

ТЕХНОЛОГИЯ БАЗ ИНФОРМАЦИИ

ПЛАН:

- 1. Базы данных: понятие, элементы и основные свойства.**
- 2. Системы управления базами данных.**
- 3. Жизненный цикл баз данных.
Проектирование.**
- 4. Логические модели данных.**
- 5. Реляционные базы данных.**

1. Базы данных: понятие, элементы и основные свойства

База данных (БД) – это совокупность специальным образом организованных (структурированных) *данных*, характеризующая актуальное состояние некоторой *предметной области* и используемая для удовлетворения информационных потребностей пользователей.

Предметной областью принято называть часть реального мира, подлежащую изучению, объекты которой описаны в базе данных.

Основные структурные единицы БД

(в порядке усложнения):

- элемент данных;
- агрегат данных;
- запись данных;
- файл данных;
- база данных.

Элемент данных (или атрибут) – это наименьшая семантически значимая поименованная единица данных.

Характеристики элемента данных: ***имя, тип, длина, точность.***

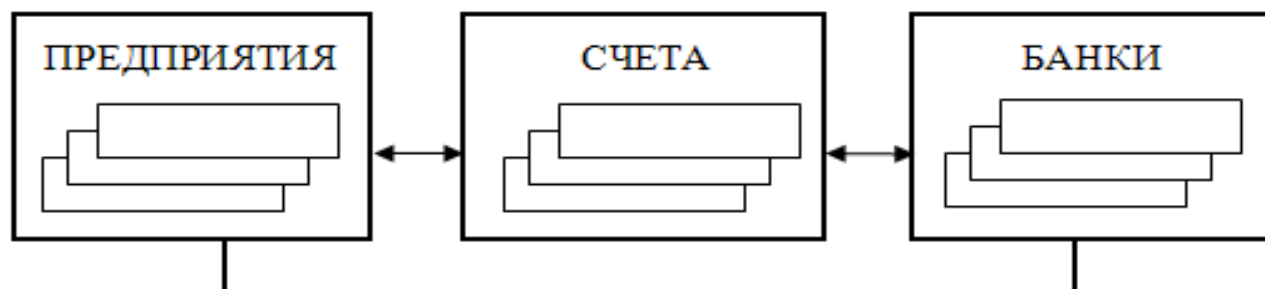
Агрегат данных – совокупность элементов данных, которую можно рассматривать как единое целое.

Запись данных – совокупность связанных элементов данных и (или) агрегатов.

Файлом данных называют набор из однотипных записей

База данных – представляет собой совокупность взаимосвязанных файлов данных.

**БАЗА
ДАННЫХ**



**ФАЙЛ
ДАННЫХ**



**ЗАПИСЬ
ДАННЫХ**

наименование	адрес				телефон
	город	индекс	улица	дом	

**АГРЕГАТ
ДАННЫХ**

адрес			
город	индекс	улица	дом

**ЭЛЕМЕНТ
ДАННЫХ**

город

Рисунок 1 – Пример взаимосвязи структурных единиц БД

Свойства БД

- *Восстанавливаемость*
- *Безопасность*
- *Целостность*
- *Эффективность*

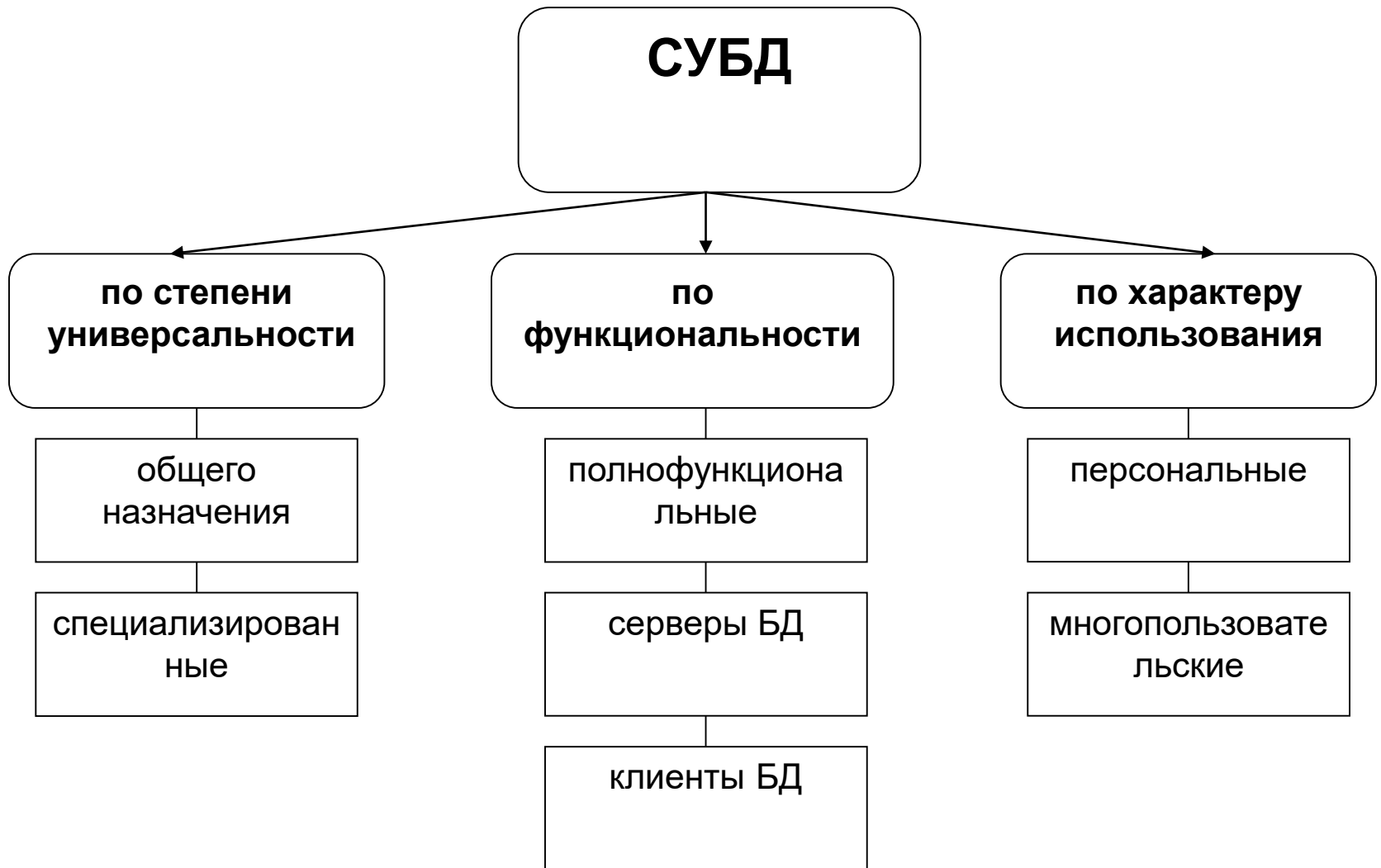
2. Системы управления базами данных

Система управление базами данных (СУБД) – совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования баз данных многими пользователями.

Основные функции СУБД:

- *непосредственное управление данными во внешней памяти;*
- *управление буферами оперативной памяти;*
- *управление транзакциями;*
- *обеспечение высокой надежности хранения данных и протоколирование;*
- *поддержка языков баз данных.*

Классификация СУБД



3. Жизненный цикл баз данных.

Проектирование

Этапы и содержание жизненного цикла БД:

- **I. проектирование**
- - 1). Анализ системной предметной области и запросов будущих пользователей в БД
 - 2). Интеграция пользовательских представлений с целью построения инфологической модели предметных областей.
 - 3). Выбор средства реализации, т.е. конкретных СУБД для разработки БД
 - 4). Датологическое проектирование
 - 5). Физическое проектирование

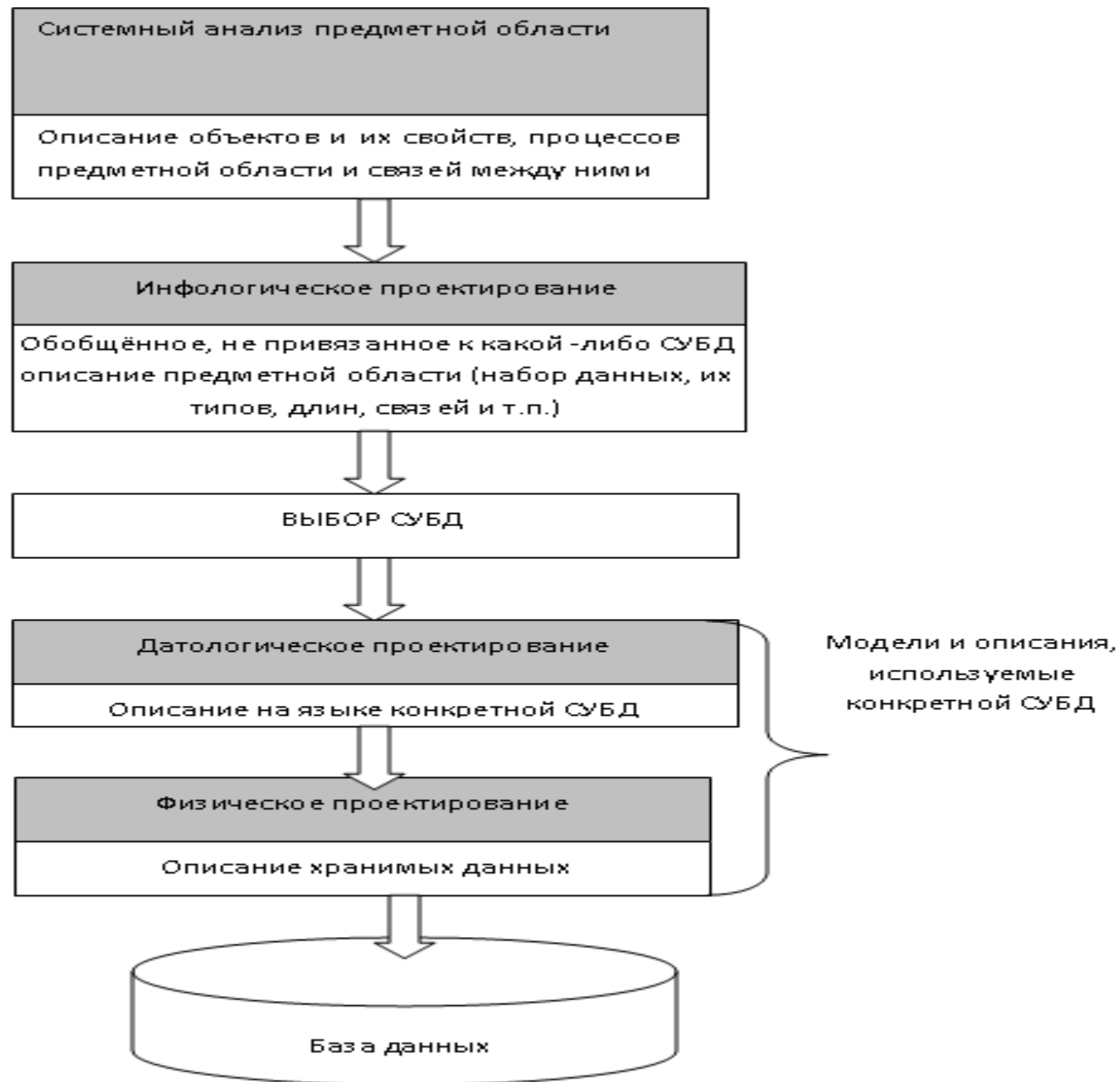
- *Инфологической моделью данных* называют описание, выполненное с использованием естественного языка, математических формул, таблиц, графиков и других средств, понятных всем людям, работающим над проектированием базы данных.
- *Даталогическое проектирование* является проектированием логической структуры БД, что означает определение всех информационных единиц и связей между ними, задание их имен и типов, а также некоторых количественных характеристик (например, длины поля).

- **II. создание**

- 1). Генерация БД
- 2). Подготовка среды хранения данных
- 3). Ввод и контроль данных
- 4). Загрузка и корректировка БД

- **III. эксплуатация**

- 1). Реорганизация БД
реструктуризация БД
реформация БД
- 2). Организация доступа к БД
поиск и обновление данных
вывод отчетов
разграничение доступа
при необходимости инициирование и завершение
работы с СУБД
- 3). Контроль состояния БД
сбор и анализ статистики использования БД
контроль целостности БД
копирование и восстановление БД (по
необходимости)



Этапы проектирования базы данных

Логическая модель

Логическая модель данных является начальным прототипом будущей
базы данных.

ER-диаграммы

Общим способом представления логической модели БД является построение **ER-диаграмм** (Entity-Relationship - **сущность-связь**). В этой модели сущность определяется как дискретный объект, для которого сохраняются элементы данных, а связь описывает отношение между двумя объектами.

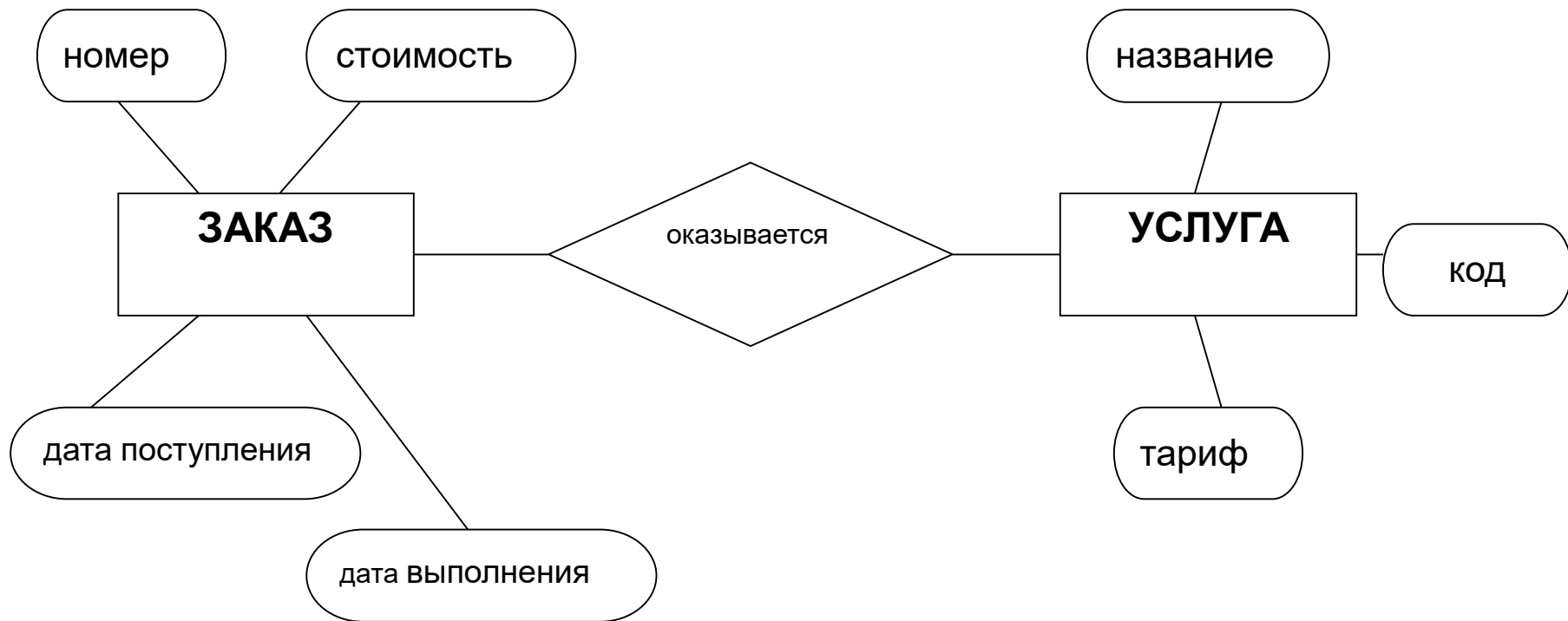


Рисунок 4 – Пример ER – диаграммы.

4. Логические модели данных

- $1 \rightarrow 1$ - «*один к одному*», когда одна запись связана только с одной записью;
- $1 \rightarrow M$ — «*один ко многим*», когда одна запись взаимосвязана со многими другими;
- $M \rightarrow M$ — «*много ко многим*», когда одна и та же запись может входить в отношения со многими другими записями в различных вариантах.

Основные логические модели данных:

- *иерархическая;*
- *сетевая;*
- *реляционная*

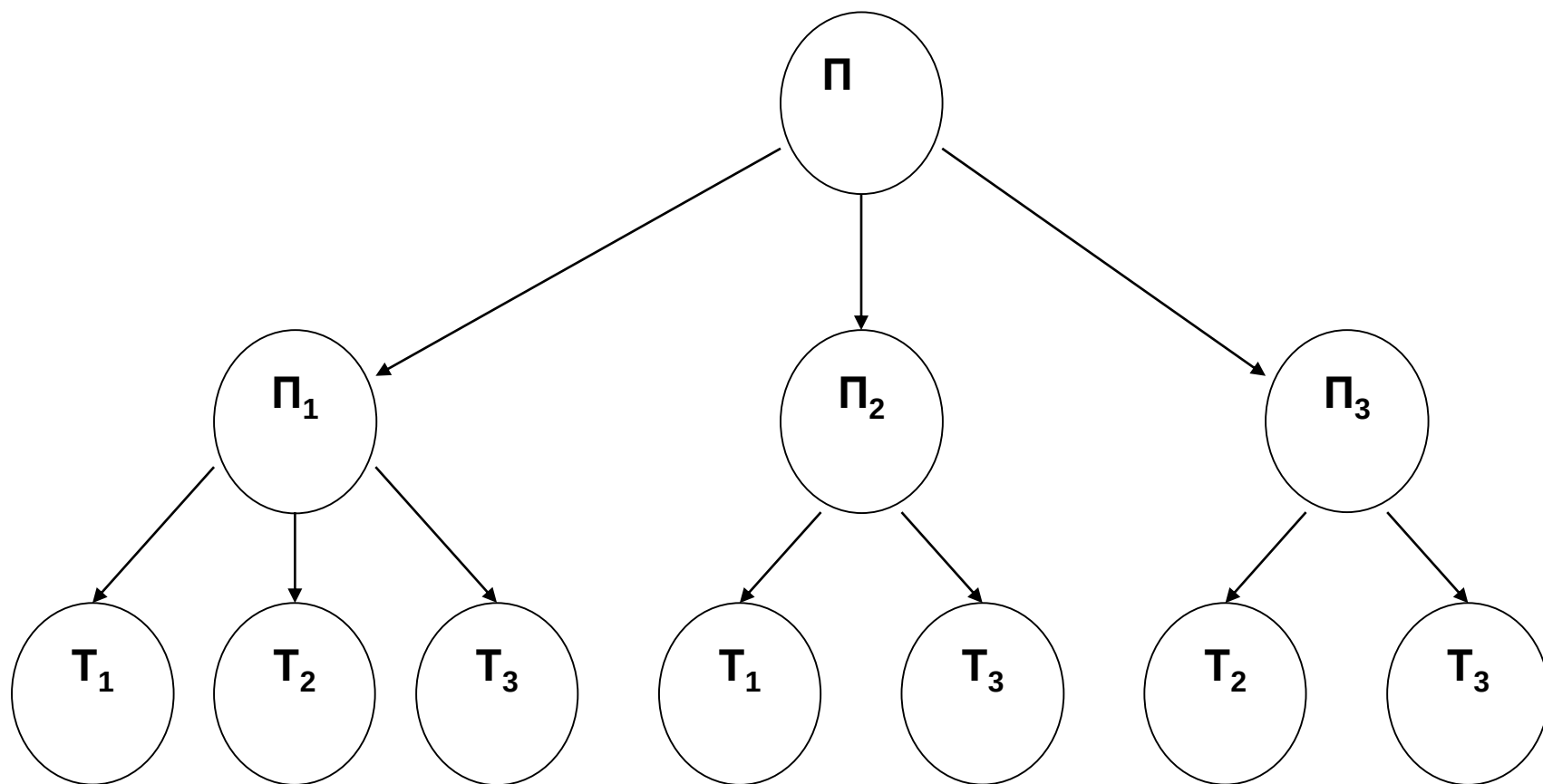


Рисунок 5 - Иерархическая модель базы данных

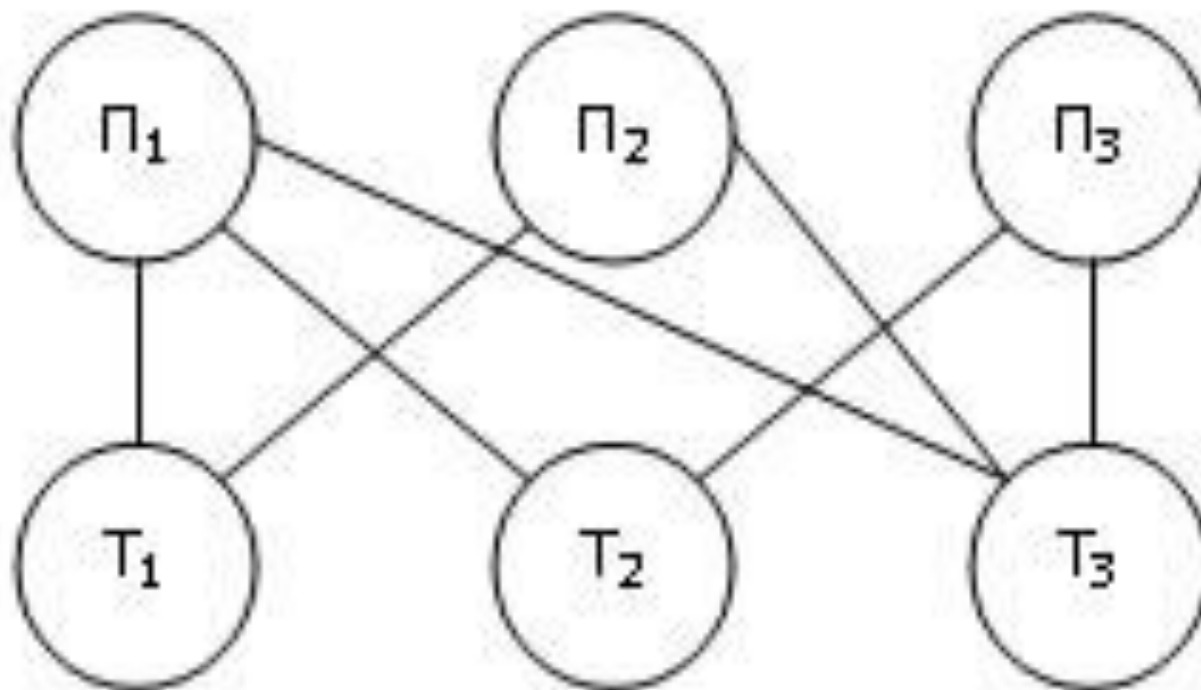


Рисунок 6 - Сетевая модель базы данных

Поставщики

Π_1
Π_2
Π_3

Товары

T_1
T_2
T_3

Поставка товаров

Π_1	T_1
Π_1	T_2
Π_1	T_3
Π_2	T_1
Π_2	T_3
Π_3	T_2
Π_3	T_3

Рисунок 7 - Реляционная модель базы данных.

5. Реляционные базы данных.

Реляционный термин	Обычный, неформальный термин
Отношение	Таблица (плоская, двумерная)
Кортеж	Запись, строка
Атрибут	Поле, столбец

- **Реляционная база данных** – это совокупность отношений, содержащих всю информацию, которая должна храниться в БД.

Сотрудники

<i>Табельный номер</i>	<i>Фамилия И.О.</i>	<i>Код отдела</i>	<i>Рабочий телефон</i>
008976	ПЕТРОВ А.В.	024	55-13-52
008980	СТЕПАНЮК О.Л.	024	55-13-43
008934	РОГОВ И.И.	025	

Отделы

<i>Код</i>	<i>Наименование отдела</i>
010	ДИРЕКЦИЯ
023	БУХГАЛТЕРИЯ
024	ОТДЕЛ КАДРОВ
025	КАНЦЕЛЯРИЯ

Между двумя таблицами можно установить отношение
«СОТРУДНИК – работает в – ОТДЕЛЕ».

008980	СТЕПАНЮК О.Л.	024	55-13-43
--------	---------------	-----	----------

+

024	ОТДЕЛ КАДРОВ
-----	--------------

=

008980	СТЕПАНЮК О.Л.	024	55-13-43	ОТДЕЛ КАДРОВ
--------	---------------	-----	----------	--------------

Нормализация отношений

- Результаты инфологического моделирования могут быть сведены в единую таблицу. Но использовать ее невозможно, т.к. скорее всего эти таблицы будут содержать избыточную, повторяющуюся информацию.
- Поэтому полученные данные подвергают преобразованиям, которые называются нормализацией .

Нормализация отношений реляционных БД

Первая нормальная форма (1НФ).

Отношение называется **приведенным к первой нормальной форме**, если все его атрибуты простые.

Отношение *Сотрудник* = (Таб_номер; ФИО_сотрудника; Дата_рождения; Дети)

Сотрудник = (Таб_номер; Фамилия; Имя; Отчество; Дата_рождения;)

Дети = (Таб_номер; Имя ребенка; Дата_рождения_ребенка)

Отношение Сотрудник

Таб_номер

Фамилия

Имя

Отчество

Дата_рождения



Отношение Дети

Таб_номер

Имя_ребенка

Дата_рождения_ребенка

Вторая нормальная форма (2НФ).

Отношение находится во второй нормальной форме, если оно находится в первой нормальной форме и значения в каждом неключевом атрибуте однозначно определяются значением первичного ключа.

- Отношение *Заказанное_изделие* = (**Номер_заказа; Код_изделия**; Название_изделия; Цена; Количество)
- *Заказанное_изделие* = (**Номер_заказа; Код_изделия**; Количество).
- *Изделие* = (**Код_изделия**; Название_изделия; Цена).

Одноименное отношение

Отношение *Заказанноеизделие*

Номер_заказа

Код_изделия

Количество

Отношение *Изделие*

1 — Код_изделия

Название_изделия

Цена



Третья нормальная форма (3НФ)

- **Отношение находится в третьей нормальной форме, если оно находится во второй нормальной форме и все неключевые атрибуты не зависят друг от друга.**

Четвертая нормальная форма (4НФ)

Отношение находится в четвертой нормальной форме , если оно находится в третьей нормальной форме и если в нем не содержатся независимые группы атрибутов, между которыми существует отношение «многие-ко-многим».