

ТЕМА 1. ВВЕДЕНИЕ. ОСНОВНЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Информационная система (ИС) — это система, предназначенная для ведения информационной модели какой-либо области человеческой деятельности (предметной области). Эта система должна обеспечивать средства для протекания **информационных процессов**:

- хранение,
- передача,
- поиск и преобразование информации.

Особенности информационных систем:

- для обеспечения их работы нужны сравнительно низкие вычислительные мощности;
- данные, которые они используют, имеют сложную структуру;
- необходимы средства сохранения данных между последовательными запусками системы.

Предметная область – часть реального мира, подлежащая изучению с целью организации управления и автоматизации. Представляется множеством фрагментов, каждый из которых характеризуется множеством объектов и процессов, использующих объекты, а также множеством пользователей, характеризующихся различными взглядами на предметную область.

Данные (калька от лат. *data*) — это представление фактов и идей в формализованном виде, пригодном для передачи и обработки в некотором информационном процессе. Данные – это выделенная (из системы, благодаря обособленности существования носителя) информация.

База данных (по Дж. Мартину) – совокупность взаимосвязанных данных, совместно используемых несколькими приложениями и хранящимися с (минимальной) регулируемой избыточностью. Данные запоминаются таким образом, чтобы они, по мере возможности, не зависели от программ. Для обработки данных применяется общий управляющий метод доступа. Если базы данных не пересекаются по структуре, то говорят о системе баз данных.

База данных (по материалам **CODASYL**, *COnference on DAta SYstems Language*) – состоит из всех экземпляров записей, экземпляров наборов записей и областей, которые контролируются конкретной схемой. (Под схемой можно понимать карту всей логической структуры базы данных)

Свойства БД:

- структурная целостность;
- непротиворечивость (семантическая целостность);
- отсутствие избыточности.

Системой управления базами данных (СУБД) называется совокупность программных средств, необходимых для использования базы данных и предоставляющих разработчикам и пользователям множество различных представлений данных.

Основные функции СУБД:

- **управление данными во внешней памяти** (обеспечение необходимых структур внешней памяти как для хранения данных, организация ускорения доступа к данным и др.);

В некоторых реализациях СУБД активно используются возможности существующих файловых систем, в других работа производится вплоть до уровня устройств внешней памяти. В развитых СУБД пользователи не обязаны знать, использует ли СУБД файловую систему, и если использует, то как организованы файлы. В частности, СУБД поддерживает собственную систему именования объектов БД.

- **управление данными в оперативной памяти;**

СУБД обычно работают с БД значительного размера; по крайней мере этот размер обычно существенно больше доступного объема оперативной памяти. Понятно, что если при обращении к любому элементу данных будет производиться обмен с внешней памятью, то вся система будет работать со скоростью устройства внешней памяти. Практически единственным способом реального увеличения этой скорости является буферизация данных в оперативной памяти. При этом, даже если операционная система производит общесистемную буферизацию (как в случае ОС UNIX), этого недостаточно для целей СУБД, которая располагает гораздо большей информацией о полезности буферизации той или иной части БД. Поэтому в развитых СУБД поддерживается собственный набор буферов оперативной памяти с собственной дисциплиной замены буферов.

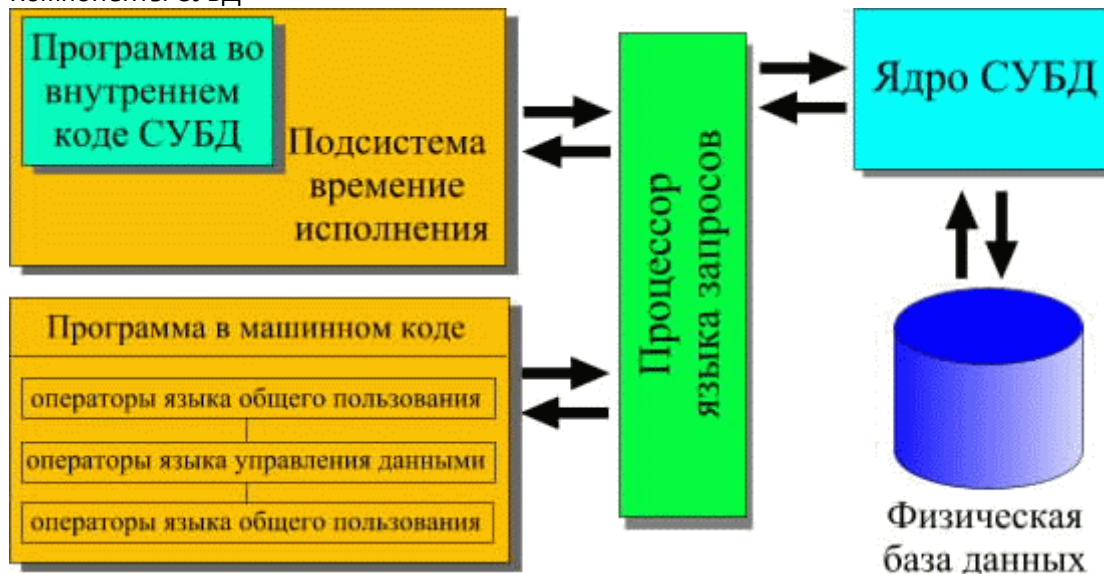
- **управление транзакциями** (поддержание целостности, сериализация);
- **журнализация изменений и восстановление базы данных после сбоев;**

Журнал – это особая часть БД, недоступная пользователям СУБД и поддерживаемая с особой тщательностью (иногда поддерживаются две копии журнала, располагаемые на разных физических дисках), в которую поступают записи обо всех изменениях основной части БД. В разных СУБД изменения БД журналируются на разных уровнях: иногда запись в журнале соответствует некоторой

логической операции изменения БД (например, операции удаления строки из таблицы реляционной БД), иногда - минимальной внутренней операции модификации страницы внешней памяти; в некоторых системах одновременно используются оба подхода.

- *поддержка языков БД* (SDL – scheme definition language, DML – data manipulation language)

Компоненты СУБД



Компоненты СУБД (необходимо модифицировать: 1) добывать утилиты от ядра; 2) приложение)

- *ядро СУБД* (Data Base Engine) – отвечает за управление данными во внешней памяти, управление буферами оперативной памяти, управление транзакциями и журнализацию; обладает собственным интерфейсом, не доступен пользователям напрямую, является основной резидентной частью СУБД;

- *процессор языка запросов* – обеспечивает оптимизацию запросов на извлечение и изменение данных и создание машинно-независимого исполняемого внутреннего кода;

Основной функцией компилятора языка БД является компиляция операторов языка БД в некоторую выполняемую программу. Основной проблемой реляционных СУБД является то, что языки этих систем (а это, как правило, SQL) являются непроцедурными, т.е. в операторе такого языка специфицируется некоторое действие над БД, но эта спецификация не является процедурой, а лишь описывает в некоторой форме условия совершения желаемого действия (вспомните примеры из первой лекции). Поэтому компилятор должен решить, каким образом выполнять оператор языка прежде, чем произвести программу. Применяются достаточно сложные методы оптимизации операторов. Результатом компиляции является выполняемая программа, представляемая в некоторых системах в машинных кодах, но более часто в выполняемом внутреннем машинно-независимом коде.

- *подсистема времени исполнения* – интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД;

- *утилиты* – реализация процедур, которые слишком накладно выполнять с использованием языка БД (загрузка и выгрузка БД, сбор статистики, глобальная проверка целостности БД и т.д.); программируются с использованием интерфейса ядра СУБД, а иногда даже с проникновением внутрь ядра.

Примеры: dBase, Paradox, FoxPro, InterBase, MS SQL Server, Oracle, MySQL и др.

Модель данных – это инструмент представления концептуальной модели предметной области и динамики ее изменения в виде базы данных.

Основные компоненты:

- структура данных – описывает точку зрения пользователя на представление данных, а также набор допустимых операций, выполняемых на структуре данных;
- SDL (scheme definition language) – язык определения данных; DML – data manipulation language) – языка манипулирования данными;
- ограничения целостности – механизм поддержания соответствия данных предметной области на основе формально описанных правил.

Классификация моделей данных по структуре данных и характеру связей между элементами:

<i>Модель данных</i>	<i>Характер связи между объектами</i>	<i>Формальное представление</i>	<i>Примеры</i>
Иерархическая	Жесткие связи	Древовидная структура	Регистр Windows, Active Directory
Сетевая	Полужесткие связи	Произвольный граф	CronosPlus (cronos.ru)
Реляционная	Изменчивые связи	Плоский файл	
Многомерная	Изменчивые связи	n-мерный куб	MS OLAP Services, Oracle Express Server
Объектно-ориентированные	Изменчивые связи	Модели объектов	IBM Lotus Domino, Cache
NoSQL	Полужесткие связи	Json объекты	MongoDB

Архитектуры данных – способы организации взаимодействия информационной системы (пользователя) с БД, обеспечивающие наиболее эффективное ее использование.

Классификация:

- централизованная архитектура (модель с автономными персональными ЭВМ),
- архитектура «файл-сервер» (модель вычислений с сетью и файловым сервером),
- архитектура «клиент-сервер» (распределенная модель вычислений),
- трехзвенная (многозвенная) архитектура (распределенная модель вычислений).