

Автор: Леонов Сергей Михайлович

Содержание

- ▶ Сетевая модель данных
- ▶ *Сетевые структуры данных*
- ▶ *Манипулирование данными*
- ▶ *Ограничения целостности*

Сетевая модель данных

- Типичным представителем систем, основанных на сетевой модели данных, является СУБД IDMS (Integrated Database Management System), разработанная компанией Cullinet Software, Inc. и изначально ориентированная на использования на мейнфреймах компании IBM.
- Архитектура системы основана на предложениях Data Base Task Group (DBTG) организации CODASYL (COncference on DAta SYstems Languages), которая отвечала за определение языка программирования COBOL.
- Отчет DBTG был опубликован в 1971 г., и вскоре после этого появилось несколько систем, поддерживающих архитектуру CODASYL, среди которых присутствовала и СУБД IDMS.
- В настоящее время IDMS принадлежит компании Computer Associates.

Сетевые структуры данных

- ▶ Сетевой подход к организации данных является расширением иерархического подхода.
- ▶ В иерархических структурах запись-потомок должна иметь в точности одного предка; в сетевой структуре данных у потомка может иметься любое число предков.
- ▶ Сетевая БД состоит из набора записей и набора связей между этими записями, а если говорить более точно, из набора экземпляров каждого типа из заданного в схеме БД набора типов записи и набора экземпляров каждого типа из заданного набора типов связи.
- ▶ Тип связи определяется для двух типов записи: предка и потомка. Экземпляр типа связи состоит из одного экземпляра типа записи предка и упорядоченного набора экземпляров типа записи потомка. Для данного типа связи L с типом записи предка P и типом записи потомка C должны выполняться следующие два условия:

- ▶ каждый экземпляр типа записи P является предком только в одном экземпляре типа связи L ;
- ▶ каждый экземпляр типа записи C является потомком не более чем в одном экземпляре типа связи L .

- ▶ На формирование типов связи не накладываются особые ограничения; возможны, например, следующие ситуации:
- ▶ тип записи потомка в одном типе связи L1 может быть типом записи предка в другом типе связи L2 (как в иерархии);
- данный тип записи P может быть типом записи предка в любом числе типов связи;
- данный тип записи P может быть типом записи потомка в любом числе типов связи;
- может существовать любое число типов связи с одним и тем же типом записи предка и одним и тем же типом записи потомка; и если L1 и L2 - два типа связи с одним и тем же типом записи предка P и одним и тем же типом записи потомка C, то правила, по которым образуется родство, в разных связях могут различаться;
- типы записи X и Y могут быть предком и потомком в одной связи и потомком и предком - в другой;
- предок и потомок могут быть одного типа записи.

На рис. 2.3 показан простой пример схемы сетевой БД. На этом рисунке показаны три типа записи: Отдел, Служащие и Руководитель и три типа связи: Состоит из служащих, Имеет руководителя и Является служащим. В типе связи Состоит из служащих типом записи-предком является Отдел, а типом записи-потомком - Служащие (экземпляр этого типа связи связывает экземпляр типа записи Отдел со многими экземплярами типа записи Служащие, соответствующими всем служащим данного отдела).

В типе связи Имеет руководителя типом записи-предком является Отдел, а типом записи-потомком - Руководитель (экземпляр этого типа связи связывает экземпляр типа записи Отдел с одним экземпляром типа записи Руководитель, соответствующим руководителю данного отдела). Наконец, в типе связи Является служащим типом записи-предком является Руководитель, а типом записи-потомком - Служащие (экземпляр этого типа связи связывает экземпляр типа записи Руководитель с одним экземпляром типа записи Служащие, соответствующим тому служащему, которым является данный руководитель).

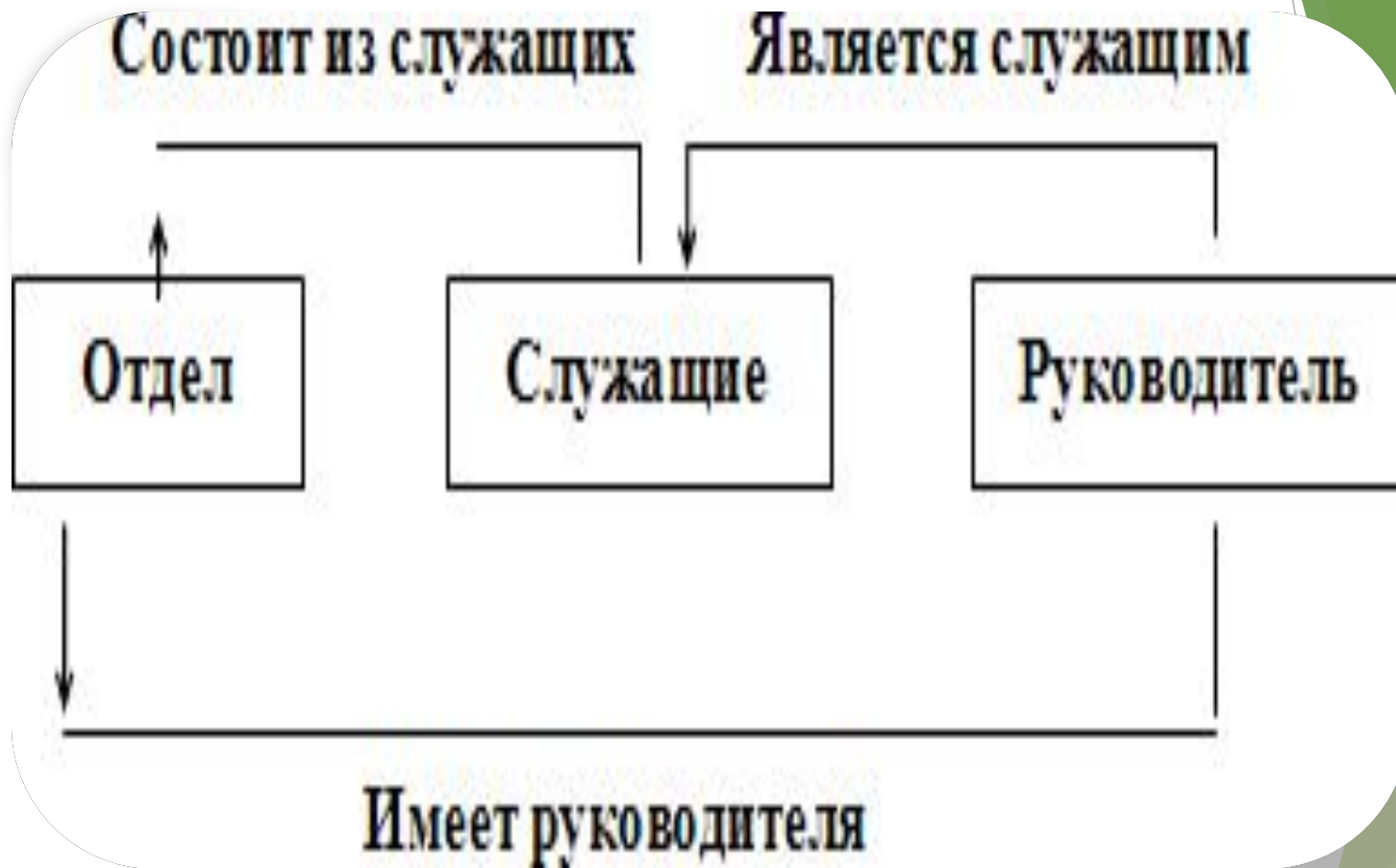


Рис. 2.3 Пример схемы сетевой базы данных.

Манипулирование данными

Вот примерный набор операций манипулирования данными:

- ▶ найти конкретную запись в наборе однотипных записей (например, служащего с именем Иванов);
- ▶ перейти от предка к первому потомку по некоторой связи (например, к первому служащему отдела 625);
- ▶ перейти к следующему потомку в некоторой связи (например, от Иванова к Сидорову);
- ▶ перейти от потомка к предку по некоторой связи (например, найти отдел, в котором работает Сидоров);
- ▶ создать новую запись;
- ▶ уничтожить запись;
- ▶ модифицировать запись;
- ▶ включить в связь;
- ▶ исключить из связи;
- ▶ переставить в другую связь и т.д.

Ограничения целостности

- ▶ Имеется (необязательная) возможность потребовать для конкретного типа связи отсутствие потомков, не участвующих ни в одном экземпляре этого типа связи (как в иерархической модели).