

Преимущества технологии баз данных

1. Исключается избыточность данных, т.к. нет необходимости хранить в нескольких экземплярах один и тот же элемент данных.

В распределенных БД может присутствовать тиражирование (избыточность) данных, но только на физическом уровне.

На логическом уровне (т.е. с точки зрения пользователя) это не заметно, т.к. СУБД постоянно поддерживает идентичность

3. Наличие в составе СУБД развитых средств резервного копирования и восстановления данных в случае нарушения целостности, а также других эффективных средств повышения
4. Применение стандартных форматов при обмене данными.
5. Возможность самостоятельного ввода любых запросов на языке SQL повышает доступность данных для конечных пользователей.
6. Более простое сопровождение приложений, работающих с БД, за счет

Недостатки технологии баз данных

1. По мере расширения функциональных возможностей СУБД они становятся все более сложными инструментами.
Как следствие, возрастают требования к квалификации специалистов для этой области информационных технологий.
2. Для мощных СУБД резко возрастает стоимость лицензионных программных средств системы.
3. Централизация ресурсов делает систему

Трехуровневая архитектура базы данных

Элементы данных, присутствующих в БД, могут рассматриваться с самых разных позиций.

Для примера укажем два крайних варианта:

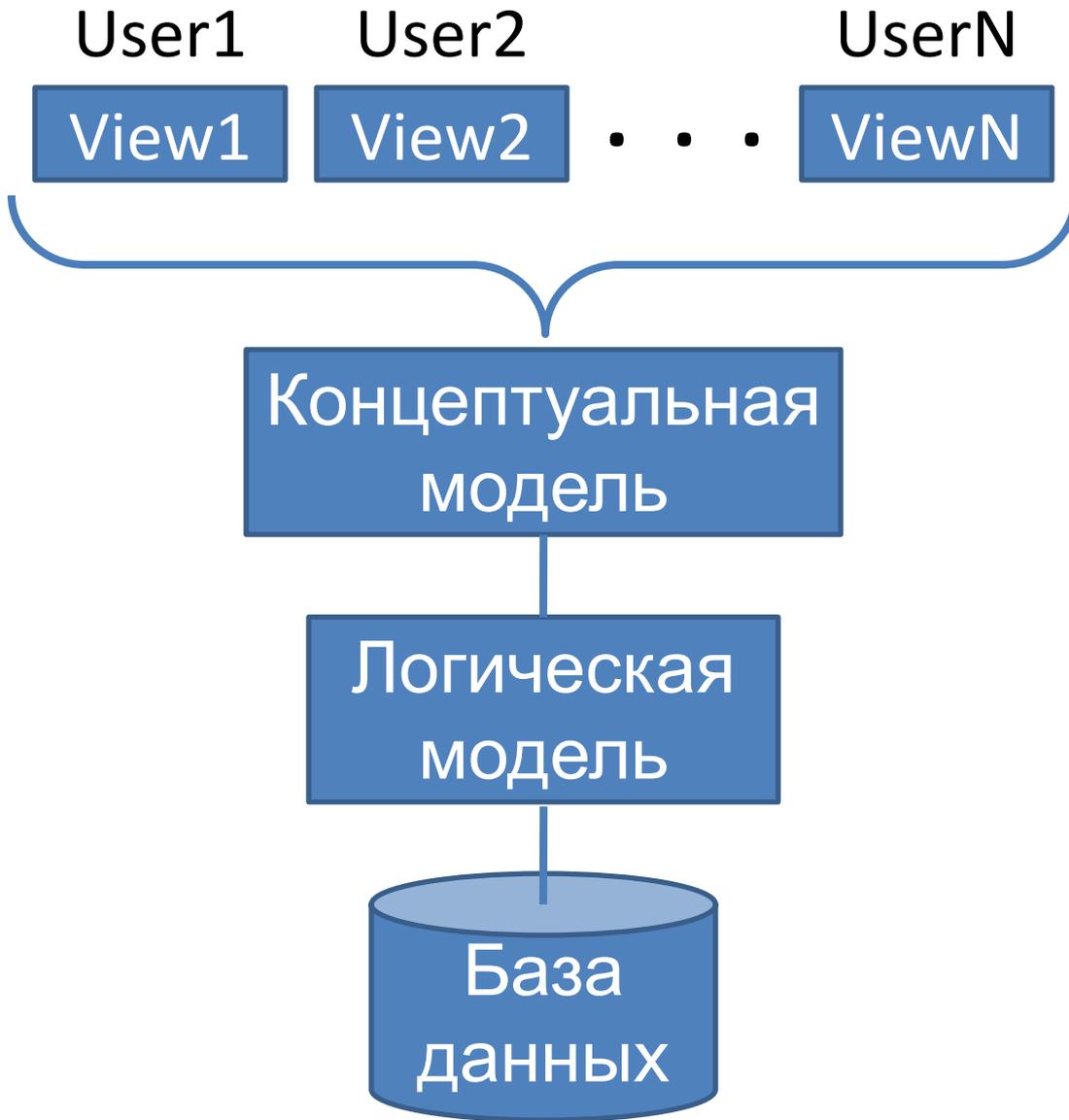
- самый верхний уровень — это точка зрения **обычного пользователя**, которого интересуют только **значения конкретных данных**;
- самый низкий уровень — это представление данных на **физическом носителе**, что важно для программных средств **операционной системы** или **системных программистов**.

Возникают и **промежуточные уровни** рассмотрения БД, когда на разных этапах жизненного цикла БД привлекаются другие специалисты, каждый из которых имеет свой взгляд на БД.

Трехуровневая архитектура для описания БД была предложена в 1975 г.

Национальным институтом стандартизации США (ANSI).

До настоящего времени эта модель имеет важное значение для понимания типичной организации баз данных.



Внешний
уровень

Концептуальный
уровень

Логический
уровень

Физическая
организация
данных

Уровень, на котором данные воспринимаются разными пользователями, называется

внешним

Этот уровень состоит из отдельных

представлений (*view*), причем каждое из них

содержит только те данные, которые интересны

конкретной категории пользователей.

Представление может содержать **производные**

(вычисляемые) данные, которые не хранятся

в БД, а создаются по мере надобности.

Помимо этого, разные представления могут

отображать одни и те же данные **в разных**

форматах.

Концептуальный уровень формирует полное представление для **общих требований** к данным со стороны **всех пользователей**.
На концептуальном уровне основными

элементами БД являются:

- 1. Сущности** – отдельные классы однотипных объектов, которые связаны с деятельностью в рассматриваемой предметной области. Понятию «сущность» (или «информационный объект») можно сопоставлять материальные объекты (человек, вещь) или различные события (факты), которые нужно регистрировать в БД.

2. Атрибуты (реквизиты) – характеристики свойств, которыми обладают сущности. Актуальные значения атрибутов как раз и подлежат хранению в БД.

3. Связи, которые отображают

взаимоотношения между сущностями. К концептуальному уровню также относят:

- предварительные типы данных для атрибутов (целочисленный, с плавающей точкой, символьный и др.);
- ограничения, которые накладываются на значения атрибутов.

Концептуальный уровень оставляет без внимания организацию хранения

При своей работе (выполнение запросов) СУБД обращается к данным на **физическом**

уровне именно здесь при хранении и поиске данных используются различные виды файлов:

- неупорядоченные файлы с линейным поиском;
- упорядоченные (отсортированные) файлы;
- хешированные файлы (для вычисления адреса страницы, на которой находится требуемая запись, используется специальная хеш-функция);
- индексные файлы (для ускорения поиска данных в неупорядоченных файлах).

Реализация концептуальной модели в виде конкретной БД отображается с помощью **промежуточного (внутреннего)** уровня, который содержит ***логическую структуру*** всей БД. Внедрение трехуровневой архитектуры облегчает независимое внесение изменений на разных уровнях организации БД.

Примеры:

1. Изменение некоторых представлений на внешнем уровне будет незаметным для других представлений.
2. Изменение логической структуры БД или переход к другим способам хранения данных и

Информационное моделирование

На этапе создания БД важно описать требования к данным в масштабе всех категорий пользователей.

Такое описание должно быть доступным для понимания всем заинтересованным лицам, чтобы проектировщики БД смогли уяснить роль тех или иных данных в деятельности пользователей БД.

С этой целью строится так называемая **информационная модель (ИМ)**

В соответствии с трехуровневой архитектурой описания БД, требуется

несколько ИМ:

- **внешняя модель данных**, которая отображает представления каждой категории пользователей;
- **концептуальную модель** – для обобщенного, но абстрактного представления о данных независимо от конкретной СУБД;
- **внутренняя (логическая) модель данных** – как отображение концептуальной

Среди этих моделей наиболее важная роль принадлежит концептуальной модели

~~Именно~~ Эта модель должна быть полным и точным описанием всех требований к данным со стороны всех категорий пользователей.

Построение такой модели, которая носит наиболее общий характер и ***не зависит*** от особенностей реализации БД, является основной задачей этапа проектирования БД.

Внутренняя (логическая) модель — это лишь ~~одна из~~

Требования к моделям данных

Модель данных — это интегрированный набор понятий, которые позволяют:

- описать структурную организацию данных, логические связи между элементами данных и ограничения, накладываемые на данные;
- указать операции, применяемые для манипулирования данными.

Наличие модели данных способствует более содержательному общению пользователей и разработчиков, если обе стороны знакомы с

С этой точки зрения наилучшая модель данных должна удовлетворять следующим **требованиям**, т.е. удобство изучения модели как профессионалами в области технологий БД, так и обычными пользователями;

- **выразительность**, т.е. способность отчетливо представлять все существенные аспекты, связанные с моделированием данных;
- **исключение избыточности**, т.е. лаконичность и отсутствие излишней информации;

Нередко эти критерии в совокупности становятся **несовместимыми**.

Например, легко утратить простоту модели данных в погоне за наибольшей выразительностью.

Поэтому постоянно приходится искать какие-то компромиссы, что привело к **многочисленным** видам моделей, которые описаны в литературе.

Известные модели данных можно разделить на два класса:

- высокоуровневые **объектные модели данных** – предназначены главным образом для концептуального описания предметной