

Содержание:

image not found or type unknown



Введение

Иерархические базы данных могут быть представлены как дерево, состоящее из объектов различных уровней. Верхний уровень занимает один объект, второй — объекты второго уровня и т. д.

Между объектами существуют связи, каждый объект может включать в себя несколько объектов более низкого уровня. Такие объекты находятся в отношении предка (объект более близкий к корню) к потомку (объект более низкого уровня), при этом возможна ситуация, когда объект-предок не имеет потомков или имеет их несколько, тогда как у объекта-потомка обязательно только один предок. Объекты, имеющие общего предка, называются близнецами.

1. Примеры

Например, если иерархическая база данных содержала информацию о покупателях и их заказах, то будет существовать объект «*покупатель*» (родитель) и объект «*заказ*» (дочерний). Объект «*покупатель*» будет иметь указатели от каждого заказчика к физическому расположению заказов покупателя в объект «*заказ*».

В этой модели запрос, направленный вниз по иерархии, прост (например: какие заказы принадлежат этому покупателю); однако запрос, направленный вверх по иерархии, более сложен (например, какой покупатель поместил этот заказ). Также, трудно представить не-иерархические данные при использовании этой модели.

Иерархической базой данных является файловая система, состоящая из корневой директории, в которой имеется иерархия поддиректорий и файлов.

2. Иерархическая модель данных

Первые системы управления базами данных использовали иерархическую модель данных, и во времени их появление предшествует появлению сетевой модели.

3. Структурная часть иерархической модели

Основными информационными единицами в иерархической модели данных являются сегмент и поле. Поле данных определяется как наименьшая неделимая единица данных, доступная пользователю. Для сегмента определяются тип сегмента и экземпляр сегмента. Экземпляр сегмента образуется из конкретных значений полей данных. Тип сегмента — это поименованная совокупность входящих в него типов полей данных.

Как и сетевая, иерархическая модель данных базируется на графовой форме построения данных, и на концептуальном уровне она является просто частным случаем сетевой модели данных. В иерархической модели данных вершине графа соответствует тип сегмента или просто сегмент, а дугам — типы связей предок — потомок. В иерархических структурах сегмент — потомок должен иметь в точности одного предка.

Иерархическая модель представляет собой связный неориентированный граф древовидной структуры, объединяющий сегменты. Иерархическая БД состоит из упорядоченного набора деревьев.

4. Преобразование концептуальной модели в иерархическую модель данных

Преобразование концептуальной модели в иерархическую структуру данных во многом схоже с преобразованием ее в сетевую модель, но и имеет некоторые отличия в связи с тем, что иерархическая модель требует организации всех данных в виде дерева.

Преобразование связи типа «один ко многим» между предком и потомком осуществляется практически автоматически в том случае, если потомок имеет одного предка, и происходит это следующим образом. Каждый объект с его

атрибутами, участвующий в такой связи, становится логическим сегментом. Между двумя логическими сегментами устанавливается связь типа «один ко многим». Сегмент со стороны «много» становится потомком, а сегмент со стороны «один» становится предком.

Ситуация значительно усложняется, если потомок в связи имеет не одного, а двух и более предков. Так как подобное положение является невозможным для иерархической модели, то отражаемая структура данных нуждается в преобразованиях, которые сводятся к замене одного дерева, например, двумя (если имеется два предка). В результате такого преобразования в базе данных появляется избыточность, так как единственно возможный выход из этой ситуации — дублирование данных.

5. Управляющая часть иерархической модели

В рамках иерархической модели выделяют языковые средства описания данных (ЯОД) и средства манипулирования данными (ЯМД). Каждая физическая база описывается набором операторов, обуславливающих как ее логическую структуру, так и структуру хранения БД. При этом способ доступа устанавливает способ организации взаимосвязи физических записей.

Определены следующие способы доступа:

- иерархически последовательный;
- иерархически индексно-последовательный;
- иерархически прямой;
- иерархически индексно-прямой;
- индексный.

Помимо задания имени БД и способа доступа описания должны содержать определения типов сегментов, составляющих БД, в соответствии с иерархией, начиная с корневого сегмента. Каждая физическая БД содержит только один корневой сегмент, но в системе может быть несколько физических БД.

Среди операторов манипулирования данными можно выделить операторы поиска данных, операторы поиска данных с возможностью модификации, операторы модификации данных. Набор операций манипулирования данными в иерархической БД невелик, но вполне достаточен.

6. Примеры типичных операторов поиска данных

- найти указанное дерево БД;
- перейти от одного дерева к другому;
- найти экземпляр сегмента, удовлетворяющий условию поиска;
- перейти от одного сегмента к другому внутри дерева;
- перейти от одного сегмента к другому в порядке обхода иерархии.

Примеры типичных операторов поиска данных с возможностью модификации:

- найти и удержать для дальнейшей модификации единственный экземпляр сегмента, удовлетворяющий условию поиска;
- найти и удержать для дальнейшей модификации следующий экземпляр сегмента с теми же условиями поиска;
- найти и удержать для дальнейшей модификации следующий экземпляр для того же родителя.

Примеры типичных операторов модификации иерархически организованных данных, которые выполняются после выполнения одного из операторов второй группы (поиска данных с возможностью модификации):

- вставить новый экземпляр сегмента в указанную позицию;
- обновить текущий экземпляр сегмента;
- удалить текущий экземпляр сегмента.

В иерархической модели автоматически поддерживается целостность ссылок между предками и потомками. Основное правило: никакой потомок не может существовать без своего родителя.

7. Известные иерархические СУБД

- Типичным представителем (наиболее известным и распространенным) является Information Management System (IMS) фирмы IBM. Первая версия появилась в 1968 г.
- Time-Shared Date Management System (TDMS) компании Development Corporation;

- Mark IV Multi - Access Retrieval System компании Control Data Corporation;
- System 2000 разработки SAS-Institute;
- Серверы каталогов, такие, как LDAP и Active Directory (допускают чёткое представление в виде дерева)
- По принципу иерархической БД построены иерархические файловые системы и Реестр Windows.
- InterSystems Caché
- Google App Engine Datastore API

Заключение.

Итогом написания дипломной работы явилось создание программного продукта «Системы автоматизированного контроля знаний студентов».

Были решены следующие поставленные передо мной задачи:

- дан обзор современному состоянию теории баз данных, основным моделям СУБД, применяемым в ПК;
- изучены принципы функционирования и основные возможности технологии OLE;
- разработан способ отображения реляционных структур данных в иерархическом виде;
- дополнен стандартный компонент Delphi OLEContainer возможностью сохранения битового изображения на его поверхности.

Программа контроля знаний TEST, которая рассматривалась в 5 главе, работает под управлением операционной системы Windows 95. Справочная система позволит легко и быстро научиться работать с системой TEST.

Это удобное добавление к традиционным методам контроля, повышающее эффективность усвоения предмета студентом. Система состоит из трёх связанных между собой частей. Первая часть это программа- редактор вопросов, позволяющая преподавателю создавать индивидуальную базу данных вопросов по своим дисциплинам. Вторая часть представляет собой тестирующую программу, предназначенную для студентов. Третья часть предназначена для преподавателя, представляет собой статистику прохождения теста, настройку параметров, с помощью этой программы преподаватель может проанализировать результаты

прохождения теста и сделать соответствующие выводы.

Таким образом, эта система может использоваться преподавателями, вне зависимости от дисциплины и одинаково подходит как для естественно-научных, так и для гуманитарных предметов.

Список литературы.

В.В. Аладьев, Ю.Я. Хунт, М.Л. Шишаков. Основы информатики. Учебное пособие. – Москва. 1999.

В.А. Извозчиков, А. Д. Ревунов. Электронно-вычислительная техника на уроках физики в средней школе. - М.: Просвещение, 1988.

А. А Жуков, Л.А Федякина. “Система контроля знаний TSTST”.

// Информатика и образование. -1997.- №2.

А.А. Ездов. Лабораторные работы по физике с использованием компьютерной модели. // Информатика и образование. -1996.- №1.

М.Г. Ермаков, Л.Е. Андреева. Вопросы разработки тестирующих программ. // Информатика и образование. -1997.- №3.

В. М. Карнаухов. Система контроля знаний. // Информатика и образование. -1995.- №6.

В. М. Казиев. Системно-алгебраический подход к основам информатики. // Информатика и образование. -1996.- №4.

Е.А. Ерёмин. Почему система интересна для образования. // Информатика и образование. -1997.- №1.

П. Дарахвелидзе, Е. Марков. Delphi - среда визуального программирования. — BHV Санкт-Петербург: 1996.

Джон Матчо, Дэвид Р. Фолкнер. Delphi . — Бином М. 1996.

Тодд Миллер, Дэвид Пауэл и др. Использование Delphi 3 . – Диалектика Киев. 1997.

В.В. Фаронов. Паскаль и Windows. – Москва. 1995.

Механизмы Windows 3.1. – Энтроп Москва. 1994