Базы данных

Модели данных

Понятие модели данных

Модель данных – это совокупность правил порождения структур данных в базе данных, операций над ними, а также ограничений целостности, определяющих допустимые связи и значения данных, последовательность их изменения [ГОСТ 20886-85].

Модель данных состоит из трёх частей:

1. Набор типов структур данных.

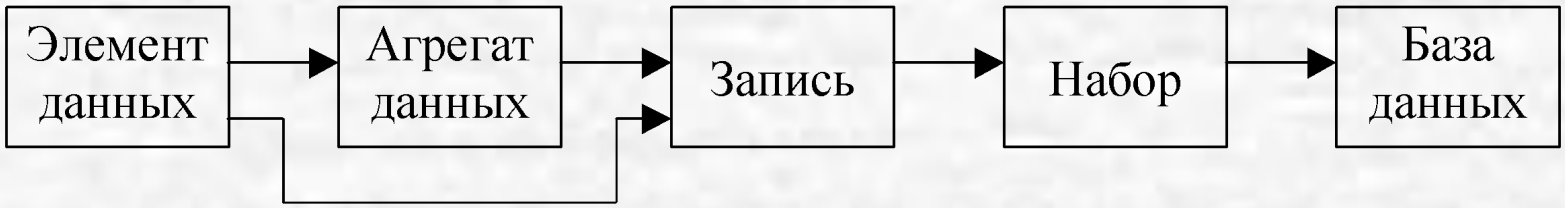
Здесь можно провести аналогию с языками программирования, в которых тоже есть предопределённые типы структур данных, такие как скалярные данные, векторы, массивы, структуры (например, тип *struct* в языке Си) и т.д.

2. Набор операторов или правил вывода, которые могут быть применены к любым правильным примерам типов данных, перечисленных в (1), чтобы находить, выводить или преобразовывать информацию, содержащуюся в любых частях этих структур в любых комбинациях.

3. Набор общих правил целостности, которые прямо или косвенно определяют множество непротиворечивых состояний базы данных и/или множество изменений её состояния.

Типы структур данных. Версия CODASYL

Структуризация данных базируется на использовании концепций "агрегации" и "обобщения". Один из первых вариантов структуризации данных был предложен Ассоциацией по языкам обработки данных (Conference on Data Systems Languages, CODASYL):



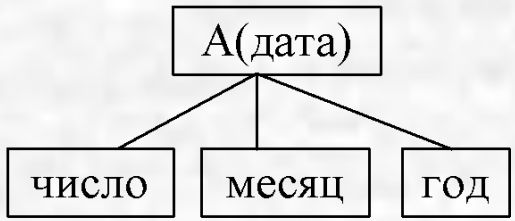
Элемент данных – наименьшая поименованная единица данных, к которой СУБД может обращаться непосредственно и с помощью которой выполняется построение всех остальных структур.

Для каждого элемента данных должны быть определены название и тип.

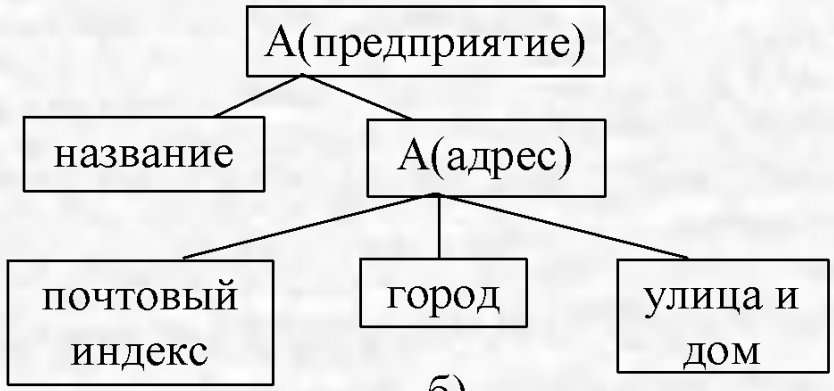
Версия CODASYL. Агрегаты

Агрегат данных – поименованная совокупность элементов данных внутри записи, которую можно рассматривать как единое целое. Агрегат может быть *простым* (включающим только элементы данных, рис. а) и *составным* (включающим наряду с элементами данных и другие агрегаты, рис. б).

A(<название>) – агрегат данных



а)



б)

Для каждого агрегата должны быть определены название и структура.

Версия CODASYL. Запись

Запись – поименованная совокупность элементов данных или элементов данных и агрегатов. Запись – это агрегат, не входящий в состав никакого другого агрегата; она может иметь сложную иерархическую структуру, поскольку допускается многократное применение агрегации. Различают тип записи (её структуру) и экземпляр записи, т.е. запись с конкретными значениями элементов данных. Одна запись описывает свойства одной сущности ПО (экземпляра).
Иногда термин "запись" заменяют термином "группа".

Пример записи типа "Сотрудник":

Ключевые поля.

Версия CODASYL. Набор. База данных

Набор (или *групповое отношение*) – поименованная совокупность записей, образующих двухуровневую иерархическую структуру. Каждый тип набора представляет собой связь между двумя или несколькими типами записей. Для каждого типа набора один тип записи объявляется владельцем набора, остальные типы записи объявляются членами набора. Для группового отношения также различают тип и экземпляр.

Фрагмент диаграммы Бахмана для БД "Город":

База данных – поименованная совокупность экземпляров групп и групповых отношений.

Операции над данными

Модель данных определяет множество действий, которые допустимо производить над некоторой реализацией БД для её перевода из одного состояния в другое. Это множество соотносят с **языком манипулирования данными** (Data Manipulation Language, DML).

Любая операция над данными включает в себя **селекцию данных** (select).

Условие селекции – это некоторый критерий отбора данных, в котором могут быть использованы логическая позиция элемента данных, его значение и связи между данными.

По типу производимых действий различают следующие операции:

- идентификация данных и нахождение их позиции в БД;
- выборка (чтение) данных из БД;
- включение (запись) данных в БД;
- удаление данных из БД;
- модификация (изменение) данных БД.

Обработка данных в БД осуществляется с помощью процедур базы данных – транзакций. **Транзакцией** называют упорядоченное множество операций, переводящих БД из одного согласованного состояния в другое.

Ограничения целостности

Ограничения целостности – это правила, которым должны удовлетворять значения элементов данных. Ограничения целостности делятся на:

- **явные** (включаются в структуру базы данных с помощью средств языка контроля данных (DCL, Data Control Language))
- **неявные** (определяются самой структурой данных).

Также различают **статические** и **динамические** ограничения целостности. Статические ограничения присущи всем состояниям ПО, а динамические определяют возможность перехода ПО из одного состояния в другое.

За выполнением ограничений целостности следит СУБД в процессе своего функционирования. Она проверяет ограничения целостности каждый раз, когда они могут быть нарушены (например, при добавлении данных, при удалении данных и т.п.), и гарантирует их соблюдение.

Таким образом, ограничения целостности обеспечивают логическую непротиворечивость данных при переводе БД из одного состояния в другое.

Сетевая модель данных (СМД)

Сетевая модель позволяет организовывать БД, структура которых представляется графом общего вида.
Организация данных в сетевой модели соответствует структуризации данных по версии CODASYL.
Связи между записями в СМД выполняются в виде указателей, т.е. каждая запись хранит ссылку на другую однотипную запись (или признак конца списка) и ссылки на списки подчинённых записей, связанных с ней групповыми отношениями.

СМД. Основные характеристики

1. Способ упорядочения подчинённых записей.

Поддерживаются три способа упорядочения:

- Очередь – добавление в конец списка (FIFO – first input, first output).
- Стек – добавление в начало списка (LIFO – last input, first output).
- Сортировка по значению ключа. При этом задаётся ключевое поле (группа полей), и вновь поступившая запись добавляется в упорядоченный список в соответствии со значением этого поля (значением ключа).

2. Режим включения подчинённых записей.

Режим включения бывает **автоматический** и **ручной**.

При автоматическом режиме подчинённая запись связана с записью-владельцем обязательной связью, поэтому она включается в групповое отношение и прикрепляется к записи-владельцу в момент внесения в БД.

При ручном режиме включения подчинённая запись может находиться в БД и не быть прикрепленной к записи-владельцу. Она вручную включается в групповое отношение тогда, когда это отношение (связь) возникает.

СМД. Основные характеристики

3. Режим исключения подчинённых записей.

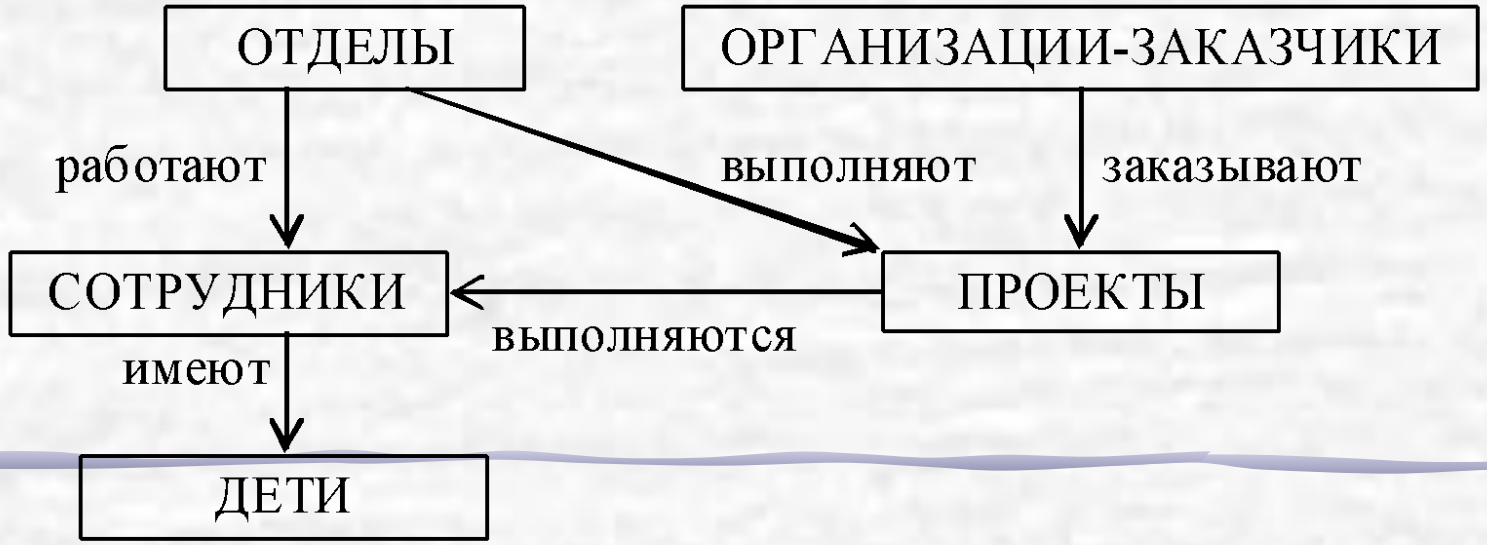
Режим исключения определяется **классом членства**.

Различают три класса членства:

– **фиксированный:** записи с фиксированным членством удаляются вместе с записью-владельцем;

– **обязательный:** записи с обязательным членством должны быть удалены до удаления записи-владельца;

– **необязательный: з**аписи с необязательным членством при удалении записи-владельца останутся в БД.

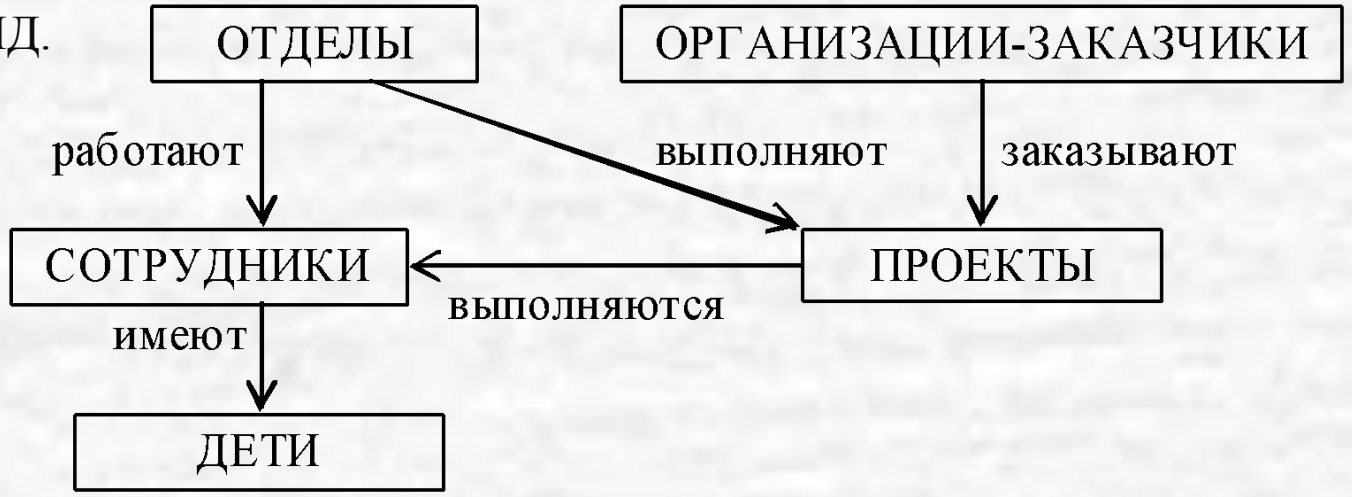


СМД. Основные характеристики

В СМД применяются следующие операции над данными:

- *запомнить*: внесение информации в БД;
- *включить в групповое отношение:* установление связей между данными;
- *переключить*: переход члена набора к другому владельцу;
- *обновить*: модификация данных;
- *извлечь*: чтение данных;
- *удалить*: физическое или логическое удаление данных;
- *исключить из группового отношения*: разрыв связей между данными.

Навигация в СМД.



СМД. Реализации. Достоинства и недостатки

Наиболее распространенной и стандартизованной из реализаций СМД является модель CODASYL. В соответствии с ней описание схемы БД осуществляется на языке COBOL, а манипулирование данными – с помощью включающего языка программирования высокого уровня. Примером сетевой СУБД является система Integrated Database Management System (IDMS).

СМД является **наиболее полной** с точки зрения реализации различных типов связей и ограничений целостности, но она является достаточно **сложной** для проектирования и поддержки. В этой модели **не обеспечивается физическая независимость данных**, т.к. наборы организованы с помощью физических ссылок. Также в СМД **не обеспечивается независимость данных от программ**. Из-за этих недостатков эта модель не получила широкого распространения.

Иерархическая модель данных (ИМД)

Иерархическая модель позволяет строить БД с иерархической древовидной структурой. Структура ИМД описывается в терминах, аналогичных терминам сетевой модели данных (версия CODASYL). Группу в ИМД принято называть *сегментом*. В основе ИМД лежит понятие дерева.

Дерево – это связный неориентированный граф, который не содержит циклов. При работе с деревом выделяют какую-то конкретную вершину, определяют её как корень дерева и рассматривают особо – в эту вершину не заходит ни одно ребро. В этом случае дерево становится ориентированным, ориентация определяется от корня. Дерево как ориентированный граф определяется так:

- имеется единственная особая вершина, называемая корнем, в которую не заходит ни одно ребро;
- во все остальные вершины заходит только одно ребро, а исходит произвольное количество ребер;
- граф не содержит циклов.

Конечные вершины, то есть вершины, из которых не выходит ни одной дуги, называются *листьями* дерева. Количество вершин на пути от корня к листьям в разных ветвях дерева может быть различным.

ИМД. Основные характеристики

Графическая диаграмма концептуальной схемы базы данных называется *деревом определения*. Пример иерархической базы данных:

ИМД не поддерживает: **необязательный класс членства и ручной режим включения записей.**

ИМД. Основные характеристики

Полный сцепленный ключ. Операции над данными. Навигация в ИМД.

Основной недостаток: дублирование данных.

Модели данных

- Иерархическая модель данных (ИМД).
- Сетевая модель данных (СМД).

} I поколение

- Реляционная модель данных (РМД). - II поколение
- Объектно-реляционная модель данных (ОРМД). - III поколение

  Стандарт SQL-3 (SQL-2003). Oracle (с версии 8.0), DB2, Informix, PostgreSQL, SQL Server 2008 и др.)
- Объектно-ориентированная модель данных (ООМД). O2, GemStone, Iris и др.

  Стандарт ODMG 3.0 (Object Database Management Group).
- Многомерные базы данных.
- Потоковые базы данных.
- ...