 10 000 записей будут скопированы с файл-сервера на клиентский компьютер; в результате возрастает сетевой трафик и увеличиваются требования к аппаратным мощностям пользовательского компьютера. Заметим, что потребности в постоянном увеличении вычислительных мощностей клиентского компьютера обусловливаются не только развитием программного обеспечения как такового, но и возрастанием обрабатываемых объемов информации;
* в БД на файл-сервере гораздо проще вносить изменения в отдельные таблицы, минуя приложения, непосредственно из инструментальных средств (например, из утилиты *Database Desktop* фирмы Borland для файлов Paradox или dBase); подобная возможность облегчается тем обстоятельством, что, фактически, у локальных СУБД база данных понятие более логическое, чем физическое, поскольку под БД понимается набор отдельных таблиц, сосуществующих в едином каталоге на диске. Все это позволяет говорить о низком уровне безопасности - как с точки зрения хищения и нанесения вреда, так и с точки зрения внесение ошибочных изменений;
* бизнес правила в системах "файл-сервер" реализуются в приложении, что позволяет в разных приложениях, работающих с одной БД, проектировать взаимоисключающие бизнес правила; смысловая целостность информации при этом может нарушаться;
* недостаточно развитый аппарат транзакций для локальных СУБД служит потенциальным источником ошибок как с точки зрения одновременного внесения изменений в одну и ту же запись, так и с точки зрения отката результатов серии объединенных по смыслу в единое целое операций над БД, когда некоторые из них завершились успешно, а некоторые - нет; это может нарушать ссылочную и смысловую целостность БД.

Приведенные недостатки решаются при переводе приложений из архитектуры *"файл-сервер "* в архитектуру *"клиент-сервер ",* которая знаменует собой следующий этап в развитии СУБД. Характерной особенностью архитектуры "клиент-сервер" является перенос вычислительной нагрузки на *сервер БД (SQL-сервер)* и максимальная разгрузка приложения клиента от вычислительной работы, а также существенное укрепление безопасности данных - как от злонамеренных, так и просто ошибочных изменений.

БД в этом случае помещается на сетевом сервере, как и в архитектуре "файл-сервер", однако прямого доступа к БД из приложений не происходит. Функции прямого обращения к БД осуществляет специальная управляющая программа сервер БД (SQL-сервер), поставляемая разработчиком СУБД.

Взаимодействие сервера БД и приложения-клиента происходит следующим образом: клиент формирует SQL-запрос и отсылает его серверу. Сервер, приняв запрос, выполняет его и результат возвращает клиенту. В клиентском приложении в основном осуществляется интерпретация полученных от сервера данных, реализация интерфейса с пользователем и ввод данных, а также реализация части бизнес правил.

Преимущества архитектуры "клиент-сервер":

• большинство вычислительных процессов происходит на сервере; таким образом, снижаются требования к вычислительным мощностям компьютера клиента;

• снижается сетевой трафик за счет посылки сервером клиенту только тех данных, которые он запрашивал; например, если необходимо сделать из таблицы объемом 10 000 записей выборку, результатом которой будут всего 2 записи, сервер выполнит запрос и перешлет клиенту НД из 2 записей;

• упрощается наращивание вычислительных мощностей в условиях развития программного обеспечения и возрастания объемов обрабатываемых данных: проще и чаще дешевле усилить мощности на сетевом сервере или полностью заменить сервер на более мощный, нежели наращивать мощности или полностью заменять 100-500 клиентских компьютеров;

• БД на сервере представляет собой, как правило, единый файл, в котором содержатся таблицы БД, ограничения целостности и другие компоненты БД. Взломать такую БД, даже при наличии умысла, тяжело; значительно увеличивается защищенность БД от ввода неправильных значений, поскольку сервер БД проводит автоматическую проверку соответствия вводимых значений наложенным ограничениям и автоматически выполняет необходимые бизнес правила; кроме того, сервер отслеживает уровни доступа для каждого пользователя и блокирует осуществление попыток выполнения неразрешенных для пользователя действий: например, изменения или просмотр таблиц; все это позволяет говорить о значительно более высоком уровне обеспечения безопасности БД иссылочной и смысловой целостности информации;

• сервер реализует управление транзакциями и предотвращает попытки одновременного изменения одних и тех же данных; различные уровни изоляции транзакций позволяют определить поведение сервера при возникновении ситуаций одновременного изменения данных;

• безопасность системы возрастает за счет переноса большей части бизнес правил на сервер; падает удельный вес противоречащих друг другу бизнес правил в клиентских приложениях, выполняющих разные действия над БД; определить такие противоречивые бизнес правила в приложениях клиента все еще можно, однако намного труднее ихвыполнить ввиду автоматического отслеживания сервером БД правильности данных.

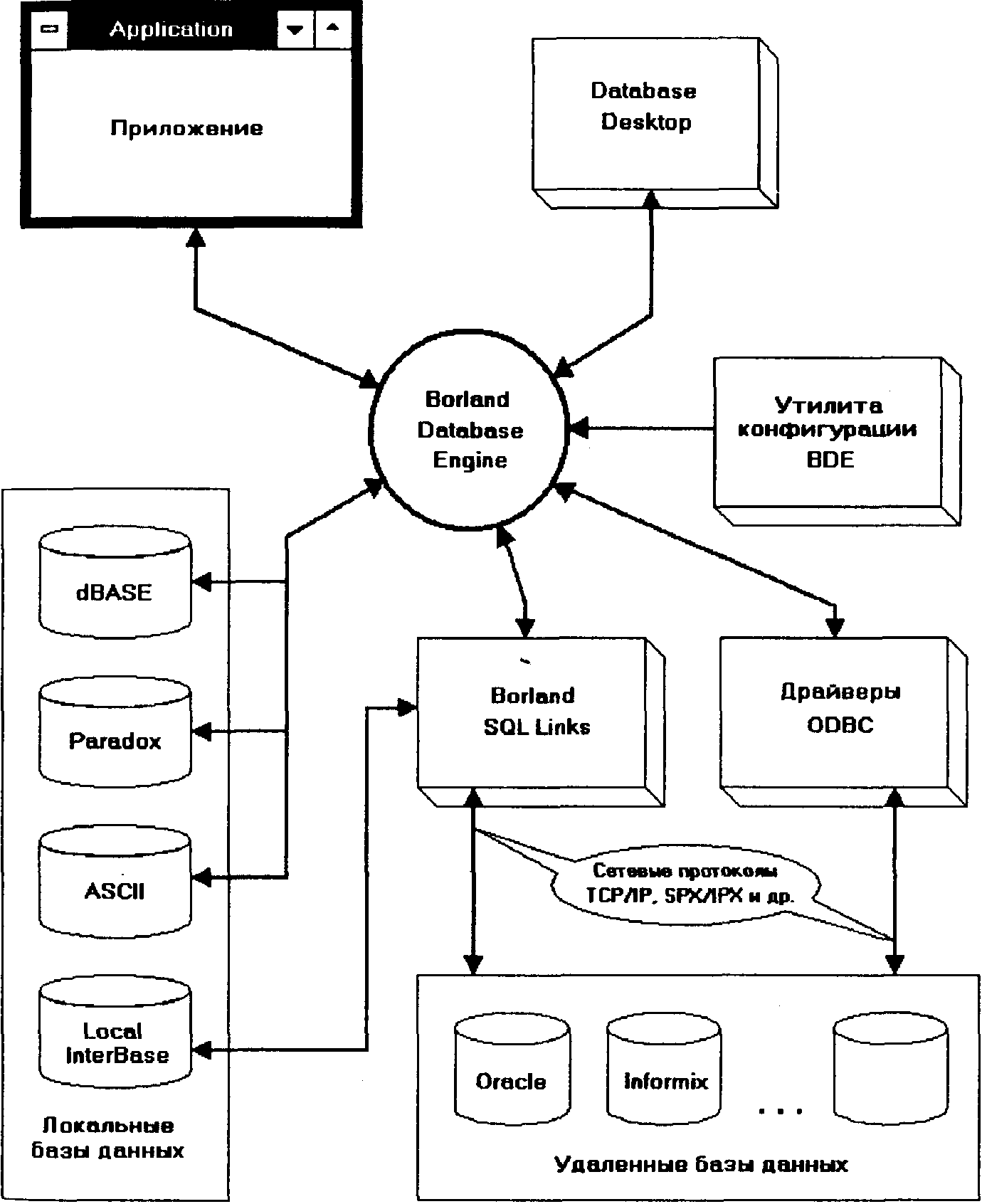
Для реализации архитектуры применяют так называемые "промышленные” СУБД, такие как Borland InterBase, Oracle, Informix, Sybase, DB2, MS SQL Server.

## 1.2 Создание приложений для работы с базами данных

Delphi 7.0 представляет собой уникальную систему разработки, в которой технология высокопроизводительной оптимизирующей компиляции сочетается с визуальными средствами разработки и масштабируемым процессором баз данных. Это позволяет создавать эффективные приложения Windows, работающие с базами данных, в том числе и программы для систем клиент/сервер. Для создания таких приложений в Delphi 7.0 используется объектно-ориентированный подход, базирующийся на применении различных компонентов (визуальных и не визуальных), что обеспечивает неограниченную расширяемость и маcштабируемость. Delphi 7.0 позволяет разработчику быстро создавать и свободно распространять приложения с архитектурой клиент/сервер, работающие существенно быстрее и надежнее предыдущего поколения программных продуктов, которые строились при помощи систем разработки, основанных на интерпретируемом коде [9].

Большим преимуществом приложений, разрабатываемых в среде Delphi 7.0,- стала доступность использования как реляционного, так и навигационного прог­раммирования при работе с данными. Такую возможность приложениям Delphi 7.0 предоставляет ядро процессора баз данных Borland Database Engine (BDE). Использование реляционных методов позволяет манипулировать большими вы­борками информации и легко проводить групповые операции. Навигационные методы дают приложению преимущества быстрого доступа к отдельным полям и записям таблиц баз данных.

Структурная схема организации доступа приложения к различным базам дан­ных отражена на рисунке 1.1. В наиболее общем случае работа с данными в Delphi 7.0 осуществляется через BDE, который обеспечивает непосредственную связь с локальными базами дан­ных и используется при организации доступа к удаленным серверам. В основе BDE лежит технология Integrated Database API (IDAPI), уже известная прог­раммистам, которые работают с СУБД фирмы Borland. Через BDE и драйверы Borland SQL Links приложение может связываться с SQL-серверами. В то же время, BDE поддерживает и интерфейс Open DataBase Connectivity (ODBC), что позволяет получить доступ не только к любому удаленному серверу баз данных, для которого имеется драйвер ODBC, но и к любому источнику структурированных данных.



**Рис. 1.1.** Механизм организации доступа приложения к базам данных

Примечание

ODBC — интерфейс для свободного доступа к данным в гетерогенной среде реляционных и не реляционных баз данных. Основываясь на базовом интерфейсе SQL — Call Level Interface, ODBC обеспечивает открытый до ступ к большинству данных, расположенных на персональном компьютере миникомпьютере и к базам данных больших ЭВМ, позволяя разработчикам иметь одновременный доступ к базам данных. Стандарт ODBC полностью поддерживает технологию клиент/сервер.

В состав стандартной поставки Delphi 7.0 включен локальный сервер Interbase, который позволяет проводить в Delphi 7.0 автономную разработку приложений с поддержкой SQL, готовых к переносу в среду клиент/сервер. Он представляет собой облегченный вариант Interbase Workgroup Server 4.0.

## 1.3 Ядро Borland Database Engine (BDE)

Как уже отмечалось, использование Delphi 7.0 позволяет разработчику создавать самые разнообразные приложения для работы с базами данных. Среди них могут быть как простейшие программки, открывающие два-три поля, так и мощные приложения, предназначенные для работы в системах клиент/сервер. Такая универсальность достигается за счет использования ядра BDE. В основе BDE лежит технология IDAPI, которая включает IDAPI-инфраструктуру и обработчик запросов [2].

Использование BDE позволяет приложению осуществлять доступ к данным не только локальных (Paradox и dBase), но и удаленных баз данных, располо­женных на SQL-серверах (Interbase, Sybase, Oracle, Informix, MS SQL Server), а также в любых форматах, доступных через драйверы ODBC (см. рис. 1.2). BDE поддерживает многопользовательский доступ к гетерогенным базам дан­ных, связанные запросы к нескольким разнотипным базам данных одновре­менно, прямой перенос данных из одного формата в другой. Программисты могут обращаться к функциям BDE с помощью языков программирования Borland C++, Borland Pascal, Visual C++, а также любых других компиляторов С и C++ для Windows.

Архитектура BDEUDAPI основана на драйверах. Для каждого источника дан­ных существует свой драйвер, который поддерживает не только последнюю версию источника, но и все предыдущие версии. Именно через такие драйверы осуществляется связывание и все обращения к данным. BDE поддерживает два класса драйверов. К первому классу относятся драйверы, обслуживающие SQL-серверы, причем каждый из этих серверов может использовать собственный диалект SQL. Во второй класс входят драйверы для локальных баз данных.

Архитектура BDEMDAPI является объектно-ориентированной, поэтому ее инфраструктура легко расширяется и обобщается. В комплекте BDE содержатся более пятидесяти языковых драйверов, которые используются всеми драйве­рами доступа к данным и всеми общими обработчиками и сортировщиками запросов. Инфраструктура BDEUDAPI предоставляет обширный набор инстру­ментов, которые могут использоваться всеми драйверами.

*Диспетчер памяти* предоставляет дополнительные возможности по управлению памятью. В отладочном режиме этот модуль помечает, трассирует и разрешает все попытки использовать память.

*Диспетчер буфера* основывается на использовании приоритетов и поддерживает режим совместного использования буферов различными драйверами.

*Сортировщик* автоматически оптимизирует процесс использования доступной памяти и вызывается через соответствующую функцию BDE. Он использует установленный языковый драйвер для работы с различными наборами сим­волов.

*Кэш для данных BLOB* позволяет производить чтение/запись произвольного места в бинарном объекте, при переполнении содержимое кэша автоматически записывается в разделяемый файл. Одновременно может быть открыто любое количество BLOB.

*Генератор SQL* транслирует запрос в формате QBE в эквивалентный запрос SQL, если он предназначен SQL-серверу.

*Модуль реструктурирования* поддерживает сложные изменения структур таблиц, в том числе добавление новых полей, удаление полей, изменение их имен, изменение свойств поля (тип, размер), ссылочной целостности таблицы и т. д. Также этот модуль создает новую таблицу, трансформирует данные и копирует их в нее.

*Функции пакетной обработки* включают копирование данных из одного формата в другой, переименование таблиц и т. д.

*Модуль Xlate* оптимизирует процесс преобразования форматов данных.

*Модуль таблиц в памяти* обеспечивает виртуальную память, ориентированную на таблицы. Он поддерживает курсоры приложений, как и любые другие курсоры IDAPI. Работа модуля тесно связана с работой диспетчера буфера.

*Модуль поддержки SQL-драйверов* используется при создании любых SQL-драйверов.

*Конфигурационный диспетчер* участвует в настройке среды BDE при начальной загрузке.

*Системный диспетчер* управляет всеми ресурсами системного уровня. Он отвечает за загрузку драйверов, отслеживание открытых баз данных, курсоров и контекста каждого приложения.

*Общий обработчик запросов* поддерживает и SQL и QBE. Он построен с использованием технологии курсоров BDE и поэтому может работать с любым источником данных. Если запрос может быть выполнен напрямую, то он сразу передается серверу. Запрос QBE предварительно транслируется в SQL.

Технология Idapter является составной частью BDE и предназначена для организации доступа к базам данных, используя стандартный программный интерфейс драйверов Borland SQL Links. Idapter транслирует вызовы функций интерфейса IDAPI в вызовы стандартных методов интерфейса ODBC, что позволяет использовать практически любой драйвер стандарта ODBC в режиме драйвера IDAPI. При этом могут использоваться любые функции интерфейса IDAPI. Технология Idapter существенно увеличивает число доступных

через BDE форматов данных. Поставляется совместно с IDAPI, как отдельная динамическая библиотека.

## 1.4 Пакет Borland SQL Links for Windows

Пакет Borland SQL Links for Windows предназначен для использования теми приложениями, работающими с BDE, которым необходим доступ и к локаль­ным базам данных и к удаленным SQL-серверам. После инсталляции соответ­ствующего драйвера SQL Links и создания необходимого псевдонима приложение получает доступ к базам данных необходимого SQL-сервера. Место SQL Links в механизме доступа к базам данных из приложений Delphi 7.0 показано на рисунке 4.7.

Установленный драйвер выполняет работу по соединению с нужным SQL-сер­вером, переводу запросов приложения в соответствующий диалект SQL и пере­даче запроса базе данных. Ответ базы данных снова преобразуется им к виду, воспринимаемому приложением.

Для установки параметров процесса связывания приложения с требуемым сер­вером SQL используется утилита конфигурации BDE. Естественно, что перед выполнением такой настройки необходимо иметь инсталлированный SQL Links с установленным драйвером для нужного сервера. Все настраиваемые пара­метры сосредоточены на странице Drivers утилиты конфигурации (см. рисунок 1.2).



**SQL Links** транслирует

Ответ клиента и посылает его BDE

**Сервер SQL** проверяет правильностьзапроса. выполняет *его* и отправляет приложению-клиенту

**Рис. 1.2.**Использование драйвера SQL Links приложением

Первым делом необходимо выбрать нужный драйвер из списка имен драйверов в левой части панели. В правой части появится список всех пара­метров драйвера и их текущих значений. При необходимости, можно переоп­ределить значения параметров, заданные по умолчанию и сохранить изменения. Эффект от сделанных установок проявится только при следующем запуске приложения.

Ниже будут рассмотрены общие для всех драйверов SQL Links параметры.

Дополнительную информацию о специфических параметрах каждого драйвера можно получить, выбрав соответствующее имя в списке утилиты конфигурации

BDE и нажав кнопку Help.

DLL — определяет имя динамической библиотеки SQL Links для драйвера.

Driver Flags — внутренний флаг, изменять этот параметр не рекомендуется.

LangDriver — задает языковый драйвер, который применяется для манипулирования данными, полученными при помощи драйвера SQL Links. Поле ввода этого параметра содержит список всех доступных языковых драй­веров. Если выбранный языковый драйвер определен также и в псевдониме приложения, то он используется для управления любыми данными, получен­ными от сервера. Языковый драйвер используется для преобразования данных, если приложение и сервер используют разные кодовые страницы. В противном случае все национальные символы превратятся в абракадабру. Если необ­ходимый языковый драйвер отсутствует, можно использовать параметр SQLQRYMODE для отмены преобразования данных по правилам базы данных.

Open Mode — определяет режим, в котором SQL Links открывает доступ к базе данных. Возможные значения: "Чтение\3апись" и "Только для чтения". Режим "Только для чтения" не работает при использовании прямых запросов.

Schema Cache Size — задает число таблиц базы данных, чья структурная информация кэшируется. Возможные значения: 0—32 (по умолчанию 8).

Schema Cache Time — задает время нахождения структурной информации о таблицах в кэше. Возможные значения: -1 (по умолчанию) — до закрытия базы данных; 0 — информация в кэше не размещается; 1-214748347 — число секунд.

Server Name — содержит имя целевого сервера. Для серверов Interbase обязательно надо задавать маршрут, как это показано в примере:

servername/usr/gds/directoryname/databasename/gdb.

SQLPASSTHRUMODE — определяет режим использования прямых и ло­кальных запросов при соединении через один псевдоним: NOT SHARED запрещает одновременное использование прямых и локальных запросов;

SHARED AUTOCOMMIT разрешает совместное использование, причем прямые запросы ведут себя в соответствии с правилами для локальных запросов, что означает режим автоматической фиксации транзакций, если только не установлено явное управление транзакциями или режим группового выполнения; SHARED NOAUTOCOMMIT разрешает совмест­ное использование, но режим неявной фиксации транзакций не исполь­зуется. Поведение прямых запросов зависит от типа сервера.

Предопределенное значение для серверов Informix — SHARED AUTOCOMMIT, для остальных серверов SQL — NOT SHARED. Режимы SHARED AUTOCOMMIT и SHARED NOAUTOCOMMIT не под­держиваются некоторыми предложениями прямых запросов, в этом слу­чае следует использовать явное управление транзакциями через функции приложения.

SQLQRYMODE — определяет режим выполнения запросов, возможные значения приведены в таблице.

**Таблица 1.1** Режимы выполнения запросов*.*

|  |  |  |
| --- | --- | --- |
| Значение | Режим | Комментарий |
| NULL | Сервер-локальный | Запрос направляется сначала на сервер, затем, при невозможности выполнения, выполняется локально |
| SERVER | Только сервер | Запрос направляется только на сервер, в случае невозможности выполнения, отменяется |
| LOCAL | Только локальный | Запрос выполняется исключительно локально |

Значение по умолчанию — NULL. На получаемый результат запросов может влиять установленный языковый драйвер, если локальные базы данных и базы SQL имеют различные кодовые страницы (см. выше). Для устранения подобных ошибок необходимо установить для параметра значение SERVER, блокируя таким образом, выполнение запросов в ло­кальных базах данных.

Type определяет тип используемого сервера. Значение SERVER определяет использование SQL-сервера. Значение FILE определяет использование обычных серверов, основанных на файловой системе.

User Name — задает имя пользователя для доступа к серверу.

Version — версия драйвера SQL Links.

Для доступа к серверу SQL необходимо иметь соответствующий псевдоним. Базовый псевдоним для каждого используемого драйвера SQL Links создается автоматически при первом изменении параметров драйвера после инсталляции.

## 1.5 Использование SQL

В этом разделе будут рассмотрены различные аспекты применения запросов SQL в приложениях, использующих базы данных. Для реализации запросов в Delphi 7.0 существует специальный компонент — TQuery, расположенный на странице Data Access Палитры компонентов. Он имеет много общих свойств с TTable и тоже используется для открытия наборов данных. В то же время TQuery обладает рядом свойств и методов, которые позволяют использовать все преимущества запросов SQL для работы с данными. Так, применение TQuery дает возможность работать с данными нескольких таблиц в одном запросе, отбирать данные сразу по нескольким критериям. Однако следует помнить, что использование SQL ведет к некоторому усложнению програм­много кода приложения. Кроме того, создание эффективного запроса — дело далеко не простое и требует наличия определенного опыта в этой области. Запросы SQL бывают статическими и динамическими. Статические запросы полностью создаются при отладке приложения, а динамические могут изменять свои параметры при выполнении приложения.

Приложения Delphi 7.0 при помощи механизма запросов SQL могут использовать данные:

• таблиц Paradox и dBase, используя синтаксис локального SQL;

• локального сервера Interbase, синтаксис языка поддерживается полностью;

• удаленных серверов SQL через драйверы SQL Links.

## 1.6 Особенности создания систем клиент/сервер

Возможность создания приложений для работы в составе систем клиент/сервер, бесспорно, стала большим преимуществом Delphi 7.0. Инструментарий для разра­ботки таких приложений интегрирован в составе среды разработчика. Приложения Delphi 7.0, функционирующие на станции-клиенте, используя возмож­ности BDE и драйверов SQL Links и ODBC, могут получать доступ к данным удаленных SQL-серверов. В качестве серверов могут быть использованы Informix, Interbase, Microsoft SQL Server, Oracle, Sybase. Кроме этого, через BDE и установленный драйвер ODBC возможен доступ к таким базам, как DB2, Btrieve, Microsoft Access и другим, для которых существует соответствующий драйвер ODBC.

Приложение, функционирующее на станции-клиенте, обычно создается отдельно и для уже работающих серверов баз данных. Поэтому, для создания рабо­тоспособной системы клиент/сервер необходимо решить ряд проблем связывания рабочих станций, совместимости данных при работе одного приложения-клиента с различными типами серверов, оптимизации производительности.

## 1.7 Совместимость / эффективность

Как известно, при создании приложений для систем клиент/сервер всегда приходится решать проблему выбора между универсальностью и производительностью. С одной стороны, чем большее количество типов сер­веров поддерживается приложением, тем лучше. Но в этом случае значительно понижается производительность. Впрочем, способ решения этой проблемы зависит от предназначения создаваемого приложения. Иногда можно пожерт­вовать совместимостью, а иногда оказывается не так уж и важна произво­дительность.

Совместимость по данным в значительной степени зависит от используемых приложением компонентов. При применении ТТаblе проблем практически не возникает, а вот при использовании TQuery приходится накладывать ограничения на синтаксис предложений SQL в зависимости от диалекта SQL, используемого сервером.

Производительность приложения-клиента может быть повышена при исполь­зовании хранимых процедур сервера, однако это приводит к дополнительной специализации программы.

В зависимости от типа оборудования, на котором работает приложение, может возникнуть необходимость в поддержке нескольких коммуникационных протоколов. Эта проблема решается инсталляцией соответствующего программного обеспечения на станции-клиенте и сервере, однако, это решение должно быть предусмотрено еще на этапе проектирования системы клиент/сервер. Информацию об инсталлированном протоколе необходимо включить в драйве SQL Links.

В дальнейшем реализация архитектуры "клиент-сервер" будет рассматриваться для сервера *Borland Interbase.* Объяснить такой выбор нетрудно. Во-первых, Interbase - "родной" сервер для Delphi 7.0 (поэтому для доступа к нему не нужно устанавливать дополнительных драйверов *SQL Links,* что необходимо при работе из приложений, написанных на Delphi 7.0, с Oracle, Sybase и другими СУБД). Во вторых, в поставку Delphi входит локальный (однопользовательский, на 2 одновременных подключения) сервер *Local Borland Interbase.* Доступен также иInterbase для Windows 95 на 4 пользователя.

Локальный Interbase может использоваться для отладочных целей. После того, как приложение отлажено на локальной версии SQL-сервера, происходит масштабирование приложения *(upsizing).* БД переносится на сетевой сервер, а изменения в клиентских приложениях при этом минимальны - необходимо изменить псевдоним БД и, может быть, скорректировать некоторые параметры соединения приложения с сервером.

## 1.8 Перенос данных

При переносе приложений, ранее разработанных для применения в архитектуре "файл-сервер", требуется не только частично или полностью переписывать приложения клиентов, но и преобразовывать локальную БД в серверную. Для этого под управлением серверной СУБД (например, Interbase) создают БД на сервере, куда затем "перекачивают" данные из локальных СУБД реализованных, например, с помощью Paradox. Основная проблема, встающая в этом случае - несовместимость некоторых форматов данных или их отсутствие. Например, Interbase не поддерживает поля типа Boolean (Logical), и их необходимо реализовывать при помощи столбцов типа CHAR(1); Interbase не поддерживает автоинкрементные поля Paradox - для обеспечения уникальности значений в числовых полях в БД Interbase используют генераторы и т.д. При возникновении подобных проблем следует изучить вопросы совместимости типов данных локальной СУБД и выбранной серверной СУБД.

Преобразование делится на два этапа: •

• модернизация баз данных до уровня сервера;

• преобразование приложений в приложения-клиенты.

Преобразование позволяет поднять систему приложение-база данных на качес­твенно новый уровень, так как архитектура клиент/сервер имеет ряд важных преимуществ. Среди них многопользовательский доступ, возможность работы с множествами, а не с отдельными записями, использование доступа ко всем данным, а не к отдельным таблицам.

Преобразование базы данных в сервер содержит ряд этапов.

1. Создание метаданных, основанных на структуре базы данных.

2. Перенос данных на сервер.

3. Разделение данных по типам.

4. Создание паролей и интеграция данных.

5. Контроль транзакций.

6. Управление доступом к данным.

7. Проверка данных.

Delphi обеспечивает два способа преобразования баз данных:

• использование возможностей утилиты Database Desktop для преобразования таблиц в формат SQL;

• использование при создании приложения компонента TBatchMove.

Оба эти способа копируют структуру данных и переносят сами данные в формат SQL-сервера. В новую структуру метаданных могут быть добавлены новые элементы: индексы, хранимые процедуры, триггеры и т. д. Для неко­торых типов серверов более эффективным может стать ручное создание струк­туры метаданных с дальнейшим переносом данных, как предлагается выше.

Приложения, созданные для работы с локальными базами данных, могут быть приспособлены для систем клиент/сервер только путем внесения ряда исправ­лений в исходный код. В простейшем случае, при полном совпадении структуры преобразованных и исходных данных необходимо лишь изменить параметры свойств DatabaseName используемых компонентов TTable и TQuery и добавить в приложение компонент TDatabase. Однако такое бывает не часто.

Локальные приложения используют, в основном, компоненты TTable, реже TQuery. Однако, в зависимости от особенностей конкретного приложения, более эффективным может стать переход на использование TQuery, особенно при использовании таблиц с большим числом записей.

Если приложение использует арифметические или логические функции, то же­лательно заменить их соответствующими хранимыми процедурами сервера. Этим достигается повышение производительности, так как сервер обладает большей мощностью, чем станция-клиент, и разгружаются сетевые каналы связи.

Часто бывает необходимым включить в исходный код приложения метода поддержки транзакций, так как в большинстве однопользовательских систем они не используются.

Также, нередко возникает необходимость в создании кода для регистрации пользователя на сервере.

## 1.9 Применение локального сервера InterBase

Локальный Interbase Server является версией сервера Borland Interbase 6.0 для

Windows и содержит полный набор функций для локального однопользовательского применения. При разработке приложений клиент/сервер локальный Interbase Server может использоваться в качестве модели сервера или для преобразования баз данных в серверы SQL (см. "Перенос данных"), кроме этого, он может применяться в качестве процессора базы данных в работ локальных приложений. Его использование позволит разработчику повысит надежность разрабатываемого приложения и избежать возможной потери данных при тестировании "сырых" приложений.

Если база данных, для работы с которой предназначено разрабатываемое приложение, уже существует, то Interbase Server может быть использован для

решения ряда других вопросов:

1. Для поиска и преобразования нестандартного синтаксиса запросов SQL и неиспользуемых типов данных;

2. Для создания приложений, использующих Interbase в качестве сервера;

3. Применение Windows ISQL возможно для создания базы Interbase, предложения SQL которой могут быть использованы в базе сервера приложения

4. В качестве проверочной базы перед последующим соединением приложения с настоящим сервером. Для этого необходимо только изменить псевдоним, используемый компонентом TDatabase приложения.

Если настоящая база данных еще не существует, то Interbase Server может использоваться для создания прототипа используемых данных, на котором будет проверяться работоспособность приложения.

Если приложение разрабатывается для уже существующей базы функционирующей на сервере Interbase:

• перед проверкой работоспособности приложения на реальных данных может использоваться утилита локального сервера Interbase для создания резервных копий данных;

• желательно перенести на локальный сервер небольшую, но представительную выборку данных и отлаживать работу приложения на ней.

При проведении преобразования локальной базы данных до уровня систем клиент/сервер, локальный сервер Interbase используется в качестве промежуточного сервера, на котором проверяется новая структура данных. После про верки данные переносятся на целевой сервер.

## 1.10 Локальный сервер InterBase

Локальный сервер Interbase представляет собой версию Borland Interbase Workgroup Server 6.0 для одного пользователя, работающую под управлением Windows. Этот сервер содержит в своем составе утилиты Windows ISQL и Server Manager, которые также могут использоваться с полномасштабным сер­вером Interbase. Доступ к базам данных, управляемым локальным сервером Interbase, осущес­твляется через Windows ISQL или само приложение.

Сервер может использоваться в среде разработчика Delphi для решения ряда задач создания приложений и наборов данных:

1. Под управлением локального сервера Interbase могут создаваться полно­ценные базы данных.

2. Проводится тестирование приложений клиент/сервер, причем Interbase используется в качестве модели реального сервера.

3. При преобразовании локальных СУБД до уровня возможностей сервера SQL, локальный Interbase может использоваться как промежуточная сту­пень (см. раздел "Особенности построения приложений для архитектуры клиент/сервер").

Доступ к данным сервера осуществляется через BDE и SQL Links. Сервер поддерживает стандарт языка ANSI SQL 92, доступ к предложениям которого осуществляется через Windows ISQL или приложение. Некоторые расширения SQL, поддерживаемые сервером, конфликтуют с диалектом SQL3, среди них хранимые процедуры, триггеры и данные типа BLOB.

Используемые локальным сервером Interbase форматы данных представлены в таблице.

**Таблица 1.2** Основные форматы данных Interbase.

|  |  |
| --- | --- |
| **Формат данных** | **Типы данных** |
| Числовой | Smallint, Integer, Float, Double Precision, Numeric, Decimal |
| Символьный | Char, VarChar |
| Календарный | Data |
| BLOB | BLOB |

Естественно, что локальный Interbase Server уступает полномасштабному варианту. Среди важнейших отличий можно отметить отсутствие следующих возможностей:

• BLOB фильтров;

• массивов данных;

• отслеживания событий;

• функций, определяемых пользователем;

• ограничена поддержка записи через журнал (WAL).

В локальном сервере реализована система защиты данных. Предусматривается возможность регистрации пользователя с использованием паролей.

## Анализ существующей системы. Обзор литературы

Поиск информации о существующих системах аттестации ИТ-специалистов привёл к выводу, что доминирующая часть таких систем работает через Web-интерфейс, вот примеры таких систем:

1. Сертификация Microsoft Office User Specialist (MOUS) - всемирно признанный стандарт для оценки навыков работы с бизнес приложениями Microsoft Office [8]

http://www.specialist.ru/MOUS/

2. Сервера - профессиональной оценки знаний в области информационных технологий

http://tests.specialist.ru/

3. **А+ сертификация** – программа тестирования, предоставленная компьютерной ассоциацией CompTIA, которая подтверждает компетенцию начинающих специалистов по обслуживанию компьютерной техники с опытом работы 6 месяцев или прошедших подготовку на соответствующих курсах.

Любой, кто хочет получить признанное во всем мире удостоверение компетентного профессионала, может сдать экзамены А+. Программа сертификации поддерживается основными производителями и продавцами аппаратного и программного обеспечения. Тесты, впервые стали доступны в 1993 г., были пересмотрены к 31 июля 1998 г и последний раз обновлены в 2001 году.

Наличие у человека сертификата означает, что он обладает достаточными знаниями, навыками (включая навыки общения с клиентами) для успешной работы в качестве технического специалиста в вычислительных центрах, что подтверждается экспертами ведущих компаний в области компьютерных технологий. Экзамен охватывает широкий диапазон аппаратных и программных технологий, но не привязан к конкретным программам или производителям

http://www.specialist.ru/aplus/aplus.asp

4. Фирма Virtual University Enterprises образована 10 апреля 1997 года для проведения электронных сертификационных экзаменов. Главные офисы фирмы расположены в Миннеаполисе, Нидерландах и Сиднее. Фирма имеет центры тестирования по всему миру.

Процесс сдачи экзамена одинаков для всех центров тестирования VUE.

Необходимо произвести регистрацию на экзамен, используя форму, расположенную на сайте или лично приехать в Центр Компьютерного Обучения, далее оплатить экзамен, после чего Вы будете записаны на сдачу экзамена. После оплаты регистрации система тестирования пересылает экзамен из банка данных VUE.

По окончании тестирования происходит автоматическая печать отчета с результатами. При успешной сдаче экзамена кандидат получает по почте соответствующий сертификат от вендора (Microsoft и пр.).

Сертификация Microsoft, CompTIA, Novell, PTC, Sybase, LPI, Ericsson, CIW, Informix является объективным критерием оценки компетентности компьютерных специалистов, признаваемым во всем мире.

http://www.specialist.ru/VUE/vue.asp

5. Cертификация **ECDL (European Computer Driving Licence).** Главная задачаобеспечить международный стандарт оценки навыков работы с персональным компьютером.

Фонд ECDL-F, осуществляющий проведение и контроль сертификации по всему миру, был создан в 1997 году при поддержке Европейского компьютерного общества CEPIS. Сертификат (документ, подтверждающий квалификацию), который получает пользователь после успешной сдачи экзаменов, называется ECDL - European Computer Driving Licence (за пределами Европы сертификат называется ICDL - International Computer Driving Licence). Сертификат признан более чем в 50 странах мира, включая Великобританию, Германию, Норвегию, Швецию, Финляндию, Канаду, Австралию, Египет и многие другие. ECDL-F гарантирует качество и соблюдение единых требований к работе центров тестирования.

Наличие у человека сертификата означает, что он обладает достаточными знаниями, навыками (включая навыки общения с клиентами) для успешной работы в качестве технического специалиста в вычислительных центрах, что подтверждается экспертами ведущих компаний в области компьютерных технологий. Экзамен охватывает широкий диапазон аппаратных и программных технологий, но не привязан к конкретным программам или производителям.

Сертификация считается пройденной, если Вы успешно сдали 1 теоретический и 6 практических модулей:

- Основные положения информатики (Basic Concepts of Information

Technology) (теоретический)

- Применение компьютера и управление файлами (Using the Computer

and Managing Files)

- Обработка текстов (Word Processing)

- Электронные таблицы (spreadsheets)

- Базы данных (Databases/Filing Systems)

- Презентации (Presentation)

- Обмен информацией (Information and Communication)

http://www.specialist.ru/ECDL/

http://www.brainbench.com/xml/bb/homepage.xml

Другой распространённый вариант – это интервью – как метод аттестации с привлечением сторонних экспертов и комиссии по оценке результатов аттестации.[1]

Задачей настоящего дипломного проекта является решение «локальной», для конкретного предприятия проблемы аттестации ИТ-специалистов, как можно более приблизив тематику вопросов к задачам, выполняемым на этом, отдельно – взятом предприятии ОАО «Троицкая ГРЭС».

# Архитектура программной системы

Архитектура программной системы представлена на рисунке 3.1 и рисунке 3.1а.



**Сервер SQL** проверяет правильностьзапроса. выполняет *его* и отправляет приложению-клиенту

**SQL Links** транслирует

Ответ клиента и посылает его BDE

### Сервер баз данных

Автоматический контроль ссылочной целостности данных

Встроенная защита данных от несанкционированного доступа

Выполнение хранимых процедур, созданных для обработки данных

Автоматическая генерация уникального ключа

Встроенная возможность резервного копирования данных

## ИТА: Аттестация

Начало сессии аттестации

Выбор любого количества категорий, вопросы из которых задавать

Свободное перемещение по базе вопросов, с сохранением уже данных ответов

Возможность выхода из системы с сохранением всех данных ответов

Возможность продолжить сессию аттестации в любое удобное время с сохранением уже данных ответов

Рис.3.1 Архитектура ПС

Приложение включает в себя четыре компонета,составляющие всей системы в

целом, работа с которыми подробно описывается в руководстве пользователя и

администратора в пункте 14 и 10 данного проекта. Как видно архитектура реализует

схему клиент- сервер, которая является наиболее предпочтительней в реализации

данной программной системы аттестации.

# 

# ИТА: Дизайнер-эксперт

Создание категории вопросов

Удаление категории вопросов

Изменение названия категории

Оперативное получение статистической информации по категории (количество вопросов, максимальный балл по категории)

Редактирование вопросов в категории

Встроенный текстовый редактор

Встроенный графический редактор

Возможность передачи текстовой части через буфер обмена

Возможность передачи графической части через буфер обмена

Внесение количества баллов за вариант ответа с автоматическим подсчётом общей суммы баллов за вопрос

Изменение категории вопроса

# 

# 

**ИТА: Руководитель**

Завершение сессии аттестации

Просмотр результатов аттестации

Построение отчётов по результатам аттестации

Сравнительный анализ результатов аттестации в каждой группе специалистов по каждой категории вопросов

Сравнительный анализ результатов аттестации по каждой категории по всем специалистам

Сравнительный анализ результатов аттестации по каждому специалисту по каждой категории

Сравнительный анализ результатов аттестации по всем вышеперечисленным пунктам в динамике с накоплением знаний.

### Сервер баз данных

Автоматический контроль ссылочной целостности данных

Встроенная защита данных от несанкционированного доступа

Выполнение хранимых процедур, созданных для обработки данных

Автоматическая генерация уникального ключа

Встроенная возможность резервного копирования данных

# ИТА: Администратор

Создание типа пользователя (от типа пользователя зависит доступ к каким данным открыт пользователю, связанным с этим типом)

Удаление типа пользователя

Создание пользователя и связь его с типом

Удаление пользователя

Завершение сессии аттестации (с автоматическим подсчётом результатов)

Удаление сессии аттестации (с потерей данных аттестации)

Получение оперативной информации о состоянии сервера баз данных

### Сервер баз данных

Автоматический контроль ссылочной целостности данных

Встроенная защита данных от несанкционированного доступа

Выполнение хранимых процедур, созданных для обработки данных

Автоматическая генерация уникального ключа

Встроенная возможность резервного копирования данных

## ИТА: Аттестация

Начало сессии аттестации

Выбор любого количества категорий, вопросы из которых задавать

Свободное перемещение по базе вопросов, с сохранением уже данных ответов

Возможность выхода из системы с сохранением всех данных ответов

Возможность продолжить сессию аттестации в любое удобное время с сохранением уже данных ответов

**Рис . 3.1a Архитектура ПС**

# 4 Разработка структуры баз данных

### i.Общая характеристика реляционной модели данных

Основы реляционной модели данных были впервые изложены в статье Е.Кодда [3] в 1970 г. Эта работа послужила стимулом для большого количества статей и книг, в которых реляционная модель получила дальнейшее развитие. Наиболее распространенная трактовка реляционной модели данных принадлежит К.Дейту[4]. Согласно Дейту [4], реляционная модель состоит из трех частей:

* Структурной части.
* Целостной части.
* Манипуляционной части.

**Структурная часть** описывает, какие объекты рассматриваются реляционной моделью. Постулируется, что единственной структурой данных, используемой в реляционной модели, являются нормализованные n-арные отношения.

**Целостная часть** описывает ограничения специального вида, которые должны выполняться для любых отношений в любых реляционных базах данных. Это **целостность сущностей** и **целостность внешних ключей**.

**Манипуляционная часть** описывает два эквивалентных способа манипулирования реляционными данными - **реляционную алгебру** и **реляционное исчисление**.

В данной главе рассматривается структурная часть реляционной модели.

#### Типы данных

Любые данные, используемые в программировании, имеют свои типы данных.

**Важно!** Реляционная модель требует, чтобы типы используемых данных были *простыми*.

Для уточнения этого утверждения рассмотрим, какие вообще типы данных обычно рассматриваются в программировании. Как правило, типы данных делятся на три группы:

- Простые типы данных.

- Структурированные типы данных.

- Ссылочные типы данных.

#### Простые типы данных

**Простые, или атомарные, типы данных** не обладают внутренней структурой. Данные такого типа называют **скалярами**. К простым типам данных относятся следующие типы:

- Логический.

- Строковый.

- Численный.

Различные языки программирования могут расширять и уточнять этот список, добавляя такие типы как:

- Целый.

- Вещественный.

- Дата.

- Время.

- Денежный.

- Перечислимый.

- Интервальный.

- И т.д.…

Конечно, понятие атомарности довольно относительно. Так, строковый тип данных можно рассматривать как одномерный массив символов, а целый тип данных - как набор битов. Важно лишь то, что при переходе на такой низкий уровень теряется **семантика (смысл) данных**. Если строку, выражающую, например, фамилию сотрудника, разложить в массив символов, то при этом теряется смысл такой строки как единого целого.

#### Структурированные типы данных

**Структурированные типы данных** предназначены для задания сложных структур данных. Структурированные типы данных конструируются из составляющих элементов, называемых компонентами, которые, в свою очередь, могут обладать структурой. В качестве структурированных типов данных можно привести следующие типы данных:

- Массивы

- Записи (Структуры)

С математической точки зрения массив представляет собой функцию с конечной областью определения. Например, рассмотрим конечное множество натуральных чисел



называемое множеством индексов. Отображение



из множества во множество вещественных чисел задает одномерный вещественный массив. Значение этой функции для некоторого значения индекса называется элементом массива, соответствующим . Аналогично можно задавать многомерные массивы.



Запись (или структура) представляет собой кортеж из некоторого декартового произведения множеств. Действительно, запись представляет собой именованный упорядоченный набор элементов , каждый из которых принадлежит типу . Таким образом, запись есть элемент множества . Объявляя новые типы записей на основе уже имеющихся типов, пользователь может конструировать сколь угодно сложные типы данных.



Общим для структурированных типов данных является то, что они *имеют внутреннюю структуру*, используемую *на том же уровне абстракции*, что и сами типы данных.

Поясним это следующим образом. При работе с массивами или записями можно манипулировать массивом или записью и как с единым целым (создавать, удалять, копировать целые массивы или записи), так и поэлементно. Для структурированных типов данных есть специальные функции - конструкторы типов, позволяющие создавать массивы или записи из элементов более простых типов.

Работая же с простыми типами данных, например с числовыми, мы манипулируем ими как неделимыми целыми объектами. Чтобы "увидеть", что числовой тип данных на самом деле сложен (является набором битов), нужно перейти на более низкий уровень абстракции. На уровне программного кода это будет выглядеть как ассемблерные вставки в код на языке высокого уровня или использование специальных побитных операций.

#### Ссылочные типы данных

**Ссылочный тип данных** (**указатели**) предназначен для обеспечения возможности указания на другие данные. Указатели характерны для языков процедурного типа, в которых есть понятие области памяти для хранения данных. Ссылочный тип данных предназначен для обработки сложных изменяющихся структур, например деревьев, графов, рекурсивных структур.

#### Типы данных, используемые в реляционной модели

Собственно, для реляционной модели данных тип используемых данных не важен. Требование, чтобы тип данных был *простым*, нужно понимать так, что *в реляционных операциях не должна учитываться внутренняя структура данных*. Конечно, должны быть описаны действия, которые можно производить с данными как с единым целым, например, данные числового типа можно складывать, для строк возможна операция конкатенации и т.д.

С этой точки зрения, если рассматривать массив, например, как единое целое и не использовать поэлементных операций, то массив можно считать простым типом данных. Более того, можно создать свой, сколь угодно сложных тип данных, описать возможные действия с этим типом данных, и, если в операциях не требуется знание внутренней структуры данных, то такой тип данных также будет простым с точки зрения реляционной теории. Например, можно создать новый тип - комплексные числа как запись вида , где . Можно описать функции сложения, умножения, вычитания и деления, и *все действия с компонентами* и выполнять только *внутри* этих операций. Тогда, если в действиях с этим типом использовать *только* описанные операции, то внутренняя структура не играет роли, и тип данных извне выглядит как атомарный.



Именно так в некоторых пост реляционных СУБД реализована работа со сколь угодно сложными типами данных, создаваемых пользователями.

#### Домены

В реляционной модели данных с понятием тип данных тесно связано понятие домена, которое можно считать уточнением типа данных.

**Домен** - это семантическое понятие. Домен можно рассматривать как подмножество значений некоторого типа данных имеющих определенный смысл. Домен характеризуется следующими свойствами:

* Домен имеет *уникальное имя* (в пределах базы данных).
* Домен определен на некотором *простом* типе данных или на другом домене.
* Домен может иметь некоторое *логическое условие*, позволяющее описать подмножество данных, допустимых для данного домена.
* Домен несет определенную *смысловую нагрузку*.

Например, домен , имеющий смысл "возраст сотрудника" можно описать как следующее подмножество множества натуральных чисел:



Если тип данных можно считать множеством всех возможных значений данного типа, то домен напоминает подмножество в этом множестве.

Отличие домена от понятия подмножества состоит именно в том, что *домен отражает семантику*, определенную предметной областью. Может быть несколько доменов, совпадающих как подмножества, но несущие различный смысл. Например, домены "Вес детали" и "Имеющееся количество" можно одинаково описать как множество неотрицательных целых чисел, но смысл этих доменов будет различным, и это будут *различные* домены.

Основное значение доменов состоит в том, что *домены ограничивают сравнения*. Некорректно, с логической точки зрения, сравнивать значения из различных доменов, даже если они имеют одинаковый тип. В этом проявляется смысловое ограничение доменов. Синтаксически правильный запрос "выдать список всех деталей, у которых вес детали больше имеющегося количества" не соответствует смыслу понятий "количество" и "вес".

**Замечание**. Понятие домена помогает правильно *моделировать* предметную область. При работе с реальной системой в принципе возможна ситуация когда требуется ответить на запрос, приведенный выше. Система даст ответ, но, вероятно, он будет бессмысленным.

**Замечание**. Не все домены обладают логическим условием, ограничивающим возможные значения домена. В таком случае множество возможных значений домена совпадает с множеством возможных значений типа данных.

#### Отношения, атрибуты, кортежи отношения

#### Определения и примеры

Фундаментальным понятием реляционной модели данных является понятие **отношения**. В определении понятия отношения будем следовать книге К. Дейта

*Определение 1.* **Атрибут отношения** *есть пара вида <Имя\_атрибута : Имя\_домена>.*

Имена атрибутов должны быть уникальны в пределах отношения. Часто имена атрибутов отношения совпадают с именами соответствующих доменов.

*Определение 2*. **Отношение** , определенное на множестве доменов (не обязательно различных), содержит две части: заголовок и тело.



**Заголовок отношения** содержит фиксированное количество атрибутов отношения:



**Тело отношения** содержит множество кортежей отношения. Каждый **кортеж отношения** представляет собой множество пар вида <Имя\_атрибута : Значение\_атрибута>:



таких что значение атрибута принадлежит домену



Отношение обычно записывается в виде:

,



или короче

,



или просто

.



Число атрибутов в отношении называют **степенью** отношения.

Мощность множества кортежей отношения называют **мощностью** отношения.

Возвращаясь к математическому понятию отношения, введенному в предыдущей главе, можно сделать следующие выводы:

Вывод 1. Заголовок отношения описывает декартово произведение доменов, на котором задано отношение. Заголовок статичен, он не меняется во время работы с базой данных. Если в отношении изменены, добавлены или удалены атрибуты, то в результате получим уже *другое* отношение (пусть даже с прежним именем).

Вывод 2. Тело отношения представляет собой набор кортежей, т.е. подмножество декартового произведения доменов. Таким образом, тело отношения собственно и является отношением в математическом смысле слова. Тело отношения может изменяться во время работы с базой данных - кортежи могут изменяться, добавляться и удаляться.

### Предварительная структура базы данных, нормализация

Прежде чем начать проектирование базы данных, необходимо определиться какие данных нам необходимо хранить и их взаимосвязь.

Таблица 4.1 Поля таблицы QUESTIONS

|  |  |  |
| --- | --- | --- |
| QUESTIONS – список вопросов | | |
| ID | Integer | Идентификатор вопроса |
| Q\_TEXT | BLOB | Текст вопроса |
| QPICTURE | BLOB | Граф. часть к вопросу |
| CID | INTEGER | Категория вопроса |
| Q1 | SMALLINT | Балл за вариант ответа |
| Q2 | SMALLINT | Балл за вариант ответа |
| Q3 | SMALLINT | Балл за вариант ответа |
| Q4 | SMALLINT | Балл за вариант ответа |
| Q5 | SMALLINT | Балл за вариант ответа |
| Q6 | SMALLINT | Балл за вариант ответа |
| Q7 | SMALLINT | Балл за вариант ответа |
| Q8 | SMALLINT | Балл за вариант ответа |
| Q9 | SMALLINT | Балл за вариант ответа |

Таблица 4.2 Поля таблицы USERS

|  |  |  |
| --- | --- | --- |
| USERS – список специалистов | | |
| ID | INTEGER | Идентификатор специалиста |
| GID | INTEGER | Принадлежность пользователя к группе |
| TID | INTEGER | Принадлежность пользователя к типу |
| LOGIN | VARCHAR | Ф.И.О специалиста |
| PWD | VARCHAR | Пароль специалиста |

Таблица 4.3 Поля таблицы STORE

|  |  |  |
| --- | --- | --- |
| STORE – данные специалиста | | |
| ID | INTEGER | Идентификатор специалиста |
| UID | INTEGER | Отвечавший пользователь |
| CID | INTEGER | Категория вопросов |
| DATED | VARCHAR | Дата аттестации |
| PERCS | SMALLINT | Результат в % |

Таблица 4.4 Поля таблицы TYPES

|  |  |  |
| --- | --- | --- |
| TYPES – Типы пользователя | | |
| ID | INTEGER | Идентификатор пользователя |
| FLAGS | INTEGER | Признак ответа |
| NAME | VARCHAR | Название типа |

Таблица 4.5 Поля таблицы QGROUPS

|  |  |  |
| --- | --- | --- |
| QGROUPS – Категории вопросов | | |
| ID | INTEGER | Идентификатор пользователя |
| NAME | VARCHAR | Категория вопроса |

Таблица 4.6 Поля таблицы GROUPS

|  |  |  |
| --- | --- | --- |
| QGROUPS – Группы пользователей | | |
| ID | INTEGER | Идентификатор пользователя |
| NAME | VARCHAR | Категория группы |

Таблица 4.7 Поля таблицы ASESSIONS

|  |  |  |
| --- | --- | --- |
| ASESSIONS – Активные сессии | | |
| ID | INTEGER | Идентификатор пользователя |
| UID | INTEGER | Отвечавший пользователь |
| NAME | VARCHAR | Ф.И.О пользователя |
| CHOOSENGROUPS | BLOB | Номера выбранной категории |
| GROUPSRP | BLOB | Формат отвеченных вопросов |

###### Нормализация

Отношение находится в **Первой Нормальной Форме** (**1НФ**), если оно содержит только скалярные (атомарные) значения [ ].

Отношение находится во **второй нормальной форме** (**2НФ**) тогда и только тогда, когда отношение находится в 1НФ и нет не ключевых атрибутов, зависящих от части сложного ключа. (**Неключевой атрибут** - это атрибут, не входящий в состав никакого потенциального ключа).



Замечание. Если потенциальный ключ отношения является простым, то отношение автоматически находится в 2 НФ.

### Окончательная структура базы данных

Структура базы данных разработана с использованием Case-средства фирмы Platinum – ErWin 3.5.2. и представлена на рисунке 4.1. Описания таблиц базы данных даны в таблице 4.8.

Таблица 4.8 Описание таблиц

Рис. 4.1 Структура базы данных

|  |  |  |
| --- | --- | --- |
| **Название таблицы** | Назначение | **Примечание** |
| QGROUPS | Список  категорий вопросов | Для этой таблицы создан генератор и триггер для получения уникального идентификатора |
| QESTIONS | Список вопросов | Для этой таблицы создан генератор и триггер для получения уникального идентификатора |
| USERS | Список специалистов, имя и пароль | Для этой таблицы создан генератор и триггер для получения уникального идентификатора |
| GROUPS | Список  групп специалистов | Для этой таблицы создан генератор и триггер для получения уникального идентификатора |
| TYPES | Список  типов пользователей. | Для этой таблицы создан генератор и триггер для получения уникального идентификатора |
| ASESSION | Хранит признак открытой сессии специалистом и дату/время начала сессии |  |

Контроль за ссылочной целостностью данных осуществляется при помощи первичных ключей (primary key), внешних ключей (foreign key) и триггеров. Полный текст метаданных структуры базы данных дан в приложении 1.

# 5 Технология проведения аттестации с использованием

**данной системы**.

## a.Технология проведения аттестации на ОАО «Троицкая ГРЭС» и ее недостатки.

На данный момент, на ОАО «Троицкая ГРЭС» действует следующий порядок проведения аттестации ИТ-специалистов рисунок 2.1:

Руководитель группы АСУ готовит вопросы для проведения аттестации.

Назначается срок аттестации и формируется состав комиссии

В назначенный срок специалисты в присутствии комиссии отвечают на подготовленные вопросы в письменном виде в свободной форме

Заключения передаются руководителю группы АСУ

Комиссия проверяет ответы на вопросы и делает заключение по каждому специалисту

Рис 5.1 Порядок проведения аттестации

По составленным заключениям и выносятся соответствующие административные *решения*:

- О служебном продвижении

- Об изменении характера трудовых обязанностей

- Об изменении оплаты труда или иных материальных условий

- О направлении на учебу

- О необходимости повторной аттестации

В ходе анализа процесса аттестации были выявлены следующие недостатки существующего порядка проведения аттестации:

- Сложно выбрать подходящее время для проведения аттестации,не нарушив

при этом рабочий график всех участвующих (комиссия и специалисты).

- Потеря времени на подготовку вопросов руководителем.

- Нет определённого критерия выбора вопросов для проведения аттестации.

- Как правило, не учитываются реальные задачи, выполняемые

специалистами.

- Неоднозначность оценки результатов аттестации, влияние субъективного

мнения.

- Продолжительность процесса аттестации и подготовки заключения.

В рамках настоящего дипломного проекта был предложен и согласован новый порядок проведения аттестации ИТ-специалистов, с учётом возможностей разработанной программной системы. Модифицированная технология аттестации с использованием программной системы рассматривается в разделе 5.

**b.Технология проведения аттестации с использованием данной системы.**

Прежде чем описать технологию проведения аттестации данной системы выявим и приведем некоторые варианты использования разрабатываемой системы , с помощью которой руководство получает данные в зависимости от сложившейся ситуации( рис.4.1):

**Ситуация**: Приём на работу специалиста.

Решение

Провести аттестацию кандидатов, построить отчёт по специалистам, выбрать кандидата с максимальным баллом во всех категориях.

**Ситуация**: Проверка знаний персонала с целью выявления соответствия знаний занимаемой

должности

Решение

Построить отчёт по группам специалистов , выбрать специалистов с максимальным баллом в группе. Присвоить

соответствующую категорию или разряд прошедшему проверку в полном объеме.

Рис. 5.2 Варианты использования системы

Вариантов ситуаций может быть множество, так же как и множество ветвей их решения. Здесь мы попытаемся обобщить какие выводы можно сделать по результатам аттестации:

-О служебном продвижении

-Степень готовности специалиста к работе на новом рабочем месте, в

новой должности

-Об изменении характера трудовых обязанностей

-Возможности специалистов приспособится к новым условиям работы

-Какие специалисты требуют большей подготовки

-О взаимозаменяемости специалистов

-О направлении на учебу

-О необходимости повторной аттестации

-Оценить результаты программ обучения персонала

-Об изменении оплаты труда или иных материальных условий

Одним из самых важных является тот факт, что сам процесс был автоматизирован (рис 4.2). Теперь для проведения аттестации, достаточно объявить об этом участникам (специалистам),которым необходимо пройти

тестирование по конкретной категории вопросов. Тот факт, что система позволяет прервать процесс аттестации, не прерывая сессии аттестации с сохранением всех ранее данных ответов, позволяет специалистам самим определить подходящее время для ответа на вопросы, при необходимости прерываться на свою основную работу и продолжить позже. Возникает вопрос, как долго может длиться такая сессия? С точки зрения системы нет ограничений на продолжительность сессии аттестации. Реально же продолжительность устанавливается в организационном порядке исходя из количества вопросов и загруженности специалистов основной работой, причём сроки могут теперь устанавливаться индивидуально.

Завершить сессию аттестации может руководитель либо администратор системы, после чего результаты аттестации уже готовы для анализа руководителем.

Важным так же является тот факт, что теперь специалист отвечает на вопросы на своём рабочем месте в привычной обстановке, что безусловно уменьшает влияние ситуативных факторов и положительно скажется на конечных результатах аттестации.

Тем самым были исключены некоторые недостатки обычного порядка проведения аттестации ИТ-специалистов, однако некоторые ещё остались.

Теперь необходимо определиться с тем, кто готовит вопросы к аттестации. Есть вариант использовать вопросы из внешних источников или сторонних экспертов, но всё же такие вопросы будут далеки от реальных задач. Решение напрашивается само собой – кто же лучше разбирается в тонкостях определённой задачи как не специалист, работающий с этой задачей ?

Был предложен следующий вариант подготовки вопросов к аттестации: Вопросы готовят сами специалисты, каждый специалист для своей задачи, т.е. каждый специалист теперь сам эксперт в рамках задач, которые он выполняет. Вопросы разбиваются на категории в соответствии с выполняемыми задачами на том или ином рабочем месте. Категории вопросов содержат несколько правильных ответов. Каждому правильному ответу присваивается свой вес от 0 до 99 , который в свою очередь влияет на общий балл тестируемого специалиста за конкретный вопрос конкретной категории. Но если в группе ответов на конкретный вопрос будет хоть один неправильный ответ ,то данный вопрос помечается как ошибочный.

Такой подход подготовки вопросов даёт ряд преимуществ:

- Тематика вопросов уходит от общих знаний и становится максимально приближена к конкретным задачам, выполняемым ИТ-специалистами на конкретном предприятии

- Со временем накапливаются информация о том, что конкретно должен знать специалист, работающий над той или иной задачей

- На основании результатов аттестации можно делать выводы о том, какого специалиста можно задействовать на какую задачу, на время отсутствия основного исполнителя

- Результаты аттестации могут помочь в составлении графика отпусков

Таким образом, созданная система становится не просто базой данных результатов аттестации, а базой знаний проверяемого персонала.

**Преимущества** проведения аттестации с использованием системы аттестации ИТ – специалистов в целом:

-Оперативность – нет необходимости собирать комиссию и становиться

реальным быстрое получение результатов проверки

-Объективность – независимость от частного мнения и отдельных суждений

- Надёжность – свобода от влияния ситуативных факторов

- Достоверность – оценивается реальный уровень владения навыками

- Прогнозтичность – данные о том, к каким видам задач специалист способен потенциально.

- Комплексность – оценивается весь круг задач выполняемых на предприятии ИТ-специалистами

Руководитель

Специалист

Результаты

Подготовка вопросов

Аттестация

Подготовка вопросов

Оперативное проведение аттестации и получение результатов

- Тематика вопросов уходит от общих знаний и становится максимально приближена к конкретным задачам, выполняемым ИТ-специалистами на конкретном предприятии

- Со временем накапливаются информация о том, что конкретно должен знать специалист, работающий над той или иной задачей

- На основании результатов аттестации можно делать выводы о том, какого специалиста можно задействовать на какую задачу, на время отсутствия основного исполнителя

- Результаты аттестации могут помочь в составлении графика отпусков

Специалист

- Объективность – независимость от частного мнения и отдельных суждений

- Надёжность – свобода от влияния ситуативных факторов

- Достоверность – оценивается реальный уровень владения навыками

- Прогнозтичность – данные о том, к каким видам задач специалист способен потенциально

- Комплексность – оценивается весь круг задач выполняемых на предприятии ИТ-специалистами

Рис. 5.3 Процесс аттестации при внедрении программной системы

## 6 Разработка компонент программной системы в среде

###### Delphi 7.0

В состав системы входят следующие компоненты:

- Сервер баз данных Interbase для платформы Linux-x86

- Рабочее место администратора системы

- Рабочее место дизайнера-эксперта

- Рабочее место проведения аттестации

- Рабочее место руководителя

*Рабочее место администратора* – компонент программной системы предназначенный для управления безопасностью хранения данных, управление пользователями системы.

Возможности:

- Создание типа пользователя (от типа пользователя зависит доступ к каким данным открыт пользователю, связанным с этим типом)

- Удаление типа пользователя

- Создание пользователя и связь его с типом

- Удаление пользователя

-Завершение сессии аттестации (с автоматическим подсчётом результатов)

- Удаление сессии аттестации (с потерей данных аттестации)

- Получение оперативной информации о состоянии сервера баз данных

#### Рабочее место руководителя – компонент программной системы предназначенный для анализа результатов аттестации и построения соответствующих отчётов.

Возможности:

- Завершение сессии аттестации

- Просмотр результатов аттестации

- Построение отчётов по результатам аттестации

а) Сравнительный анализ результатов аттестации в каждой группе специалистов по каждой категории вопросов

б) Сравнительный анализ результатов аттестации по каждой категории по всем специалистам

в) Сравнительный анализ результатов аттестации по каждому специалисту по каждой категории

г) Сравнительный анализ результатов аттестации по всем вышеперечисленным пунктам в динамике с накоплением знаний.

*Рабочее место дизайнера-эксперта* – компонент программной системы, предназначенный для разработки вопросов, группировки их по категориям и внесение в базу данных.

Возможности:

- Создание категории вопросов

- Удаление категории вопросов

- Изменение названия категории

- Оперативное получение статистической информации по категории (количество вопросов, максимальный балл по категории)

- Редактирование вопросов в категории

а) Встроенный текстовый редактор

б) Встроенный графический редактор

в) Возможность передачи текстовой части через буфер обмена

г) Возможность передачи графической части через буфер обмена

- Внесение количества баллов за вариант ответа с автоматическим подсчётом общей суммы баллов за вопрос

- Изменение категории вопроса

*Рабочее место проведения аттестации* – компонент программной системы, предназначен для автоматизированного проведения аттестации специалистов на своих рабочих местах.

Возможности:

- Начало сессии аттестации

- Выбор любого количества категорий, вопросы из которых задавать

- Свободное перемещение по базе вопросов, с сохранением уже данных ответов

- Возможность выхода из системы с сохранением всех данных ответов

- Возможность продолжить сессию аттестации в любое удобное время с сохранением уже данных ответов

*Сервер баз данных Interbase для платформы Linux – x86* – компонент программной системы предназначенный для хранения и предварительной обработки всех данных используемых системой и как следствие осуществляющий связь между компонентами.

Возможности (использованные):

- Автоматический контроль ссылочной целостности данных

-Встроенная защита данных от несанкционированного доступа

- Выполнение хранимых процедур, созданных для обработки данных

- Автоматическая генерация уникального ключа

- Встроенная возможность резервного копирования данных

**Общие возможности для всех компонент системы:**

- Однотипная регистрация в системе личным именем пользователя и паролем

- Автоматическое сжатие данных при передаче на сервер баз данных (текстовая и графическая часть вопроса), с использованием алгоритма адаптивного сжатия Хаффмена

- Автоматическая распаковка сжатых данных при приёме с сервера баз данных (текстовая и графическая часть вопроса)

- Работа всех компонент в корпоративной сетевой среде либо в сети Internet

Обоснование выбора сервера баз данных Interbase:

- Поддерживает архитектуру клиент-сервер

- Поддерживает процедурное расширение языка SQL

- Не сложен в установке и настройке

- Компактен

Обоснование выбора серверной платформы

- С финансовой точки зрения Linux обладает одним весьма существенные, достоинством — она не коммерческая. В отличие от операционной системы Unix, Linux распространяется бесплатно по генеральной открытой лицензии GNU в рамках Фонда бесплатного программного обеспечения (Free Software Foundation), благодаря чему эта ОС доступа всем желающим.

- Система Linux отличается высокой произ­водительностью и гибкостью и предоставляет все средства Unix, включая возможность работы в многозадачном и многопользовательском режимах.

- Высокая надёжность

- Развитая сетевая архитектура

###### 7 Формирование отчетов

## Отчеты формируются на основе результатов полученных при закрытии сессии . На рисунке 7.1 и 7.2 представлены формы формирования отчетов программной системы.

## 

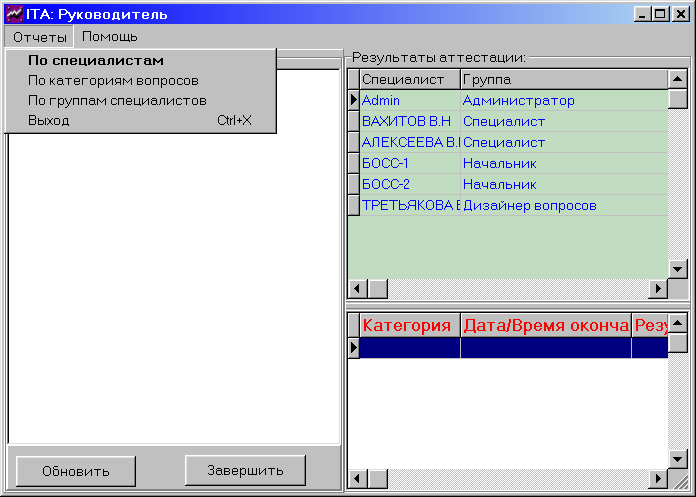
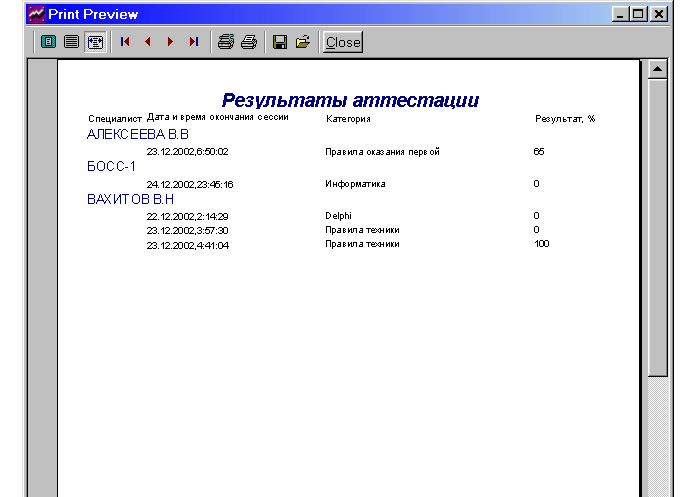


Рис.7.1 Формирование отчетов из ITA-Руководитель



## 

Рис.7.2 Формирование отчетов с выводом результатов

# Решение проблемы защиты информации

**8.1 Основные принципы**

Решение задачи разработки и создания технических систем обеспечения безопасности состоит из трех обязательных и взаимосвязанных этапов. На первом этапе необходимо количественно оценить и определить все характеристики объекта информационной защиты, т.е. понять, каким образом может произойти утечка информации, и достоверно «вычислить» вероятные каналы утечки. В результате должны быть получены обоснованные рекомендации по проведению мероприятий, гарантирующих достижение требуемого уровня информационной безопасности.

На втором этапе необходимо провести комплексную разовую проверку, так называемую «очистку» объекта, т.е. выявить и нейтрализовать уже созданные с помощью технических средств нападения (если таковые будут обнаружены) каналы утечки. И только затем следует приступать к третьему этапу — установке на объекте системы информационной защиты, т.е., начиная с момента окончания «очистки», «отрезать» возможность утечки информации.

При этом, если на первых двух этапах имеется возможность проведения дополнительного обследования и проверки, уточнения неясных моментов и т.п. (семь раз проверить), то последний этап, связанный, как правило, с какими-либо монтажными, строительными, интерьерными работами, желательно проводить как единое, цельное мероприятие (один раз отрезать). Это тем более справедливо потому, что разработка подобной системы представляет собой проектирование комплекса технических средств и систем, осуществляемое по критерию «максимально эффективного использования вложенных средств с точки зрения достижения требуемого уровня информационной защиты».

**8.2 Защита информации средствами СУБД InterBase 6.0**

Для защиты информации используется встроенный механизм сервера баз данных Interbase. В дополнении к нему был модифицирован метод регистрации пользователей в системе (рис. 1.1).

Система аттестации

СЕРВЕР БАЗ ДАННЫХ

**Имя и зашифрованный пароль**

**пользователя**

**Реальное подключение**

Рис 1.1 Двухуровневая регистрация

Алгоритм:

1. Клиент логинится своим именем и паролем в сети.
2. Сервером сети проверяется введённый пароль
3. Клиент подключается от имени пользователя к базе данных.
4. Проверяется зашифрованный пароль
5. Выполняется реальное подключение с реальным пользователем

**8.3 Защита от повреждения носителей**

Данная глава посвящена защите информации от повреждения физических носителей**.**

Существует два основных метода:

-резервное копирование;

-избыточность носителей;

Мы используем оба этих метода. Резервное копирование InterBase поддерживает в двух режимах: “горячее” и “холодное”. “Горячее” используется во время работы БД для уменьшения потерь и времени восстановления после сбоев. “Холодное” резервное копирование выполняется в нерабочее время при остановке экземпляра InterBase путем копирования средствами ОС файлов БД. Для повышения надежности резервное копирование выполняется на отдельный сервер. Для уменьшения объема хранимых данных организовано инкрементальное резервирование. В качестве средств решения данной задачи служат:

-утилиты tar и gzip;

-ftp-сервер;

-ОС Linux;

Для предупреждения сбоев при отказе дисков используется RAID-массив. При рассмотрении вариантов мы остановились на организации программного RAID-1-массива(зеркало). Данный тип RAID-массива выбран с учетом минимизации времени восстановления работоспособности после сбоя и уменьшения финансовых затрат**.**

Каждый месяц полная копия архива СУБД, в том числе конфигурационные и исполняемые файлы InterBase записываются на компакт диск, что позволит, даже в случае серьезной аварии, восстановить систему в течение нескольких часов.

В Interbase встроен механизм резервного копирования, которым необходимо пользоваться администратору системы для предоставления сохранности данных. Доступен он через программу администрирования IBconsole.exe и выбора соответствующего пункта меню. Программа администрирования позволяют выполнять следующие функции:

* 1. Установка связи с сервером (локальным или удаленным);
  2. Ведение списка пользователей сервера;
  3. Создание баз данных;
  4. Резервное копирование и восстановление данных.

**8.4Опознавание пользователя**

Первой задачей СУБД является опознавание пользователя. InterBase предоставляет 2 способа идентификации пользователей:

-по паролю;

-операционной системой;

Опознавание ОС имеет недостаток в том, что пользователь должен быть зарегистрирован в системе, на которой работает экземпляр InterBase.

Опознавание по паролю может производиться как самой СУБД, так и специальными службами, такими как LDAP, NDS, KERBEROS, что позволяет повысить защищенность и предоставляет дополнительные возможности в корпоративной работе.

При опознавании по паролю самой СУБД нет контроля длины и сложности пароля, необходимо использовать продукты сторонних производителей.

Мы используем опознавание по паролю самой СУБД, так как InterBase работает на отдельном сервере и создавать на нем пользователей ОС неоправданно как с точки зрения безопасности, так и с точки зрения администрирования. Опознавание специальными службами не используем ввиду малого числа пользователей и невысоких требований конфиденциальности.

**8.5 Целостность данных**

Помимо управления доступом к данным необходимо защитить сами данные от потери целостности, то есть обеспечит непротиворечивость данных. InterBase имеет для этого следующие средства:

-типизация;

-поддержка ключей(первичных и внешних);

-индексация;

-последовательности;

-триггеры;

-транзакции;

Отдельно нужно отметить механизм транзакций. Любая современная СУБД должна поддерживать транзакции для однозначности изменений БД.

Для реализации механизма транзакций InterBase использует очень надежные средства сегменты отката, и журнал транзакций.

**8**.**6 Защита средствами ОС**

##### Защитить операционную систему гораздо сложнее, чем СУБД. Это обусловлено тем, что число различных типов защищаемых объектов в современных ОС может достигать нескольких десятков, а число различных типов защищаемых информационных потоков – нескольких сотен. ОС имеет очень сложную внутреннюю структуру и поэтому задача построения адекватной политики безопасности для ОС решается значительно сложнее, чем для СУБД.

##### Для полноценной работы СУБД InterBase необходима надежная, пусть даже не очень высокоскоростная платформа, которая в состоянии обеспечить требуемые ресурсы при распределении памяти, процессорного времени и обладающая высоким уровнем защиты. Вышеуказанным требованиям соответствует ОС Linux Red Hat 7.0, на которой, собственно находится база. Пользователи, проходящие тест, логинятся пользователем с минимальным набором прав, достаточных для работы только с базой данных, но недостаточных для ,копирования и проведения над базой деструктивных действий. Между сервером и клиентом поддерживается протокол шифрования DES или MD5.

**8.7 Прочие меры защиты информации**

К таким методам можно отнести следующие:

-ограничение доступа в помещение, где находится сервер;

-хранение резервных копий информации в отдельном здании на случай стихийных бедствий;

-использование бесперебойных источников питания UPS;

-меры воздействия на пользователей при нарушении политики безопасности предприятия;

**8.7 Законодательная часть.**

Вопросы безопасности на предприятии регулируются официальным документом – «Концепция безопасности информационно-вычислительных систем» утверждённой 30.12.1999г. заместителем председателя правления РАО «ЕЭС России», начальником департамента экономической безопасности и режима РАО «ЕЭС России» В.Ю. Платоновым. Приведём некоторые положения из этого документа:

1. Управление доступом к ресурсам информационной системы должно осуществляться с помощью идентификации и проверки подлинности субъектов доступа при входе в систему по идентификатору (коду) и паролю условно-постоянного действия.
2. Особенно должен контролироваться доступ в ИВС через каналы обмена данными и должны быть предприняты особые меры защиты сетевого уровня ЛВС.
3. Уровень конфиденциальности используемых средств вычислительной техники должен быть не ниже уровня конфиденциальности обрабатываемой с помощью этого средства информации.
4. Должна осуществляться идентификация терминалов, ЭВМ, узлов сети ЭВМ, каналов связи, внешних устройств ЭВМ по логическим именам и/или адресам.
5. Должна осуществляться регистрация входа/выхода субъектов доступа в систему/из системы.
6. Должна осуществляться регистрация изменений полномочий субъектов доступа и статуса объектов доступа и сигнализация попыток нарушения защиты.
7. Должен быть предусмотрен администратор (служба) защиты информации, ответственный за ведение, нормальное функционирование и контроль работы средств защиты информации от НСД. Администратор должен иметь необходимые средства оперативного контроля и воздействия на систему защиты информации.
8. Должны использоваться сертифицированные средства защиты информации.

## 9 Методика обработки данных, полученных в результате аттестации.

Алгоритм методики обработки следующий:

- Каждый вопрос имеет свой вес – т.е. то максимальное количество баллов, которые можно набрать, выбрав все правильные ответы. Это даёт возможность разбивать вопросы по их значимости в конкретной категории задач.

- С каждым ответом на вопрос связан балл, который получает специалист, выбрав этот вариант ответа. Это даёт возможность выделять наиболее оптимальные пути решения той или иной задачи, по которой составлен вопрос.

* Если специалист, отвечая на вопрос, выбрал хоть один неверный вариант ответа, т.е. вариант с баллом равным нулю, ответ на весь вопрос считается неверным, даже если он выбрал при этом все правильные варианты.
* По выбранной категории количество правильных вопросов

рассчитывается в процентном соотношении по следующей формуле:



Где A*min –* минимальное количество баллов по отвеченным правильно вопросам из данной категории

A*max -* максимальное число баллов правильных вопросов в данной категории

Result – количество правильных вопросов в процентном отношении

# 10 Инструкция по экплуатации

## Компонент «ИТА: Аттестация»

Компонент системы «Аттестация ИТ-специалистов», предназначен для использования специалистами на своих рабочих местах при проведении аттестации. Файл запуска - ita\_att.exe

### Минимальные системные требования

- IBM-совместимый компьютер, не ниже:

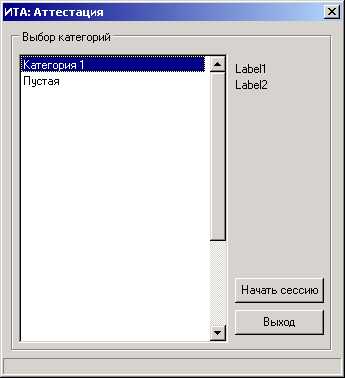
- Pentium I-200ММХ, RAM-32Mb, SVGA-800\*600\*16bit

- Свободное пространство на жёстком диске не менее 5Мб.

- Операционная система Windows 95/98/ME или Windows NT 4.0/5.0

### Порядок работы

Аттестация начинается с процесса регистрации в системе, после чего предлагается выбрать категории, вопросы из которых будут задаваться.



Список доступных категорий

Начать сессию аттестации

Рис. 10.1.1. Выбор категорий

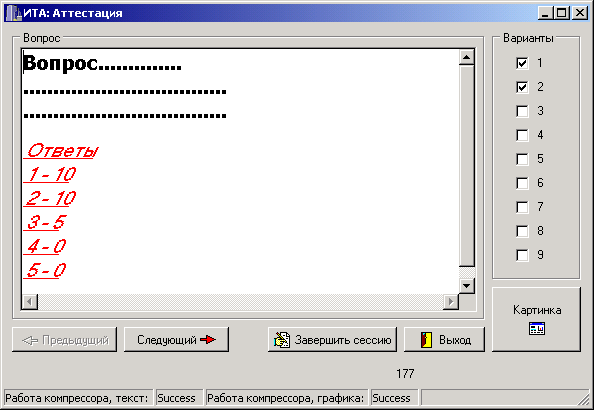
Несколько категорий можно выбрать, удерживая кнопку CTRL. Когда категории выбраны, начинается сессия аттестации.

Перемещение по базе вопросов

Графическая часть вопроса

Выбор вариантов ответов

Рис. 10.1.2. Сессия аттестации



Прерывает процесс аттестации, с возможность вернуться к нему позже

Если вопрос содержит графическую часть, тогда становится доступна кнопка «Картинка», нажав на которую можно ознакомиться с графической частью вопроса.

Рис. 10.1.3. Графическая часть вопроса

Когда даны ответы на все вопросы, необходимо воспользоваться кнопкой «Выход» и выйти из программы. Можно прервать процесс аттестации, и вернуться к нему вновь, при этом все ранее данные ответы будут сохранены.

## Компонент «ИТА: Дизайнер-эксперт»

Компонент системы «Аттестация ИТ-специалистов», предназначен для использования экспертами при подготовке вопросов к аттестации. Файл запуска – ita\_dsgn.exe

### Минимальные системные требования

- IBM-совместимый компьютер, не ниже:

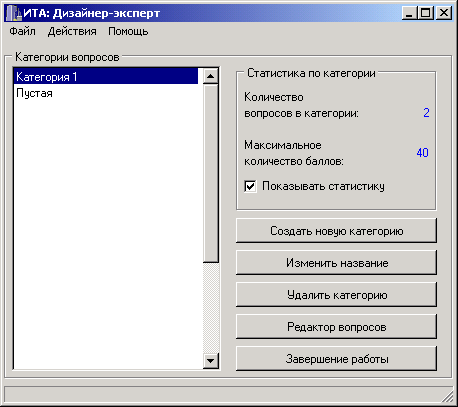
- Pentium I-200ММХ, RAM-32Mb, SVGA-800\*600\*16bit

- Свободное пространство на жёстком диске не менее 10Мб.

- Операционная система Windows 95/98/ME или Windows NT 4.0/5.0

### Порядок работы

После регистрации эксперта в системе, появляется окно со списком категорий вопросов.



Окно статистики

Кнопки управления категориями

Переход в режим редактирование вопросов в выбранной категории

Рис. 10.2.1. Окно управления категориями

В этом окне (рис.10.2.2) можно создать новую категорию вопросов, удалить категорию, изменить название категории, а так же посмотреть статистику – количество вопросов в категории и максимальное количество балов. При удалении категории удаляются все вопросы, входящие в эту категорию. Переход в режим редактирования вопросов в категории осуществляется нажатием соответствующей кнопки.

Баллы за варианты ответов на вопрос

Рис 10.2.2 Дизайнер категории

Переход в режим редактирования вопроса

изменить категорию вопроса

Перемещение по базе вопросов

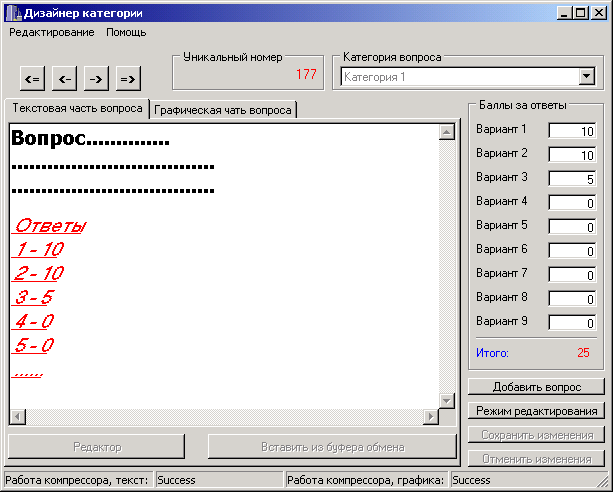


Рис. 9.2.2 Дизайнер вопросов

В этом режиме можно свободно перемещаться по базе вопросов. Для того чтобы внести изменения в текстовую или графическую часть вопроса, изменить количество баллов за ответ или категорию вопроса, необходимо перейти в режим редактирования вопроса. Для перехода обратно, в режим перемещения по базе вопросов, необходимо либо сохранить сделанные изменения, либо отменить их.

Для редактирования вопросов предусмотрен редактор – графической части , который предлагает стандартные графические примитивы для подготовки графической части вопроса ,если возможностей этого редактора не достаточно, можно выполнить редактирование в любом текстовом и графическом редакторе, а затем перенести подготовленные части через стандартный буфер обмена. На рисунке 10.2.4 представлен интерфейс встроенного графического редактора.

Картинка

для вопроса

Встроенный графический редактор

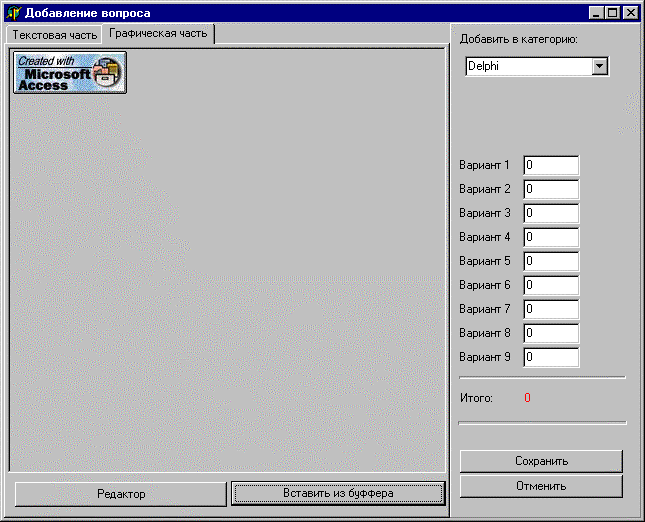


Рис 10.2.3 Текстово-графический редактор

Вставка картинки из буфера

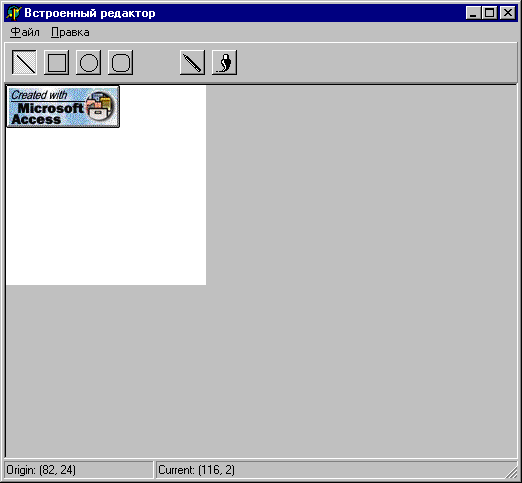


Рис 10.2.4 Встроенный графический редактор

## Компонент ИТА: Руководитель

Компонент предназначен для анализа результатов аттестации руководителем. Файл запуска ita\_boss.exe.

### Минимальные системные требования

- IBM-совместимый компьютер, не ниже:

- Pentium II-450, RAM-64Mb, SVGA-800\*600\*16bit

- Свободное пространство на жёстком диске не менее 10Мб.

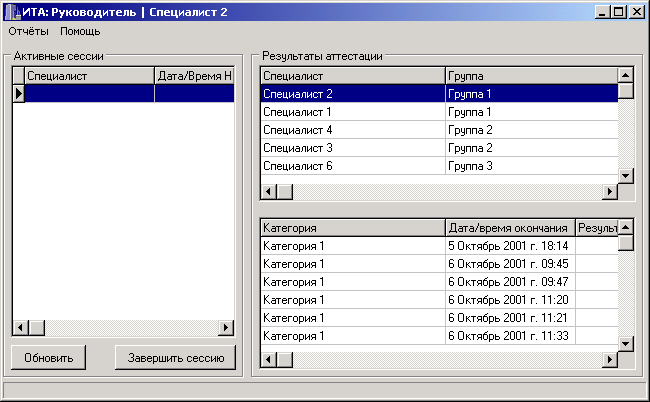
- Операционная система Windows 95/98/ME или Windows NT 4.0/5.0

### Порядок работы

После регистрации руководителя в системе, открывается главное окно:

Результаты аттестации специалиста

Выбор специалиста



Завершение выбранной сессии аттестации

После выбора специалиста в нижнем окне появляется все результаты аттестации по выбранному специалисту. Для того чтобы построить комплексный отчёт необходимо выбрать меню отчёты ,в котором можно просмотреть текущее состояние законченных сессий аттестации по выбранным категориям рисунок 10.3.1.При необходимости можно вывести комплексный отчет по результатам аттестации ,показанный на рисунке 10.3.2,в котором предлагается возможность более детальное ознакомление с результатами в процентном содержании полученных данных ,время завершении сессии, дата и другое ,а также вспомогательные функции такие как распечатка на принтер, запись в файл и т.д.

Меню выбора отчёта

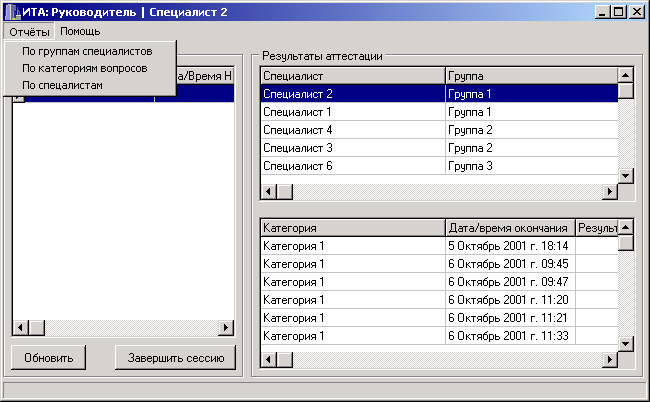
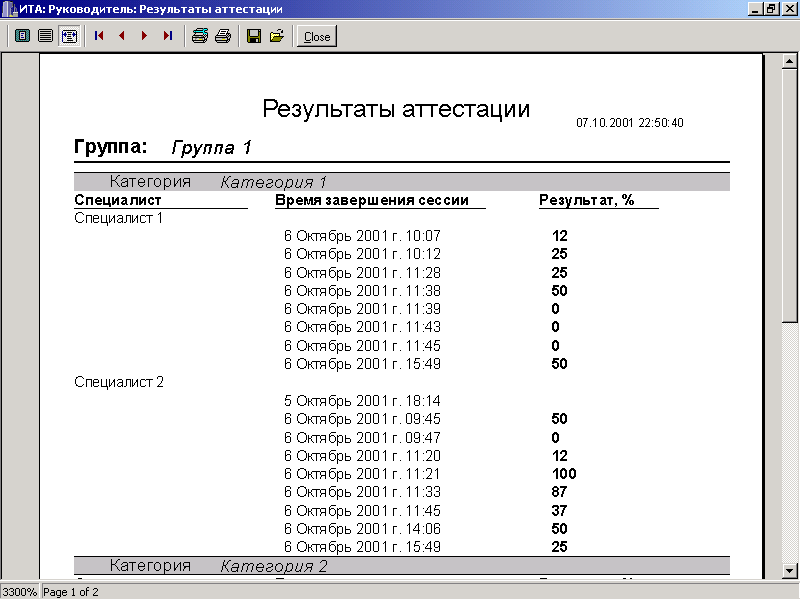


Рис. 10.3.1 Отчеты по группам

**Рис.10.3.2 Комплексный отчет**



# 11 Экономический раздел

**11.1 Постановка задачи**

При разработке любого проекта огромную роль играют организация производства и управление производством. Это прослеживается как в научно-исследовательской работе и опытно-конструкторской разработке, так и при проектировании программной системы.

Целью представленного дипломного проекта является разработка ПС аттестации ИТ- специалистов.

Ввиду большой сложности и комплексности проведения работ по созданию программных систем, одновременного участия многих исполнителей, необходимости параллельного выполнения работ, зависимости начала одних работ от результатов других применялись системы сетевого планирования и управления. Такие системы основываются на применении сетевых моделей планируемых процессов, допускающих использование современной вычислительной техники, а также, позволяющих быстро определить последствия различных вариантов управляющих воздействий и находить наилучшие из них.

В разделе построена сетевая модель выполнения дипломного проекта, определены затраты при производстве системы и при её использовании, определены показатели экономического эффекта.

## Построение сетевого графика и расчет его параметров

### 11.2.1Построение сетевого графика

Метод управления комплексными разработками, который позволяет графически отразить и связать между собой все узловые события и работы, обеспечивающие выполнение конечной цепи, а также проследить за возможными отклонениями при выполнении тех или иных работ называется сетевым планированием и управлением. Сетевое моделирование, то есть разработка, корректировка модели и управление комплексом работ с помощью модели, лежит в основе этого метода.

Сетевой метод планирования дает возможность руководителям своевременно получать достоверную информацию о состоянии дел, о возникших проблемах и возможностях их устранения, так как этот метод позволяет детально рассмотреть весь график работ и сконцентрировать внимание на “критических” работах что обеспечивает качество и своевременность выполнения работ, а также дает необходимые данные для расчета себестоимости проекта.

Итак, планирование проекта с применением сетевого метода состоит из нескольких этапов. Во-первых, перечисляются события и работы. Во-вторых, устанавливается топология сети. В третьих, по данной теме строится сетевой график. Затем, необходимо определить продолжительность работ и рассчитать параметры, после чего, определить продолжительность критического пути и, по окончанию вышеуказанного, провести анализ и оптимизацию сетевого графика, если это необходимо.

Сетевой график представляет собой наглядно изображенный план, определяющий логическую последовательность всех действий. Он изображается в виде ориентированного графика, дугами которого являются работы, а вершинами - события.

Работой является тот или иной процесс. Для каждой работы существует начальное и конечное событие. Начальное событие – это событие, после которого начинается работа, конечное событие наступает после завершения этой работы.

Событием называется определенное состояние (момент времени) в процессе выполнения комплекса работ, означающее изменение состава выполняемых или доступных к выполнению работ. Это факт окончания одной или нескольких работ и возможность начала одной или нескольких работ.

При составлении перечня событий и работ указывают кодовые номера событий и их наименование в последовательности от исходного события к завершающему. Все события, лежащие между исходным и завершающим событием, называются промежуточными событиями. При расположении кодовых номеров и наименований работ перечисляются все работы, имеющие общее начальное.

Методом экспертных оценок, в процессе планирования, получают исходные данные.

Ожидаемое время выполнения работы (*tожид*.) рассчитано по двухоценочной методике, исходя из минимальной (*tмин*.) и максимальной (*tмакс*.) оценок продолжительности работы. При этом, предполагается, что минимальная оценка соответствует наиболее благоприятным, а максимальная - наиболее неблагоприятным условиям работы.

Итак, ожидаемая продолжительность каждой работы рассчитывается по формуле:

*tожид.= 0,6 tмин. + 0,4 tмакс*.  (11.1)

Комплекс работ и их характеристики представлены в табл. 6.1

Таблица 11.1 Комплекс работ и их характеристики

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код | Наименование работы | Продолжительн., дни | | | Исполнители, чел. | | | |
| мин. | макс. | Ожид | Гл. сп-т | Прогр. | Рук-ль проекта | Конс |
| 0,1 | Получение и анализ задания | 1 | 3 | 2 | 1 | - | - | - |
| 1,2 | Анализ состояния вопроса и изучение литературы | 1 | 3 | 2 | 1 | 1 | - | 1 |
| 1,3 | Обзор инструментальных средств для разработки системы | 3 | 5 | 4 | - | 1 | - | - |
| 1,4 | Анализ ТЗ и составление плана-графика разработки | 3 | 5 | 4 | - | 1 | - | 1 |
| 2,4 | Проектирование архитектуры системы | 3 | 5 | 4 | 1 | 1 |  | - |
| 2,5 | Построение сетевого графика | 2 | 4 | 3 | - | - | 1 | - |
| 2,6 | Выработка требований к функциям разрабатываемой системы | 3 | 5 | 4 | - | 1 | - | 1 |
| 2,7 | Разработка раздела “Безопасность жизнедеятельности” | 2 | 5 | 4 |  | 1 | - | 1 |
| 3,4 | Установка и настройка ПО для реализации системы | 1 | 3 | 2 | 1 | 1 | - | - |
| 4,9 | Предварительная разработка основных процедур системы | 2 | 4 | 3 | - | 1 | - | 1 |
| 5,12 | Расчет экономической эффективности | 2 | 4 | 3 | - | - | 1 | - |
| 6,8 | Определение общего перечня выпускаемых документов | 2 | 4 | 3 | 1 | 1 | - | - |
| 6,9 | Разработка базы данных | 2 | 4 | 3 |  | 1 | - | - |
| 7,13 | Разработка концепции ПС Аттестации | 4 | 6 | 5 | - | 1 | - | - |
| 8,10 | Согласование концепций с соисполнителями | 2 | 4 | 3 | - | 1 | - | - |
| 9,10 | Разработка системы тестирования | 3 | 5 | 4 | - | 1 | - | 1 |
| 9,11 | Определение перечня документов, предъявляемых на предварительные испытания | 4 | 6 | 5 | - | 1 | - | 1 |
| 10,11 | Разработка графического интерфейса системы | 1 | 3 | 2 | - | 1 | - | - |
| 10,16 | Разработка методики построения задания | 3 | 5 | 4 | 1 | 1 | - | - |
| 11,14 | Согласование концепций соисполнителями | 3 | 5 | 4 | - | 1 | - | - |
| 11,15 | Корректирование интерфейса системы под различное клиентское ПО | 3 | 5 | 4 | 1 | 1 | - | - |
| 12,19 | Оформление листа “Сетевой график” | 2 | 4 | 3 | - | - | 1 | - |
| 13,19 | Согласование с заказчиком | 2 | 4 | 3 | - | 1 | - | - |
| 14,16 | Тестирование и анализ системы | 3 | 5 | 4 | - | 1 | - | - |
| 15,17 | Оформление листа “Архитектура системы” | 2 | 4 | 3 | - | 1 | - | - |
| 16,17 | “Создание компонент  Delphi 7.0” | 2 | 4 | 3 | - | 1 | - | - |
| 16,18 | Написание руководства пользователя, руководства программиста | 5 | 7 | 6 | 1 | 1 | - | - |
| 16,19 | Доработка системы | 4 | 6 | 5 | - | 1 | - | 1 |
| 17,20 | Выпуск документа “Структура программы” | 1 | 3 | 2 | - | 1 | - | - |
| 18,20 | Оформление листа “Технология Аттестации” | 1 | 3 | 2 | - | 1 | - | - |
| 19,20 | Оформление пояснительной записки | 3 | 5 | 4 | - | 1 | - |  |
| 20,21 | Прохождение нормоконтроля | 1 | 3 | 2 | - | 1 | - | - |
| 21,22 | Получение рецензии | 1 | 3 | 2 | - | 1 | - | - |
| 22,23 | Сдача проекта заказчику | 1 | 3 | 2 | 1 | - | - | - |

Сетевой график строится исходя из перечня работ, приведенного в таблице 11.1. Сетевой график представлен на рис. 11.1.

40

40

0

23

7

24

17

5

10

27

347

12

3

8

22

14

7

5

13

27

32

13

2

11

14

3

8

3

4

3

3

2

15

15

0

10

4

3

4

6

38

38

0

22

4

5

30

30

0

19

4

21

21

0

14

17

17

0

11

8

8

0

6

4

4

0

2

34

34

0

20

25

25

0

16

4



4



3

2

2

2



5

4

6



2

4



3



11

11

0

9

8

9

1

4

2

2

0

1

2

31

32

1

18

4

4

2

2

36

36

0

21

3

21

29

8

15

24

32

8

17

0

0

0

0

6

7

1

3

Рис. 11.1 Сетевой график проекта.

На сетевом графике образовалось несколько непрерывных линий. Каждая из них является путем, который образуют совокупность событий и работ. Один из таких путей будет критическим. Критический – это такой путь, продолжительность которого является наивысшей, а резерв времени равен нулю. На схеме он изображен утолщенной линией.

Таблица 11.2 Расчет параметров сетевого графика в целом:

|  |  |
| --- | --- |
| Количество событий | 24 |
| Количество работ | 34 |
| Коэффициент сложности | Кс = 34 / 24 = 1,42 |
| Продолжительность критического пути | 40 дней |

### 11.2.2 Расчет временных параметров событий сетевого графика

Существует три параметра событий:

**-** Ранний срок свершения промежуточного события **tpj** . Известно, что:

**tpj=max(tpi+tij) ,** (11.2)

где **tij**- ожидаемая продолжительность работы,

**tpi**- ранний срок свершения события, непосредственно предшествующего данному.

Ранний срок свершения исходного события принимается равным нулю (**tp0=0**).

**-** Поздний срок свершения промежуточного события **tïi** . Рассчитывается по формуле:

(11.3)

**tïi=min(tïi-tij) ,**

где **tij** - ожидаемая продолжительность работы,

**tïj** - поздний срок свершения события непосредственно следующего за данным промежуточным событием.

Поздний срок свершения завершающего события сетевого графика принимается равным раннему сроку свершения завершающего события: **tïi=tpi**, то есть для завершающего события никакие резервы времени не планируются.

**-** Резерв времени свершения события - это промежуток времени, на который может быть отсрочено событие без нарушения сроков разработки в целом. Образуется у событий, для которых поздний срок свершения события больше раннего срока:

**Ri=tïi-tpi ,** (11.4)

Если **tïi=tpi** (поздний срок свершения события равен раннему сроку свершения события), то такое событие не имеет резерва времени и это событие лежит на критическом пути.

Результаты расчетов временных параметров событий сетевого графика приведены в таблице 11.3.

Таблица 11.3 Временные параметры событий сетевого графика.

|  |  |  |  |
| --- | --- | --- | --- |
| Номер события | Ранний срок свершения | Поздний срок свершения | Резерв времени |
| 1 | 2 | 3 | 4 |
| 0 | 0 | 0 | 0 |
| 1 | 2 | 2 | 0 |
| 2 | 4 | 4 | 0 |
| 3 | 6 | 7 | 1 |
| 4 | 8 | 9 | 1 |
| 5 | 7 | 24 | 17 |
| 6 | 8 | 8 | 0 |
| 7 | 8 | 22 | 14 |
| 8 | 11 | 14 | 3 |
| 9 | 11 | 11 | 0 |
| 10 | 15 | 15 | 0 |
| 11 | 17 | 17 | 0 |
| 12 | 10 | 27 | 17 |
| 13 | 13 | 27 | 14 |
| 14 | 21 | 21 | 0 |
| 15 | 21 | 29 | 8 |
| 16 | 25 | 25 | 0 |
| 17 | 24 | 32 | 8 |
| 18 | 31 | 32 | 1 |
| 19 | 30 | 30 | 0 |
| 20 | 34 | 34 | 0 |
| 21 | 36 | 36 | 0 |
| 22 | 38 | 38 | 0 |
| 23 | 40 | 40 | 0 |

### Расчет временных параметров работ сетевого графика

Ранний срок начала работы: **Tрнij= tpi***.* (11.5)

Поздний срок начала работы: **Тпнij= tïj­ ­­- ti*j****.* (11.6)

Ранний срок окончания работы: **Tpoij= tpi + tij***.*  (11.7)

Поздний срок окончания работы: **Троij= tïj***.*  (11.8)

Полный резерв времени работы: **Rïij= tïj - tpi - tij***.* (11.9)

Частичный резерв времени работы первого рода: **Rïij=tïj- tïi - tij**(11.10)

Частный резерв времени работы второго рода: **Rïij=tpj -tpi -tij**(11.11)

Свободный резерв времени работы: **Rcij=tpj -tïi -tij**(11.12)

Коэффициент напряженности работы: , (11.13)

где **t[max.НЕСОВП]**- продолжительность отрезков максимального пути, проходящего через данную работу, несовпадающих с критическим путем;

**t[кр.несовп.]** - продолжительность отрезков критического пути, несовпадающих с максимальным путем, проходящим через данную работу.

Например, для работы 1-3 :

k1-3 = ( t1-3 + t3-4 + t4-9 ) / ( t1-2 + t2-6 + t6-9 ) = ( 2+4+2 ) / ( 2+4+3 ) = 8/9 = 0,88.

Коэффициент напряженности работ, лежащих на критическом пути, равен единице, так как для них нет резервов времени.

Временные параметры работ сетевого графика представлены в таблице 11.4.

Таблица 6.4 Временные параметры работ сетевого графика.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код работ | Ожидаем-ая продолжитель-ность | Ранн-ий срок начала | Поздний срок начала | Ранний срок окончания | Поздн-ий срок окончания | Полный резерв времени | Частный резерв времени 1 рода | | Частный резерв времени 2 рода | | Свобод-ный резерв времени | | Коэффи-циент напряженности |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | | 8 | | 9 | | 10 | 11 |
| 0, 1 | 2 | 0 | 0 | 2 | 2 | 0 | | 0 | | 0 | | 0 | 1 |
| 1, 2 | 2 | 2 | 2 | 4 | 4 | 0 | | 0 | | 0 | | 0 | 1 |
| 1, 3 | 4 | 2 | 3 | 6 | 7 | 1 | | 1 | | 0 | | 0 | 0,88 |
| 1, 4 | 4 | 2 | 5 | 6 | 9 | 3 | | 3 | | 2 | | 2 | 0,9 |
| 2, 4 | 4 | 4 | 5 | 8 | 9 | 1 | | 1 | | 0 | | 0 | 0,8 |
| 2, 5 | 3 | 4 | 21 | 7 | 24 | 17 | | 17 | | 0 | | 0 | 0,81 |
| 2, 6 | 4 | 4 | 4 | 8 | 8 | 0 | | 0 | | 0 | | 0 | 1 |
| 2, 7 | 4 | 4 | 18 | 8 | 22 | 14 | | 14 | | 0 | | 0 | 0,89 |
| 3, 4 | 2 | 4 | 7 | 6 | 9 | 3 | | 0 | | 0 | | -1 | 0,87 |
| 4, 9 | 3 | 8 | 8 | 11 | 11 | 0 | | 0 | | 0 | | -1 | 0,87 |
| 5, 12 | 3 | 7 | 24 | 10 | 27 | 17 | | 0 | | 0 | | -17 | 0,75 |
| 6, 8 | 3 | 8 | 11 | 11 | 14 | 3 | | 3 | | 0 | | 0 | 0,76 |
| 6, 9 | 3 | 8 | 8 | 11 | 11 | 0 | | 0 | | 0 | | 0 | 1 |
| 7, 13 | 5 | 8 | 22 | 13 | 27 | 14 | | 0 | | 0 | | -14 | 0,92 |
| 8, 10 | 3 | 11 | 12 | 14 | 15 | 1 | | 0 | | 1 | | -2 | 0,90 |
| 9, 10 | 4 | 11 | 11 | 15 | 15 | 0 | | 0 | | 0 | | 0 | 1 |
| 9, 11 | 5 | 11 | 12 | 16 | 17 | 1 | | 1 | | 1 | | 1 | 0,92 |
| 10, 11 | 2 | 15 | 15 | 17 | 17 | 0 | | 0 | | 0 | | 0 | 1 |
| 10, 16 | 4 | 15 | 21 | 19 | 25 | 6 | | 6 | | 6 | | 6 | 0,69 |
| 11, 14 | 4 | 17 | 17 | 21 | 21 | 0 | | 0 | | 0 | | 0 | 1 |
| 11, 15 | 4 | 17 | 25 | 21 | 29 | 8 | | 8 | | 0 | | 0 | 0,69 |
| 12, 19 | 3 | 10 | 27 | 13 | 30 | 17 | | 0 | | 17 | | 0 | 0,81 |
| 13, 19 | 3 | 13 | 27 | 16 | 30 | 14 | | 0 | | 14 | | 0 | 0,92 |
| 14, 16 | 4 | 21 | 21 | 25 | 25 | 0 | | 0 | | 0 | | 0 | 1 |
| 15, 17 | 3 | 21 | 29 | 24 | 32 | 8 | | 0 | | 0 | | -8 | 0,69 |
| 16, 17 | 3 | 25 | 29 | 28 | 32 | 4 | | 4 | | 0 | | -4 | 0,69 |
| 16, 18 | 6 | 25 | 26 | 31 | 32 | 1 | | 1 | | 0 | | 0 | 0,92 |
| 16, 19 | 5 | 25 | 25 | 30 | 30 | 0 | | 1 | | 0 | | 0 | 1 |
| 17, 20 | 2 | 24 | 32 | 26 | 34 | 8 | | 0 | | 8 | | 0 | 0,81 |
| 18, 20 | 2 | 31 | 32 | 33 | 34 | 1 | | 0 | | 1 | | 0 | 0,92 |
| 19, 20 | 4 | 30 | 30 | 34 | 34 | 0 | | 0 | | 0 | | 0 | 1 |
| 20, 21 | 2 | 34 | 34 | 36 | 36 | 0 | | 0 | | 0 | | 0 | 1 |
| 21, 22 | 2 | 36 | 36 | 38 | 38 | 0 | | 0 | | 0 | | 0 | 1 |
| 22, 23 | 2 | 38 | 38 | 40 | 40 | 0 | | 0 | | 0 | | 0 | 1 |

## Технико-экономические показатели

Чтобы определить экономический эффект от внедрения разрабатываемой ПС, необходимо сравнить суммарные затраты на разработку и использование данной системы и затраты при отказе от разработки. Если затраты при отказе от разработки будут превышать затраты на разработку и использование программной системы, то целесообразно приступать к созданию данной ПС.

Производственные затраты на разработку данного проекта рассчитываются по статьям[7]:

- Учет амортизации;

-расходы на заработную плату;

-основные материалы;

-затраты на обслуживание и ремонт техники;

-налоги, относимые на себестоимость.

### Учет амортизации

Возмещение стоимости техники рассчитаем по формуле:

**C = S \* N \* 1/ Q** (11.14) , где

N – норма амортизационных отчислений;

S – балансовая стоимость техники;

Q – количество выполняемых заказов.

Норма амортизационных отчислений вычисляется по формуле:

N = 1/ T \* 100 % , где (11.15)

N – норма амортизационных отчислений;

Т – время полезного использования.

Срок полезного использования техники примем в расчете – 2,5 года.

Техника модернизируется или заменяется по истечении этого срока, из-за устаревания оборудования. Таким образом, норма амортизационных отчислений, учитывая наши данные, составит:

N = 1/ 2,5 \* 100% = 40%.

Балансовая стоимость используемой техники – это первоначальная ее стоимость. На момент разработки данного проекта балансовая стоимость IBM PC составляет 25 тыс.руб.

Примем количество выполненных в год заказов, равноценных по времени и выполняемых на данном компьютере ориентировочно равным 25.

Следовательно, сумма возмещения стоимости техники на заказ составит:

С = 25000 \* 40% \* 1/25 = 400 руб.

### 11.3.2 Расходы на заработную плату исполнителей проекта

Сумма оклада соответствует месячному окладу руководителя отдела АСУ, инженера-программиста третьей категории на предприятии Троицкая ГРЭС на период разработки программного продукта.

Здесь учитываются отчисления во внебюджетные фонды, такие как: отчисления на единый социальный налог (ЕСС), величина которых составляет 35,6% и отчисления в специальные фонды (травматизм), величина которых на предприятии Троицкая ГРЭС составляет 0,4%. В общей сложности получается 36%.

Расчет затрат на заработную плату исполнителей проекта приведен в табл. 11.5.

Таблица 11.5 Заработная плата исполнителей проекта, руб*.*

|  |  |  |  |
| --- | --- | --- | --- |
| Исполнители проекта | Оклад, руб./мес | Трудоём-кость, дн. | Основная зарплата,  Руб |
| Руководитель проекта | 3500 | 29 | 4500 |
| Главный специалист | 2500 | 40 | 3000 |
| Программист | 1500 | 10 | 2500 |
| Консультант | 1000 | 10 | 1000 |
| Итого: | - | - | 11000 |
| Отчисления во внебюджетные фонды 36%: | - | - | 3960 |
| Итого с отчислениями во внебюджетные фонды: |  |  | 14960 |

### Затраты на разработку программной системы аттестации ИТ-специалистов

Расчёт релевантных затрат на разработку проекта приведён в таблице 11.9.

Таблица 11.9 Расчёт релевантных затрат на разработку программной системы

|  |  |  |
| --- | --- | --- |
| N п/п | Статьи затрат | Сумма, руб. |
| 1 | Возмещение стоимости ВТ | 400 |
| 2 | Основная заработная плата | 11000 |
| 3 | Отчисления во внебюджетные фонды | 3960 |
|  | **Итого** | 15360 |

### Целесообразность использования данного программного продукта

Существующие преимущества от внедрения ПС можно подразделить на количественные и качественные, рассмотрим каждые из них подробнее.

### Анализ качественных преимуществ

Таблица 11.10 Качественные преимущества

|  |  |
| --- | --- |
| **Программная система внедряется** | **Программная система не внедряется** |
| Исключается влияние субъективного мнения на результат аттестации | Возможно влияние субъективного мнения на результат аттестации |
| Появляется возможность быстрой оценки уровня подготовки специалистов при приеме на работу | Потеря времени на проведение аттестации в обычном режиме |
| Появляется возможность проводить аттестацию чаще и качественнее, не отрывая специалистов от основной работы | Невозможно проводить аттеста-цию чаще, в связи с длительностью процесса и необходимостью отвлекать специалистов от основной работы |
| Аттестация проходит прямо на рабочем месте специалиста, без участия комиссии | Необходимо собрать комиссию для проведения и оценки результатов аттестации |
| Оперативное получение результатов аттестации | Потеря времени на оценку результатов комиссией |

### 11.4.2. Оценка эффективности приминения системы ИТ-

### тестирования на предприятии

Поскольку для проведения аттестации в обычном порядке, необходимо собрать комиссию, отвлекая тем самым членов комиссии от основной работы, затраты на оплату аттестации за период один год без приминения разработанной системы составляют порядка 21000 рублей

При внедрении программной системы больше нет необходимости собирать комиссию для проведения аттестации и оценки результатов. Исходя из этого факта расчитаем экономический эффект от внедрения программной системы аттестации ИТ-специалистов, за полезный срок службы программной системы из расчета устареваемости и износа **,**принятый равным 5 лет.

Таблица 11.12 Расчёт экономического эффекта от внедрения программной системы аттестации ИТ-специалистов

|  |  |
| --- | --- |
| **З/п комиссии на проведение аттестации, руб.** | **21000** |
| **Релевантные затраты на разработку программной системы, руб.** | **15360** |
| **Экономический эффект, руб.** | **5640** |

## Вывод

Исходя из выше приведённых расчётов, делаем вывод – разработка и внедрение программной системы аттестации ИТ-специалистов является экономически выгодной. Кроме того, в расчётах не был учтён тот факт, что при внедрении этой программной системы, специалисты не прерывают своей текущей работы. аттестация проводится прямо на их рабочих местах в любое удобное им время, что несомненно положительно повлияет на работу предприятия в целом. Помимо этого, происходит значительная экономия времени и человеческих ресурсов на анализ результатов аттестации.

# 12 Безопасность жизнедеятельности

## 12.1 Планировка помещения

Проведем анализ потенциально опасных и вредных производственных факторов на рабочем месте пользователя системы.

Планировка помещения и размещение рабочих мест показаны на рисунке 12.1

**Юг**

Шкаф

**Запад**

Стол

Рабочее

место

**Восток**

Стол

Дверной

проем

Стол

#### Оконный проем

### 

##### Север

Рис 12.1 Схема помещения

Размеры помещения – 6 х 10 м, высота помещения – 3 м. На высоте 0,8 м расположены 2 оконных проема размерами 4 х 2 м с деревянными двойными рамами, застекленными оконным листовым стеклом.

Потолок побелен известью белого цвета, стены оклеены обоями светлых тонов, пол покрыт березовым паркетом.

## Требования охраны труда к помещениям

Размеры помещения (площадь, объем) должны в первую очередь соответствовать количеству работающих и размещенному в них комплексу технических средств. Для обеспечения нормальных условий труда санитарные нормы устанавливают на одного работающего объем производственного помещения не менее 30 м3 , а площадь помещения не менее 3 м2  на человека

с учетом максимального числа одновременно работающих в смену [10].

Так как площадь рассматриваемого помещения составляет 60 м2, объем помещения - 180 м3, а максимальное число одновременно работающих(тестируемых специалистов) - 3 человека, то на одного работающего приходится площадь 20 м2 и объем производственного помещения - 60 м3. Эти значения соответствуют требуемым параметрам.[11]

## 12.3.Условия труда на рабочем месте

Условия среды помещений определяются действующими на организм человека сочетаниями температуры, влажности и скорости движения воздуха и теплового облучения.

Допустимые параметры, определяющие условия труда на рабочем месте, могут вызывать переходящие и быстро нормализующиеся изменения функционального и теплового состояния организма и напряжение реакции терморегуляции, не выходящие за пределы физиологической приспособленности организма, не создающие нарушений состояния здоровья, но вызывающие дискомфортные ощущения, ухудшение самочувствия и снижение работоспособности.

Концентрация пыли в воздухе составляет не более 0,5 мг/м3.

Условия труда на рабочем месте регламентирует ГОСТ 12.1.005-88, который определяет оптимальные и допустимые параметры для рабочей зоны производственных помещений (то есть для пространства высотой до 2 м над уровнем пола).

Выполняемые на рабочем месте работы относятся к категории легких физических с затратой энергии до 120 Ккал/ч (категория I), а рассматриваемое помещение - к помещениям с незначительными избытками явной теплоты (до 23 Вт/м2 ). Оптимальные параметры микроклимата приведены в таблице 7.1.

**Таблица 12.1** Оптимальные нормы температуры, влажности и скорости движения воздуха в рабочей зоне производственных помещений

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Период года** | **Категория работ** | **Температура, °С** | **Влажность, %** | **Скорость воздуха, м/с не более** |
| Холодный | I | 21-24 | 60-40 | 0,1 |
| Теплый | I | 22-25 | 60-40 | 0,1; 0,2 |

Для обеспечения микроклиматических условий труда в помещении имеется система отопления и вентиляции ,что обеспечивает поддержание оптимальных условий труда на рабочем месте.

## 12.4. Освещение рабочих мест

Свет имеет исключительно важное значение для человека, обеспечивая связь организма с окружающей средой.

### Расчет естественного освещения

В рассматриваемом помещении используется одностороннее естественное фронтальное освещение, осуществляемое через два окна общей площадью 16 м2. Считается, что при работе с дисплеями площадь световых проемов в помещении должна составлять 25% площади пола. Площадь помещения 60 м2, значит, площадь световых проемов должна составлять 15 м2. Отсюда следует, что естественное освещение соответствует условиям труда. При недостатке естественного освещения используется искусственное освещение, которое осуществляется с помощью осветительных приборов общего назначения.

### Расчет искусственного освещения

Для расчета общего равномерного освещения при горизонтальной рабочей поверхности, основным является метод светового потока (коэффициента использования), учитывающий световой поток, отраженный от потолка и стен. Световой поток лампы Фл (лм) при лампах накаливания или световой поток группы ламп светильника при люминесцентных лампах рассчитывают по формуле:

Фл=100ЕнSzκ/(Nη)

где:

Ен - нормированная минимальная освещенность, лк

Для нашего помещения равно 100лк;

S - площадь освещаемого помещения, м2;

z - коэффициент минимальной освещенности, равный отношению Еср/ *Emin,*

значение которого для ламп накаливания равно 1,15;

κ *-* коэффициент запаса, значение которого для нашего помещения равно 1,3;

N - число светильников в помещении, в нашем помещении 4;

z - коэффициент использования светового потока ламп, зависящий от КПД и

кривой распределения силы света светильника, коэффициента

отражения потолка ρп и стен ρс, высоты подвеса светильников и показателя помещения i.

Значение коэффициента использования светового потока определяют по таблице . Показатель помещения рассчитывают по следующей формуле:

i=AB/Hp(A+B)

где:

*А* -длина помещения, м;

*В -* ширина помещения, м;

Hp *-* высота светильников над рабочей поверхностью, м.

**Таблица 12.2** Коэффициент использования светового потока

Светильник Астра», УПМ-15 УПД НСП-07

ρп, % 30 50 70 30 50 70 30 50 70

ρс,% 10 30 50 10 30 50 10 30 50

i Коэффициент использования η

0,5 17 21 25 21 24 28 14 16 22

0,6 23 27 31 25 28 34 19 21 27

0,7 30 34 39 29 39 38 23 24 29

0,8 34 38 44 33 36 42 25 26 33

0,9 37 41 47 38 40 44 27 29 35

1,0 39 43 49 40 42 47 29 31 37

1,5 41 50 55 46 51 57 34 37 44

2,0 51 55 60 54 58 62 38 41 48

3,0 58 62 66 61 64 67 44 47 54

4,0 62 66 70 64 67 70 46 50 59

5,0 64 69 73 66 69 72 48 52 61

Для нашего помещения выбираем светильник «Астра» - шар молочного стекла.

Так как потолок и стены у нас белые принимаем коэффициенты отражения 70 и 50 соответственно.

i=6 х 10 / *1*,8 х (6 + 10)=2.08 ≈ 2.0

*η=60%*

Фл=100х100х60х1,15х1,3/(10\*60)=1495лм

Из таблицы 12.3 подбираем подходящую лампу.

**Таблица 12.3** Световые и электрические параметры ламп накаливания

по (ГОСТ 3239-79)

|  |  |  |
| --- | --- | --- |
| **Тип** | **Световой поток,**  **лм** | **Световая отдача,**  **лм/Вт** |
| В-125-135-15 | 135 | 9,0 |
| В 215-225-15 | 105 | 7,0 |
| Б 125-135-40 | 485 | 12,0 |
| Б 220-230-40 | 460 | 11,5 |
| БК 125-135-100 | 1630 | 16,3 |
| БК 215-225-100 | 1450 | 14,5 |
| Г 125-135-1000 | 19 100 | 19,1 |
| Г 215-225-1000 | 19 600 | 18,6 |

Подбираем ближайшую стандартную лампу с напряжением питания 215-225 В. На практике допускается отклонение потока выбранной лампы от расчетного в пределах -10% . . . +20%, в противном случае выбирают другую схему расположения светильников.

В нашем случае условиям принятой нами схемы расположения светильников удовлетворяют лампы БК 215-225-100.

Планировка помещения и размещение осветительных приборов показаны на рис. 12.2.

Светильники

Бк 215-225-100

**Рис 12.2** Схема осветительных приборов

Высота потолка и размещение светильников обеспечивают угол падения света, исключающий отраженную блескость; отсутствие прямой блескости обеспечивается оптимальным взаимным расположением рабочих мест и индивидуальных светильников. Кроме того, окраска мебели и цвет обоев исключает интенсивное отражение света.

Таким образом, освещение рабочего места соответствует нормам.

## Анализ воздействия электромагнитных излучений

Основным источником различного вида излучений на рабочем месте являются мониторы. Их спектр излучения включает в себя рентгеновскую, ультрафиолетовую и инфракрасную области излучений, а также широкий диапазон электромагнитных волн более низких частот. Из вышеперечисленных излучений наиболее опасно рентгеновское, которое обладает большой проницаемостью.

На сегодня считается, что кратковременное и длительное воздействие всех видов излучений мониторов, особенно при наличии защитных экранов, не представляют опасности для здоровья оператора.

Рекомендуется применение мониторов, удовлетворяющих стандарту безопасности MPR II.

Максимальная напряженность на кожухе монитора Samsung SyncMaster 510s, который соответствует стандарту MPR II, составляет по паспортным данным 3,6 В/м, что соответствует фоновому уровню.

Интенсивность электромагнитного излучения в 5 см от экрана составляет 64 В/м, но на расстоянии 30 см, не превышает 2,4 В/м, что ниже, чем допустимый уровень. Это же можно сказать и об интенсивности ультрафиолетового и инфракрасного излучения.

Таким образом, при работе на настоянии 40 - 50 см от экрана дисплея вредное воздействие исключено.

## Анализ электробезопасности на рабочем месте

Электрический ток, действуя на организм человека, может вызвать нарушения его деятельности, вплоть до летального исхода. Тяжесть поражения определяется величиной протекающего через тело человека тока, частотой тока, длительностью протекания и другими факторами. Значения предельно допустимых уровней напряжения и токов устанавливаются ГОСТ 12.1.038-82.

Рассматриваемое помещение относится к классу помещений без повышенной опасности поражения электрическим током, так как в данном помещении отсутствуют признаки повышенной или особой опасности (влажности, проводящей пыли, токоведущих оснований (металлических, земляных), повышенной температуры (длительное превышение 35 °С или кратковременное превышение 40°С) и т.д.

Возникновению вышеперечисленных факторов препятствуют:

* соблюдение требований охраны труда на рабочем месте;
* характер работ в помещении;
* использование в качестве покрытия пола дерева.

Применяемая техника (компьютеры IBM PC и принтеры HP Laser Jet 1100) относятся к электроустановкам напряжением до 1000 В, питание которых осуществляется от сети однофазного переменного тока напряжением 220 В, частотой 50 Гц. Напряжение подается через автоматический выключатель с силовым расцепителем, имеющим ток срабатывания 25 А. В мониторах компьютеров имеются и более высокие напряжения (до 25 кВ ), которые надежно изолированы от оператора и не могут представлять опасности, доступ к цепям с таким напряжением возможен лишь при проведении ремонтных работ, которые осуществляются в специализированных ремонтных организациях специально подготовленным персоналом.

Устранить опасность поражения током при замыкании на корпус можно с помощью установки в помещении нулевого защитного проводника .

Занулением называется преднамеренное электрическое соединение с нулевым защитным проводником металлических нетоковедущих частей оборудования, которые могут оказаться под напряжением.

Нулевым защитным проводником называется проводник, соединяющий зануляемые части с глухозаземленной нейтральной точки источника питания или ее эквивалентом. Назначение нулевого защитного проводника – создание для тока короткого замыкания цепи с малым сопротивлением, чтобы этот ток был достаточным для быстрого срабатывания защиты, то есть быстрого отключения поврежденной установки от сети.

Кроме того, на рабочем месте оператора ЭВМ необходимо обеспечить защиту от статического электричества.

В вычислительных центрах разрядные токи статического электричества чаще всего возникают при прикосновении обслуживающего персонала к любому из элементов ЭВМ. Такие разряды могут привести к выходу из строя ЭВМ. Они оказывают неблагоприятное воздействие на работающих, ухудшают условия труда.

Для снижения величины возникающих разрядов статического электричества в ВЦ, покрытие технологических полов следует выполнять из однослойного антистатического линолеума марки АСН. К общим мерам защиты от статического

электричества в ВЦ можно отнести увлажнение воздуха (до 75%), ионизацию воздуха. Данные защитные меры регламентирует ГОСТ 12.4.124-83.

Из всего вышесказанного можно сделать вывод о том, что помещение удовлетворяет условиям обеспечения электробезопасности.

## Пожарная профилактика.

Пожар - это неконтролируемое горение, развивающееся во времени и пространстве. Понятие пожарной безопасности означает состояние объекта, при котором с установленной вероятностью исключается возможность возникновения и развития пожара с воздействием на людей опасных факторов, а также обеспечивается защита материальных ценностей. Опасными факторами пожара для людей являются:

• открытый огонь;

• повышенные температуры воздуха и предметов;

• токсичные продукты горения;

• дым;

• пониженная концентрация кислорода;

• взрыв и т. д.

Все помещения по пожарной и взрывоопасности делятся на пять категорий:

• А, Б - взрывопожароопасные;

• В, Г, Д - пожароопасные.

Данное помещение относится к категории В - горючие и трудно горючие помещения, в которых в обращении имеются жидкости, твердые, горючие и трудно горючие вещества и материалы, способные при взаимодействии с кислородом воздуха или друг другом только гореть, при условии что помещения, в которых они находятся, не относятся к категориям А и Б.

При подходе к обеспечению пожарной безопасности особое внимание следует уделять наличию токоведущей проводки в помещении. Данная проводка должна быть выполнена согласно требований ГОСТ 9098-59, данное требование соблюдения условий пожарной безопасности вызвано тем фактом, что согласно статистическим данным 85% всех пожаров происходит именно по причине не качественно выполненной проводки.

Противопожарная профилактика:

• Наличие двух ручных порошковых огнетушителей, марки ОП-2М, предназначенных для тушения загорания с расстояния 2 м, при температуре 40-50 °С. Огнетушители хранятся в защищенном от солнечных лучей и нагревательных приборов месте, хорошо доступном при возникновении возгорания;

• Для отопления помещений используется центральное водяное отопление;

• Установлена система электрической пожарной сигнализации, два тепловых пожарных излучателя реагируют на повышение температуры окружающей среды до значения 80°С и выше в радиусе 3 м. Для помещений вычислительных центров рекомендуется использовать тепловые пожарные излучатели типа ДТЛ, АТП-ЗМ и др.;

• Помещение расположено в здании таким образом, что имеется как минимум два пути эвакуации, один из которых не может быть перекрыт огнем - пожарная лестница.

## Анализ шума на рабочем месте

Шум - любой нежелательный для человека звук. Сильный шум в условиях производства снижает производительность труда до 40 - 60% и может явиться причиной несчастного случая.

Согласно ГОСТ 12.1.001-83 нормируемой шумовой характеристикой рабочих мест при постоянном шуме является уровень звукового давления в октавных полосах ,выраженный в децибелах. Совокупность таких уровней называется предельным спектром ПС, номер которого численно равен уровню звукового давления в октавной полосе со среднегеометрической частотой 1000 Гц.

В помещении установлено 3 компьютера типа IBM PC и 1 лазерный принтер. Уровень шума, издаваемый компьютером, составляет около 10 дБА, издаваемый принтером - около 15 дБА. Принтер является источником «механического» шума, вызванного бумагоподающим механизмом. Компьютер генерирует в основном аэродинамический шум, вызванный движением воздуха в системе охлаждения машины. Суммарный уровень шума, получаемый в результате наложения звуковых волн друг на друга, в рассматриваемом помещении составит 45 дБА, что ниже минимального уровня 50 дБА по ГОСТ 20.445-75.

## Эргономические требования

Персональный компьютер типа IBM PC спроектирован с учетом необходимых для работы эргономических требований:

• имеется возможность поворота дисплея в горизонтальной и вертикальной плоскостях.

• нет жесткой связи клавиатуры с дисплеем.

• существует возможность регулировки яркости и контрастности на дисплее.

• обеспечивается удобное расположение кнопок на панели компьютера.

• клавиатура обеспечивает удобный угол наклона к поверхности стола – 150.

• мягкость нажатий клавиш на клавиатуре.

• принтер HP 1100 также удовлетворяет ряду эргономических требований:

• красивое оформление и окраска мягкого цвета.

• удобное расположение кнопок управления.

• простота и удобство смены картриджа;

• небольшие мускульные усилия при работе.

• Рабочие места оборудованы легкими стульями, что соответствует требованиям ГОСТ 12.2.031-78. Высота сиденья не регулируется, что допускается нормами.

• Рабочий стол оператора имеет габариты: длина 1,25 м; ширина 0,7 м;

высота 0,8 м, что соответствует требованиям ГОСТ 12.2.031-78. Высота поверхности не регулируется, что также допускается нормами.

Вышеперечисленные характеристики обеспечивают минимальные затраты мускульной и нервной энергии оператора.

Также обеспечивается рациональный режим труда и отдыха, установленный с учетом психофизической напряженности труда, динамики функционального состояния систем организма и работоспособности, кроме того предусматривается строгое соблюдение регламентированных перерывов.

Для поддержания нормальной работоспособности работников рекомендуется продолжительность работы с монитором не более 50% рабочего времени, при этом время непрерывной работы не более 1,5-2 часов; время перерыва -15 мин.; и также время перерыв на обед - 40 мин. Все вышеперечисленные мероприятия по режиму работы и отдыха соблюдаются.

Рациональное цветовое оформление помещения направлено на улучшение санитарно-гигиенических условий труда, повышения его производительности и безопасности. Окраска производственных помещений влияет на нервную систему человека, его настроение, а также играет важную роль при организации системы освещения. Окраска стен не раздражает глаз и гармонирует с цветом технических средств.

# 13 Заключение

В ходе работы над проектом была создана программная система аттестации ИТ – специалистов, которая позволила автоматизировать и упростить процесс проведения аттестации. Дана методика обработки результатов аттест ации с использованием системы, рассмотрены методы защиты информации от несанкционированного доступа и от потери данных. Рассчитан экономический эффект от внедрения этой системы, который составит порядка 5640 руб.,поэтому если учесть что система реально может функционировать в течении 5 лет без дополнительных расходов не привышающих допустимые,то за полезный срок эксплуатации в течении этих лет прибыль составит 28200 рублей,что подтверждает экономическую обоснованность внедрения данной системы.

Со временем созданная система будет развиваться и совершенствоваться, будут добавлены новые критерии оценки результатов аттестации, и усовершенствованна логика и алгоритмы её работы.

# Список литературы

1. Оценка и аттестация персонала http://www.cpt21.ru/ocenka.htm
2. Сертификация Microsoft Office User Specialist http://www.specialist.ru/MOUS/
3. Аттестация – назначение и методы организации http://www.karjera.lv
4. Оценка и аттестация персонала http://www.tarasov.ru/Oz\_att.htm
5. Дейт К. Руководство по реляционной СУБД DB2. - М.: Финансы и статистика, 1988. - 320 с.
6. Дейт К. Введение в системы баз данных //6-издание. - Киев: Диалектика, 1998. - 784 с
7. Организационно – экономический раздел дипломного проекта конструкторского направления. Учебное пособие для ПС: - ЧГТУ, 1999

8. Лилия Козленко "Проектирование информационных систем"

9. Зиндер Е.З.Соотнесение и использование стандартов организации

жизненного цикла систем.//СУБД, №3, 1997,

10.Безопасность жизнедеятельности /Под общей редакцией С. В. Белого/. – М.:

Высшая школа, - 1999

11.СанПиН 2.2.2 542-96.Гигиенические требования к видеодисплейным

терминалам ПЭВМ

# 14 Руководство системного администратора

Компонент системы «Аттестация ИТ-специалистов», предназначен для использования администратором системы. Файл запуска - ita\_adm.exe

Минимальные системные требования

* IBM-совместимый компьютер, не ниже:
* Pentium II-450, RAM-64Mb, SVGA-800\*600\*16bit
* Свободное пространство на жёстком диске не менее 5Мб.
* Операционная система Windows 95/98/ME или Windows NT 4.0/5.0

Все компоненты снабжены однотипной настройкой подключения к серверу баз данных. Окно настройки представлено на рис 14.1.

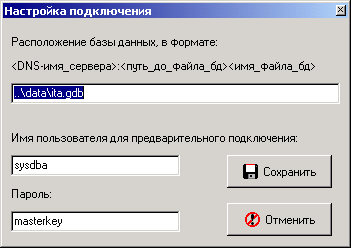
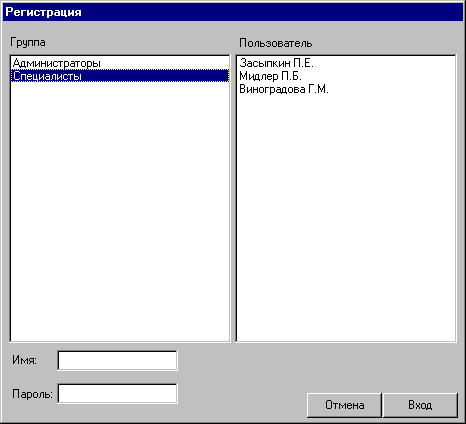


Рис. 14.1 Окно настройки подключения

Настроив подключение к серверу можно регистрироваться в системе для этого необходимо ввести имя и пароль ,форма изображена на рисунке 14.2



Вход в систему

Рис 14.2 Регистрация в системе

Настройка системы:

Настройки необходимые для первоначального подключения к системе хранятся в файле ita.ini который должен находится в той же папке, что и файл запуска компонеты системы. Формат этого файла следующий:

[defaults]

datasource=base.gdb

pusername=SYSDBA

ppassword=masterkey

где параметр datasource – означает расположение базы данных системы в стандартной форме interbase, например база данных располагается на Unix сервере server.domain.com, находится в каталоге /data/base и файл называется ita.gdb, в этом случае в параметр datasource необходимо записать следующее:

datasource=server.domain.com:/data/base/ita.gdb

Параметры pusername и ppassowrd – задают имя и пароль пользователя для предварительного подключения к серверу interbase.

Если на момент запуска любого из компонент системы, файл ita.ini не будет обнаружен, этот файл будет создан с параметрами по умолчанию.

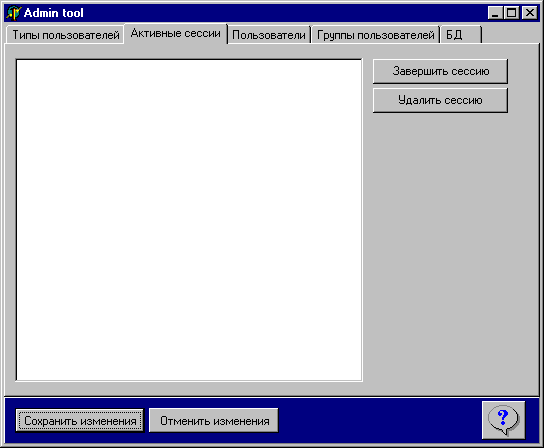
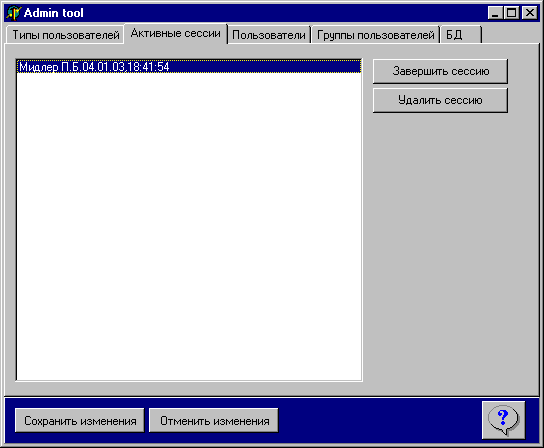
Если по каким либо причинам предварительное подключение к серверу вызовет ошибку, пользователь будет об этом информирован и ему будет предложено выполнить настройку параметров подключения. Такой порядок позволяет избежать "ручного" редактирования файла конфигурации при первом запуске компонентов системы.

В случае успешного предварительного подключения пользователю будет предложено зарегистрироваться в системе. Для этого необходимо выбрать группу, к которой принадлежит, имя пользователя и ввести пароль.

Выбрав группу и имя пользователя, необходимо ввести пароль и нажать кнопку «Вход в систему». Если пароль введён правильно, открывается главное

Рис 14.3 Главное окно администратора системы

окно администратора показанное на рисунке 14.3.

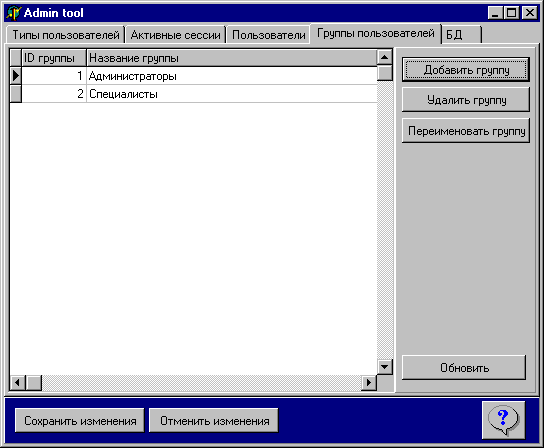


Возможности:

* Создание типа пользователя ( от типа пользователя зависит доступ к каким данным открыт пользователю, связанным с этим типом ) Рис. 14.4
* Удаление типа пользователя (Рис.14.5)
* Создание пользователя и связь его с типом
* Удаление пользователя
* Завершение сессии аттестации ( с автоматическим подсчётом результатов )
* Удаление сессии аттестации ( с потерей данных аттестации ) Рис. 14.3

Рис 14.5 Управление типами пользователей

Рис 10.5 Управление группами пользователей



**Приложение 1**

# Исходный текст программного комплекса:

За исходниками и базой данных Interbase обращаться на yuta@mailru.com или http://yuta.mailru.com

Плакаты формата А1:

****************************************