

# **Методические рекомендации**

## **по выполнению лабораторных работ по предмету**

### **СУБД**

#### **для студентов специальности «ЗИ»**

Автор: Арбатский Е.В.

Курс СУБД рассчитан на то, чтобы студенты изучили основные вопросы, связанные с проектированием и разработкой реляционных баз данных, имели представление об основных моделях данных, знали принципы функционирования систем управления базами данных и систем, основанных на работе с базами данных.

В рамках этого курса студенты на практике осваивают язык SQL, работу с Oracle, принципы построения ER диаграмм и интерфейсов к базам данных.

Весь процесс практического изучения разбит на 10 лабораторных работ. Причем последняя работа не является обязательной для выполнения. Она разработана для освоения принципов тестирования производительности баз данных.

Список лабораторных работ:

1. Описание предметной области;
2. ER-диаграммы;
3. SQL. Создание БД;
4. SQL. Работа с простыми запросами;
5. SQL. Построение сложных запросов;
6. SQL. Обеспечение ссылочной целостности;
7. SQL. Создание триггеров и функций;
8. Разработка интерфейса. QBE. Delphi. MS Access. OpenOffice.Base;
9. SQL. Использование представлений;
10. SQL. Тестирование производительности.

### **Лабораторная работа № 1**

В рамках курса студентам требуется выполнить курсовую работу. В состав курсовой работы входит:

- Выделение и изучение предметной области;
- Описание предметной области;
- Построение ER-диаграммы;
- Построение схемы базы данных;

- Проектирование и разработка интерфейса к БД;
- Создание отчета по курсовой работе.

Для освоения студентами навыков совместной работы и разработки многопользовательских приложений одна тема курсовой работы выдается на группу из двух человек (если в группе нечетное число студентов, то рекомендуется одну команду создать из трех человек и в дальнейшем увеличить для них детализацию предметной области, в крайнем случае выдать задание для одного человека). Для выполнения курсовой работы и лабораторных работ группе студентов предоставляется уникальная учетная запись для командного доступа к БД Oracle.

На первой лабораторной работе парам студентам предоставляется список возможных тем курсовых работ из которого они выбирают одну тему. Выбранная тема вычеркивается из списка доступных работ для других студентов. Тем самым обеспечивается обособленность работы команд студентов над своими базами данных.

С учетом того, что на первой лабораторной работе у студентов еще нет представления о том, как следует формализовать предметную область, как работать с SQL сервером, то им ставится задача описать ту предметную область, которая относится к ним в рамках выбранной темы курсовой работы.

Для этого им следует выделить основные объекты в предметной области, о которых в дальнейшем следует хранить данные. Описать характеристики каждого объекта ( $5 < \text{количество характеристик} < 10$ ). Описать связи между объектам, способы их взаимодействия. Описать какая информация может потребоваться о совокупности данных объектов и их функционировании. Описание производится в произвольной форме. Примерная структура отчета по лабораторной работе:

Фамилия Имя Отчество (группа)
Название курсовой работы
Описание предметной области и задач, для решения которых создается БД.
Выявленные базовые объекты:
1. Название1
2. Название2
3. Название3
Описание объектов:

Наименование1.

Краткое описание объекта и перечень его характеристик.

Наименование2.

Краткое описание объекта и перечень его характеристик.

Наименование3.

Краткое описание объекта и перечень его характеристик.

Связи объектов:

Наименование1 и Наименование2.

Описание связи.

Наименование3 и Наименование1.

Описание связи.

Работа считается выполненной, если правильно определена предметная область, объекты, их характеристики и связи. Минимальное количество характеристик может снижено с учетом особенностей объектов.

На выполнение лабораторной работы отводится одно занятие.

## **Лабораторная работа № 2**

Вторая лабораторная работа является логическим продолжением первой работы. Студентам следует представить ранее описанную предметную область в виде ER-диаграммы. Для примера предоставляется два-три подхода к построению ER-диаграмм. Студенты самостоятельно выбирают наиболее подходящий вариант и реализуют свою диаграмму с учетом требований выбранного подхода.

На диаграмме следует указать все сущности, связи между ними и кардинальность связей. Изучить виды сущностей своей предметной области. Определить где:

- стержневые сущности;
- ассоциации;
- характеристики.

В ходе построения диаграммы могут выявиться недочеты в описании предметной области. В таком случае требуется внести коррективы в

описание – добавить новые объекты, удалить ненужные, добавить связи и так далее.

Работа считается выполненной, если правильно спроектирована ER-диаграмма и её содержимое соответствует описанной предметной области.

В качестве средств визуализации можно использовать: IBM Rational Data Architect, Erwin, Microsoft Visio 2002(и выше) и подобные инструменты. Если специализированных инструментов нет, тогда возможно использовать графические инструменты MS Word, OOo Writer.

На выполнение лабораторной работы отводится одно-два занятия.

### **Лабораторная работа № 3**

Целью третьей лабораторной работы является ознакомление с консольным режимом работы с SQL сервером. Данную работу рекомендуется выполнять командно, но каждый член команды должен быть за отдельным компьютером.

Шаги выполнения работы:

1. На каждую команду выдается один логин для подключения к БД Oracle. Каждый из студентов должен запустить приложение SQLPlus и осуществить процедуру входа согласно полученных регистрационных данных;
2. Следует создать учебные таблицы edu\_salers, edu\_customers, edu\_orders, edu\_test (их описание приведено далее);
3. Удалить таблицу edu\_test;
4. Занести по одной строке в учебные таблицы;
5. Просмотреть содержимое таблиц;
6. Отключиться от БД и подключиться снова и посмотреть содержимое таблиц;
7. COMMIT;
8. Создать таблицы для своей курсовой работы.

Описание структуры таблиц:

#### **edu\_salers**

snum	номер продавца
sname	имя продавца
city	город, местонахождение продавца
comm	комиссионные продавца в десятичной форме

### edu\_customers

cnum	номер покупателя
cname	имя покупателя
city	город
rating	рейтинг покупателя (чем выше, тем лучше)

### edu\_orders

onum	номер заказа
amt	сумма заказа
odate	дата заказа
cnum	номер покупателя
snum	номер продавца

### edu\_test

tnum	номер теста
tdate	дата теста
tres	результат теста

Пример создания таблицы:

```
CREATE TABLE edu_test (
tnumc INTEGER,
tdate DATE,
tres VARCHAR2(255) NOT NULL);
```

Для таблиц edu\_salers, edu\_customers, edu\_orders студенты должны самостоятельно определить типы данных и размеры полей. Так же они должны самостоятельно разработать команды создания таблиц с учетом тех типов данных, что есть в учебной СУБД. Для этого им предоставляется электронный документ с описанием всех типов данных, которые поддерживаются учебной СУБД (Oracle 9i).

Для удаления таблицы следует использовать команду DROP TABLE.

Формат команды:

```
DROP TABLE <имя_таблицы>;
```

Для добавления данных следует использовать команду INSERT. Формат команды:

```
INSERT INTO <имя_таблицы>[(<имя_поля>[,...])] VALUES(<значение>[,...]);
```

Для просмотра данных следует использовать команду SELECT. Формат команды:

```
SELECT *|<имя_поля> FROM <имя_таблицы>;
```

При работе с базой данных следует помнить о том, что:

- текстовые поля таблицы должны иметь размер, заданный в символах;
- текстовые значения выделяются символом «'»;
- в конце каждой команды следует ставить символ «;»;
- команды создания объектов БД приводят к мгновенному созданию объекта в БД. Если один пользователь БД создал объект, то другой пользователь уже не сможет создать объект с таким же названием;
- тип поля DATA / DATE / DATETIME является зависимым от конкретной СУБД, поэтому следует изучать команду работы с датами, которая представлена в описании типов данных для учебной СУБД;
- вещественные числа следует задавать типом NUMBER, у которого в учебной СУБД есть несколько параметров, объясненных в описании по типам данных.

Шестой шаг работы требуется для получения студентами ответа на вопрос вида: «Что будет с нашими данными завтра?». В случае отключенного параметра AUTOCOMMIT должна произойти утеря введенных данных. Но созданные объекты должны остаться в БД.

После этого шага ставится более понятным требование завершать работу с базой данных, фиксацию изменений в БД командой COMMIT. Так же студентам следует провести следующий эксперимент:

1. Студент А вносит запись в таблицу, скажем, edu\_salers;
2. Студент Б вносит запись в таблицу edu\_salers;
3. Студенты А и Б смотрят содержимое таблицы edu\_salers и каждый видит только свои данные;
4. Студент А фиксирует изменения командой COMMIT;
5. Студенты А и Б смотрят содержимое таблицы edu\_salers и студент А видит только свои данные, а студент Б видит свои данные и данные студента А;
6. Студент Б фиксирует изменения командой COMMIT;

7. Студенты А и Б смотрят содержимое таблицы edu\_salers и видят все данные.

Данный эксперимент позволяет лучше освоить особенности совместной работы с одной базой данных.

#### **Дополнения.**

- Студентам следует фиксировать все свои команды в отдельном текстовом файле, который пригодится им для дальнейших лабораторных работ, а так же позволит им сформировать свой набор запросов;
- Для удобства оперативного редактирования запросов в SQLPlus есть дополнительные команды:
  - edit - вызов режима редактирования последней SQL команды;
  - run - выполнение отредактированной SQL команды или же повторное выполнение последней SQL команды.

На выполнение лабораторной работы отводится одно занятие.

### **Лабораторная работа № 4**

Текст задания:

1. Вставьте в таблицы тестовые данные, приведенные в конце задания.
2. Напишите команду, которая бы поместила следующие значения, в их нижеуказанном порядке, в таблицу Продавцов:
  - city - San Jose,
  - name - Bianco,
  - comm - NULL,
  - snum - 1100.
3. Напишите команду, которая бы удалила все заказы покупателя Clemens из таблицы заказов.
4. Напишите команду, которая бы увеличила оценку всех покупателей в Риме на 100.
5. Напишите запрос, который может дать вам все заказы со значениями суммы выше чем \$1,000.
6. Напишите запрос, который может выдать вам поля sname и city для всех продавцов в Лондоне с комиссионными выше .10 .
7. Напишите запрос, который сосчитал бы все суммы приобретений на 3 Октября.
8. Напишите запрос, который сосчитал бы число различных не-NULL значений поля city в таблице Покупателей.
9. Напишите запрос, который выбрал бы заказы с наименьшей суммой

для каждого заказчика.

10. Напишите запрос, который бы выбирал покупателей в алфавитном порядке, чьи имена начинаются с буквы G.
11. Напишите запрос, который выбрал бы высшую оценку в каждом городе.
12. Предположим что каждый продавец имеет 12% комиссионных. Напишите запрос к таблице заказов, который мог бы вывести номер заказа, номер продавца, и сумму комиссионных продавца для этого заказа.
13. Напишите запрос к таблице Заказчиков который мог бы найти высшую оценку в каждом городе. Вывод должен быть в такой форме:  
For the city (city), the highest rating is: (rating).
14. Напишите запрос который выводил бы список заказчиков в нисходящем порядке. Вывод поля оценки(rating) должен сопровождаться именем заказчика и его номером.

### edu\_salers

SNUM	SNAME	CITY	COMM
1001	Peel	London	.12
1002	Serres	San Jose	.13
1004	Motika	London	.11
1007	Rifkin	Barcelona	.15
1003	Axelrod	New York	.10

### edu\_customers

CNUM	CNAME	CITY	RATING
2001	Hoffman	London	100
2002	Giovanni	Rome	200
2003	Liu	SanJose	200
2004	Grass	Berlin	300
2006	Clemens	London	100
2008	Cisneros	SanJose	300

2007	Pereira	Rome	100
------	---------	------	-----

### edu\_orders

ONUM	AMT	ODATE	CNUM	SNUM
3001	18.69	10.03.90	2008	1007
3003	767.19	10.03.90	2001	1001
3002	1900.10	10.03.90	2007	1004
3005	5160.45	10.03.90	2003	1002
3006	1098.16	10.03.90	2008	1007
3009	1713.23	10.04.90	2002	1003
3007	75.75	10.04.90	2004	1002
3008	4723.00	10.05.90	2006	1001
3010	1309.95	10.06.90	2004	1002
3011	9891.88	10.06.90	2006	1001

Для удаления данных из таблицы следует использовать команду DELETE. Формат команды:

```
DELETE FROM <имя_таблицы> [WHERE <условие>]
```

Для просмотра данных следует использовать команду SELECT. Формат команды:

```
SELECT *|<имя_поля>[,<имя поля>][,...] FROM <имя_таблицы> [WHERE <условие>];
```

Особенности выполнения работы:

- При выполнении первой задачи следует помнить о том, что текстовые значения следует выделять символом «'», а также об особенностях работы с датами;
- При выполнении второй задачи следует обратить внимание на то, что последовательность полей отличается от последовательности полей, заданной при создании таблицы, а значит надо использовать расширенный вариант команды INSERT;
- Для третьей задачи следует визуально определить код покупателя, а уже затем написать запрос на удаление его заказов;
- В четвертой задаче при командной работе над заданием студенты должны встретить ситуацию «зависания программы SQLPlus». На самом деле в данном случае происходит первое знакомство с понятием блокировки в СУБД. Для разрешения проблемы

«зависания» (удаления блокировки) следует выполнить команду COMMIT;

- При выполнении седьмой задачи зачастую выявляется проблема неправильной обработки дат;
- В восьмой задаче студентам следует уделять внимание каждому слову задания. Многие пропускают слово «различных»;
- Десятую задачу можно выполнить двумя способами (используя LIKE или BETWEEN). Студентам следует освоить оба способа;
- В тринадцатой задаче студенты сталкиваются с проблемой вывода в запросе дополнительной информации, которой нет в БД.

### **Лабораторная работа № 5**

Занести все тестовые данные, приведенные в 4-й работе.

1. Занести тестовые данные, приведенные в 3-й работе.
2. Напишите запрос который бы вывел список номеров заказов, сопровождающихся именем заказчика, который создавал эти заказы.
3. Напишите запрос, который бы выдавал имена продавца и заказчика для каждого заказа после номера заказа.
4. Напишите запрос, который бы выводил всех заказчиков обслуживаемых продавцом с комиссионными выше 12% . Выведите имя заказчика, имя продавца, и ставку комиссионных продавца.
5. Напишите запрос, который вычислил бы сумму комиссионных продавца для каждого заказа заказчика с оценкой выше 100.
6. Напишите запрос, который бы вывел все пары продавцов живущих в одном и том же городе. Исключите комбинации продавцов с ними же, а также дубликаты строк выводимых в обратном порядке.
7. Напишите запрос, который вывел бы имена(sname) и города(city) всех заказчиков с такой же оценкой(rating) как у Hoffmana. Напишите запрос, использующий поле snum Hoffmana, а не его оценку, так чтобы оно могло быть использовано, если его оценка вдруг изменится.
8. Напишите запрос, который бы использовал подзапрос для получения всех заказов для заказчика с именем Cisneros. Предположим, что вы не знаете номера этого заказчика, указываемого в поле snum.
9. Напишите запрос, который вывел бы имена и оценки всех заказчиков которые имеют усредненные заказы.
10. Напишите запрос, который бы выбрал общую сумму всех приобретений заказов для каждого продавца, у которого эта общая сумма больше, чем сумма наибольшего заказа в таблице.

11. Напишите запрос который бы использовал оператор EXISTS для извлечения всех продавцов, которые имеют заказчиков с оценкой 300.
12. Напишите запрос, использующий оператор EXISTS, который выберет всех продавцов с заказчиками, размещенными в их городах, которые ими не обслуживаются.
13. Напишите запрос, который извлекал бы из таблицы Заказчиков каждого заказчика, назначенного к продавцу, который в данный момент работает по крайней мере еще с одним заказчиком (кроме заказчика, которого вы выбираете).
14. Напишите запрос, который бы выбирал все заказы с суммой больше, чем любая (в обычном смысле) сумма заказа для заказчиков в Лондоне.

Особенности выполнения работы:

- В данной работе уже требуется использовать сложные запросы, в которых объединяются данные двух или более таблиц;
- Для объединения таблиц можно использовать вариант *FROM a, b WHERE a.id=b.id* или *FROM a JOIN b ON a.id=b.id*;
- Рекомендуется использовать алиасы (псевдонимы) для таблиц;
- В восьмой задаче под усредненными заказами понимаются заказы величина которых равна средней величине всех заказов от всех заказчиков;
- Результаты запросов должны содержать только те данные, которые указаны в задании.

### **Лабораторная работа № 6**

1. Занесите тестовые данные (рисунок 1) в БД.
2. Удалите поставщика с кодом = 1000.
3. Измените код поставщика 1003 на 1000.
4. Вставьте заказ для поставщика с кодом 1020 на сумму = 500.
5. Укажите в таблицах первичные ключи(PRIMARY KEY) и внешние ключи(FOREIGN KEY).
6. Снова занесите тестовые данные в БД.

## 7. Повторите п.2-4.

Поставщики				
1000	Иванов	Иркутск	0.10	
1001	Петров	Иркутск	0.12	
1002	Сидоров	Братск	0.08	
1003	Семенов	Улан-Удэ	NULL	

Покупатели				
2000	Федоров	Иркутск	100	
2001	Милов	Братск	110	
2002	Кузнецов	NULL	130	
2003	Синин	Ангарск	120	

Заказы				
3000	09/10/05	100	1000	2000
3001	10/10/05	200	1001	2003
3002	11/10/05	300	1002	0
3003	12/10/05	400	1003	NULL
3004	13/10/05	500	1002	2002

Особенности выполнения работы:

- В тексте задания названия таблиц с данными даны на русском, но сами названия таблиц в базе данных должно соответствовать текстовым названиями (edu\_salers и т.п.);
- В тексте задания явно не указаны поля для данных, студентам следует самостоятельно определить соответствие тех или иных столбцов нужным полям в базе данных;
- При создании первичных ключей или задании внешних ключей может (и должна) проявиться проблема наличия некорректных данных. Следует разрешить проблему.

### **Лабораторная работа № 7**

Создать последовательности для учебных таблиц.

Создать триггеры на вставку и обновление для учебных таблиц.

Эта работа ориентирована на то, чтобы дать студентам возможность освоить навыки работы со встроенными версиями языка SQL, а так же созданием первичных ключей с помощью триггеров.

## **Лабораторная работа № 8**

### MS Access

1. Создать таблицы
2. Указать связи между таблицами
3. Создать формы для заполнения таблиц
4. Создать отчет по поставщикам
5. Создать отчет по заказам, сгруппированным по дням
6. Напишите запрос, который бы вывел список номеров заказов, сопровождающихся именем заказчика, который создавал эти заказы.
7. Напишите запрос, который бы выдавал имена продавца и заказчика для каждого заказа после номера заказа. (использовать QBE)
8. Создать отчеты для созданных запросов.

### Delphi

1. Создать таблицы в Database Desktop (использовать форматы DBF/Paradox)
2. Создать формы для заполнения таблиц
3. Напишите запрос, который бы вывел список номеров заказов, сопровождающихся именем заказчика, который создавал эти заказы.
4. Напишите запрос, который бы выдавал имена продавца и заказчика для каждого заказа после номера заказа.
5. Создать отчеты для созданных запросов.

### OpenOffice Database

1. Создать таблицы
2. Указать связи между таблицами
3. Создать формы для заполнения таблиц
4. Создать отчет по поставщикам
5. Создать отчет по заказам, сгруппированным по дням
6. Напишите запрос, который бы вывел список номеров заказов, сопровождающихся именем заказчика, который создавал эти заказы.
7. Напишите запрос, который бы выдавал имена продавца и заказчика для каждого заказа после номера заказа. (использовать QBE)
8. Создать отчеты для созданных запросов.

## **Лабораторная работа № 9**

Создать представления:

1. Список поставщиков из Иркутска и London
2. Сумма и количество заказов на каждый день

Выполнить запросы:

1. Вставить в 1-ое представление поставщика из Иркутска (1110, Сидоров, Иркутск, 0.1).
2. Вставить в 1-ое представление поставщика из Братска (1112, Петренко, Братск, 0.2).
3. Вставить во 2-ое представление заказ на сумму 150.
4. Обновить название поставщика из Иркутска Сидорова на Симонова.
5. Изменить количество заказов на последнюю дату.
6. Создать представление по покупателям из Иркутска и London с опцией CHECK.
7. Вставить в представление покупателя из Иркутска (2110, Сидоров, Иркутск, 10).
8. Вставить в представление покупателя из Братска (2112, Петренко, Братск, 20).
9. Предоставить пользователю с другим логином доступ к первому представлению на чтение.
10. Получить от пользователя с другим логином доступ к первому представлению на чтение.
11. Получить данные из чужого представления.

Для создания представления следует использовать команду CREATE VIEW. Формат команды:

```
CREATE VIEW <имя_представления> AS <запрос>
```

Для предоставления доступа следует использовать команду GRANT. Формат команды:

```
GRANT PRIVILEGES ON <имя_объекта> TO <имя_пользователя>
```

Для снятия доступа следует использовать команду REVOKE. Формат команды:

```
REVOKE PRIVILEGES ON <имя_объекта> FROM <имя_пользователя>
```

## **Лабораторная работа № 10**

Тестирование производительности работы СУБД на простой базе данных:

1. Создать таблицу edu\_test с тремя полями id (NUMBER 10), name (VARCHAR 200), value (NUMBER 10).

2. Зафиксировать время начала.
3. Сгенерировать 10000 записей в БД (каждая запись должна сразу фиксироваться после добавления).
4. Зафиксировать время окончания.
5. Определить среднее время на запись одной записи.
6. Зафиксировать время начала.
7. Выполнить запрос: `SELECT COUNT(*) FROM edu_test`
8. Зафиксировать время окончания.
9. Определить время выполнения запроса.
10. Зафиксировать время начала.
11. Выполнить запрос: `SELECT SUM(value)/COUNT(*) FROM edu_test.`
12. Зафиксировать время окончания.
13. Определить время выполнения запроса.
14. Зафиксировать время начала.
15. Выполнить запрос: `SELECT COUNT(name) FROM edu_test WHERE name > 'I' ORDER BY name.`
16. Зафиксировать время окончания.
17. Определить время выполнения запроса.
18. Зафиксировать время начала.
19. Выполнить запрос: `SELECT COUNT(*) FROM edu_test a INNER JOIN edu_test b ON a.value=b.value WHERE name > 'I'.`
20. Зафиксировать время окончания.
21. Определить время выполнения запроса.
22. Повторить 2-6 шаги для 200000 и 500000 записей.
23. Повторить 2-й шаг для 10000, 200000 и 500000 записей, но фиксируя записи через 10тыс.
24. Построить график изменения скорости добавления записей / выполнения запросов.
25. Уничтожить и создать таблицу `edu_test` с тремя полями `id (NUMBER 10) PRIMARY KEY`, `name (VARCHAR 200) INDEX`, `value (NUMBER 10) INDEX`.
26. Повторить 2-6 шаги для 10000, 200000 и 500000 записей.
27. Дополнить график изменения скорости добавления записей / выполнения запросов.

Для решения этих задач следует использовать какой-либо алгоритмический язык. Для наглядности эксперименты лучше произвести на базе `mySQL` или `Dbase`.

## **Темы курсовых работ на 2009-2010 учебный год**

1. БД системы проверки олимпиадных заданий
2. БД пропускной системы
3. БД динамики характеристик информационной системы
4. БД системы доступа к персональным данным
5. БД социальной сети
6. БД рекламной сети (для показа рекламы на сайтах)
7. БД по помещениям и их планировке
8. БД органайзера для студента
9. БД продукционной базы знаний
10. БД системы формирования заявок на покупку ПК
11. БД учета использования компьютеров
12. БД учета лицензий на ПО
13. БД службы поддержки
14. БД учета кадров и доступа к сети

## Пример описания предметной области

Тема: «Информационная система активности локальной вычислительной сети»

### 2. ОСНОВНАЯ ЧАСТЬ

#### 2.1 Анализ предметной области

В основе любой локальной вычислительной сети лежит активное оборудование. Активным оборудованием называется оборудование, которое читает информацию из передаваемых данных (например, mac-адреса) и в зависимости от неё определяет свои дальнейшие действия.

Активное оборудование соединено либо с другим активным оборудованием, либо с неким конечным устройством. Таким образом, оборудование можно разделить на два вида: промежуточное и конечное. Активное оборудование может иметь одно и более соединений. Примерная структура сети имеет вид (рис. 2.1).

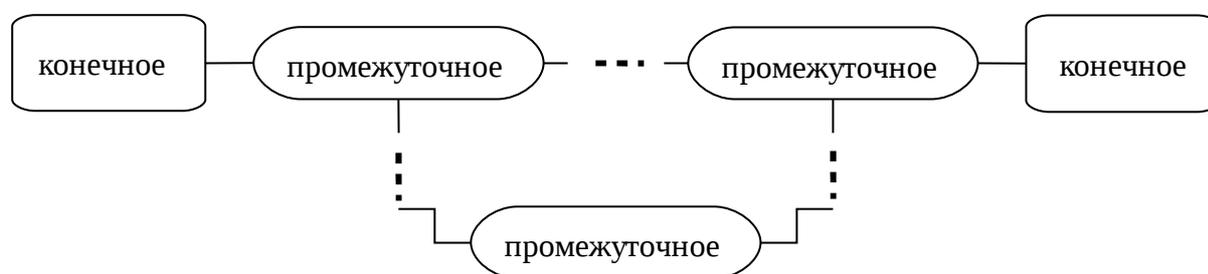


Рис. 2.1. Соединение видов устройств

Основной частью активного оборудования является порт – наименьшая и неделимая единица ЛВС. Порты принадлежат активному или конечному оборудованию и не могут существовать отдельно от него.

Активным оборудованием рассматриваемой локальной сети являются маршрутизаторы, концентраторы и коммутаторы, конечным –

персональные компьютеры. Между компьютерами может производиться обмен информацией. Параметры каждого обмена регистрируются на соответствующих портах. Кроме физического адреса (mac-адреса) персональные компьютеры имеют также адрес логический (ip-адрес), который и фиксируется при вычислении объёма исходящего и входящего трафика.

По таким принципам построена ЛВС, объединяющая отделы некой организации (рис. 2.2).

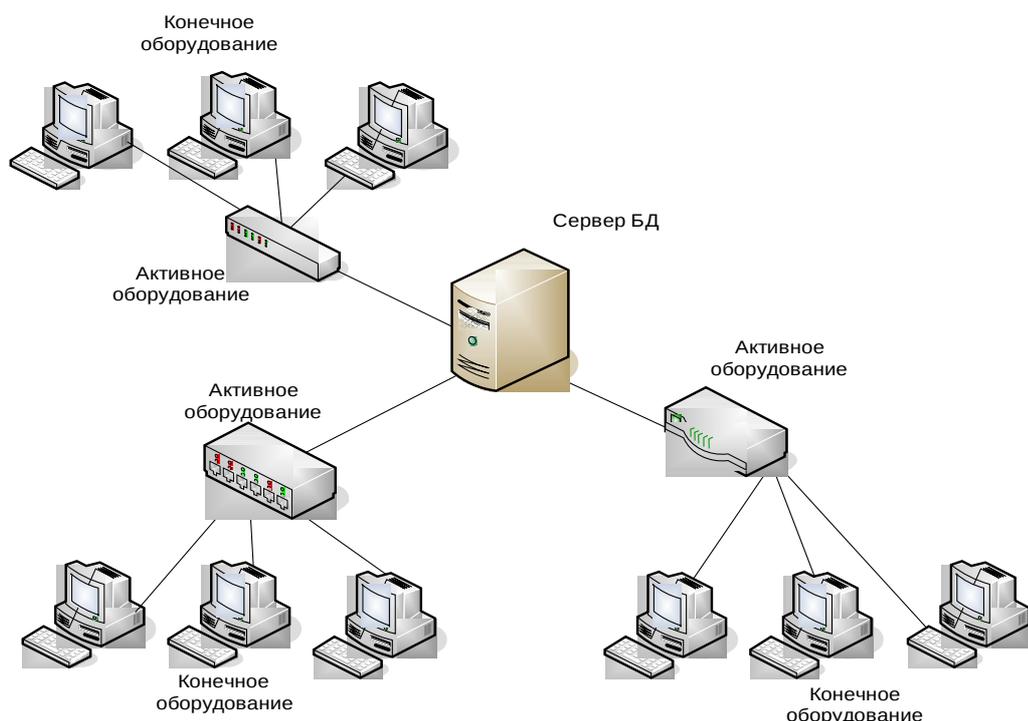


Рис. 2.2. Структура ЛВС организации

С помощью создаваемой информационной системы разработчики ЛВС хотят проверить, насколько эффективна рассматриваемая сеть, в какие периоды она наиболее активна и велика ли нагрузка на активное оборудование. Поэтому ИС должна хранить достоверную информацию об оборудовании (промежуточном и конечном), а также о потоках данных, передаваемых по сети.

На основании проведённого анализа выделяем следующие объекты:

- o активное оборудование
- o порты активного оборудования
- o трафик
- o конечное оборудование

Характеристики активного оборудования представлены в таблице 2.1.

Можно отметить следующие операции, производимые с активным оборудованием:

- установка нового оборудования
- замена оборудования с сохранением подключения
- изменение местоположения оборудования
- техническое обслуживание

Табл. 2.1. Основные характеристики активного оборудования

<i>№ n/n</i>	<i>Название характеристики</i>	<i>Описание</i>
1	Тип оборудования	Указывает на тип активного оборудования. Необходим для определения принципа работы устройства.
2	Производитель	Указывает на производителя активного оборудования. Имеет значение при организации технического сопровождения производителем выпущенного им оборудования.
3	Модель	Наименование модели. Необходимо для поиска однотипного оборудования.
4	Количество портов	Параметр, указывающий на число возможных подключений
5	Скорость	Скорость передачи данных
6	Дата установки	Необходима для определения срока службы устройства и замены его в случае морального или физического износа.

7	Инвентарный номер	Необходим для идентификации каждой единицы оборудования
---	-------------------	---

Порты активного оборудования характеризуются внутренним номером (т.е. порядковым номером внутри активного оборудования), а также устройством, частью которого они являются.

Основные операции, производимые техническим персоналом с портами активного оборудования:

- подключение порта
- отключение порта
- изменение адресата подключения
- поиск неисправных портов
- поиск свободных портов для новых подключений

Трафик регистрируется на портах активного оборудования. В таблице 2.2 приведены необходимые характеристики трафика сети.

Табл. 2.2 Основные характеристики сущности «Трафик»

<i>№ n/n</i>	<i>Название характеристики</i>	<i>Краткое описание</i>
1	Отправитель	IP адрес компьютера, с которого пересылаются данные
2	Адресат	IP адрес компьютера, принимающего данные
3	Размер	Объём пакета данных (измеряется в байтах)
4	Дата и время	Указывает на время прохождения данных через порт

Описанные объекты и связи между ними отражены на диаграмме 2.1.

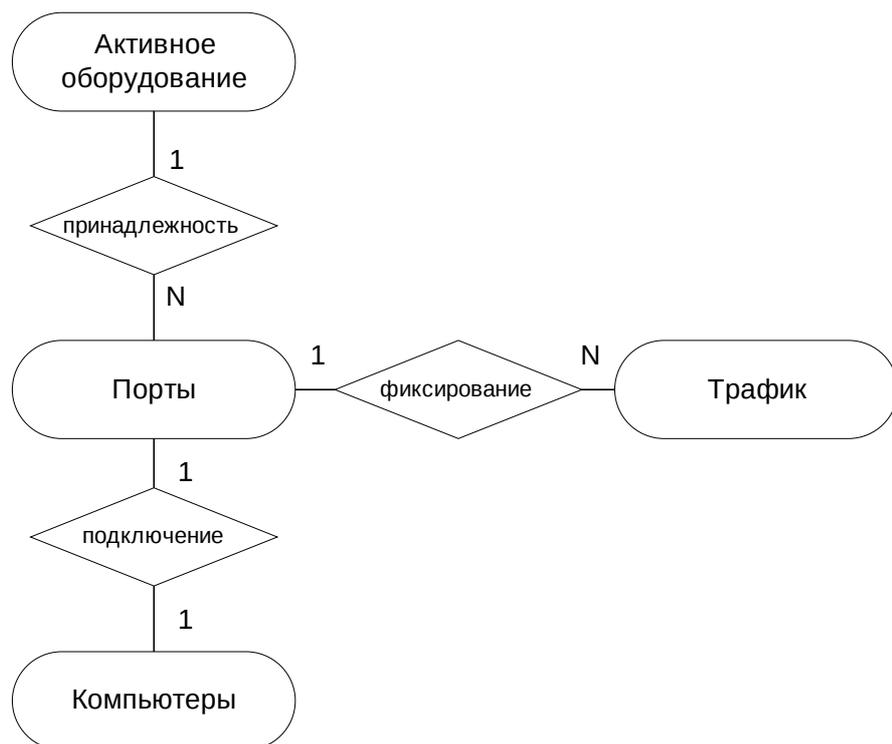


Диаграмма 2.1. Концептуальная диаграмма «Сущность - связь»

Благодаря разнообразию информации, хранящейся в базе данных, ею могут воспользоваться не только разработчики сети, желающие провести статистический анализ активности оборудования. Потенциальными пользователями этой ИС являются системные администраторы, специалисты по обслуживанию оборудования, специалисты по средствам передачи данных, а также специалисты по безопасности сети.