

Ієрархічна модель даних

План

- 1. Ієрархічна структура даних
- 2. Операції над ієрархічною структурою
- 3. Вибірання даних
- 4. Маніпулювання даними

Ієрархічна структура даних

Ієрархічна структура даних

- Ієрархічна модель даних уперше була задіяна в системі IMS (Information Management System — інформаційна керуюча система) у межах проекту висадки на Місяць.
- Реалізовані функції :
 - мови визначення та маніпулювання даними
 - опис і підтримка обмежень цілісності
 - паралелізм
 - відновлення
 - механізми ефективної обробки запитів.

Ієрархічна структура даних

- Ієрархічна структура даних визначається ієрархічною впорядкованістю своїх компонентів (або вузлів), тобто кожен вузол має не більше одного «батька» - старшого за ієрархією вузла.
- Структура складається зі *схем елементів даних* (описова інформація) та їхніх *екземплярів*.
- Схема задає логічну структуру (або тип) елементу даних, а екземпляр — його значення.

Ієрархічна структура даних

- Елементарним значенням структури є поймаєменоване *поле даних*, а його екземпляр - це елементарне значення.
- *Схема сегмента* (яку називатимемо також просто *сегментом*) — це поймаєменована впорядкована сукупність імен полів. Сегмент є одиницею доступу до даних ієрархічної структури під час взаємодії зовнішньої та оперативної пам'яті.
- *Екземпляр сегмента* - впорядкована сукупність значень полів.

Ієрархічна структура даних

- *Ієрархічна схема даних* — це ієрархічно впорядкована сукупність сегментів, що має такі властивості:
 1. на найвищому рівні ієрархії розташований єдиний сегмент, що називається *кореневим*;
 2. кожен інший сегмент, окрім кореневого, зв'язаний з одним і тільки одним сегментом вищого рівня, який є для цього сегмента *батьківським* (початковим);
 3. сегмент може бути зв'язаний з одним або кількома сегментами нижчого рівня, які називаються *дочірніми* (породженими);
 4. сегменти, що підпорядковані одному батьківському сегменту, називаються *близнюками*;
 5. сегменти, що не мають дочірніх, вважаються *лишковими*, або їх ще називають *листками*.

Ієрархічна структура даних

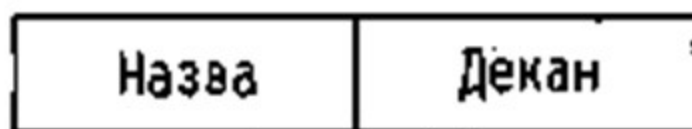
- Ієрархічний шлях (або просто шлях) — це послідовність сегментів, починаючи з кореневого, де кожний попередній є «батьком» наступного. Рівень сегмента визначається як кількість сегментів, що містяться на шляху, який веде від кореня до даного сегмента.

Ієрархічна структура даних

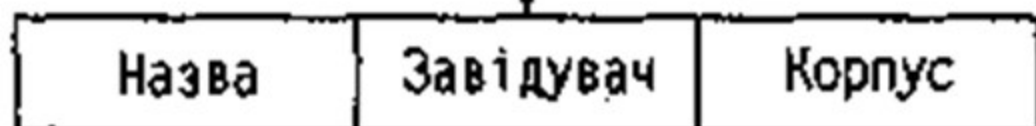
- Для ієрархічної схеми використовується така графічна схема:
- *Кожний сегмент зображується у вигляді поймаєного прямокутника. У середині прямокутника записуються імена полів.*
- *Ієрархічний зв'язок між сегментами позначається лініями зі стрілками, що проведені від батьківського сегмента до дочірнього. Батьківські сегменти, як правило, розміщують над дочірніми.*

Ієрархічна структура даних

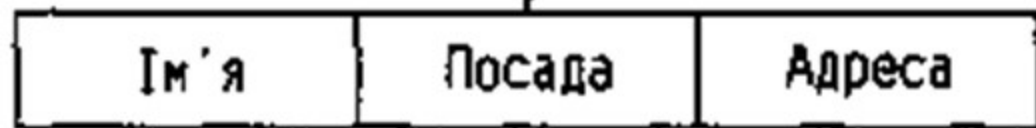
ФАКУЛЬТЕТ



КАФЕДРА

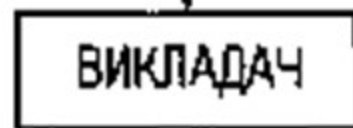
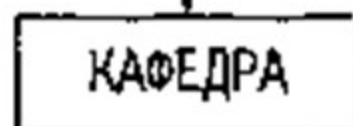


ВИКЛАДАЧ



а

ФАКУЛЬТЕТ



б

Ієрархічна структура даних

- *Екземпляр ієрархічної схеми даних складається з одного екземпляра кореневого сегмента і, можливо, кількох екземплярів дочірніх сегментів для кожного екземпляра батьківського сегмента.*
- Припускається існування таких зв'язків між екземплярами сегментів:
- кожний екземпляр будь-якого сегмента підпорядкований одному екземпляру батьківського сегмента;

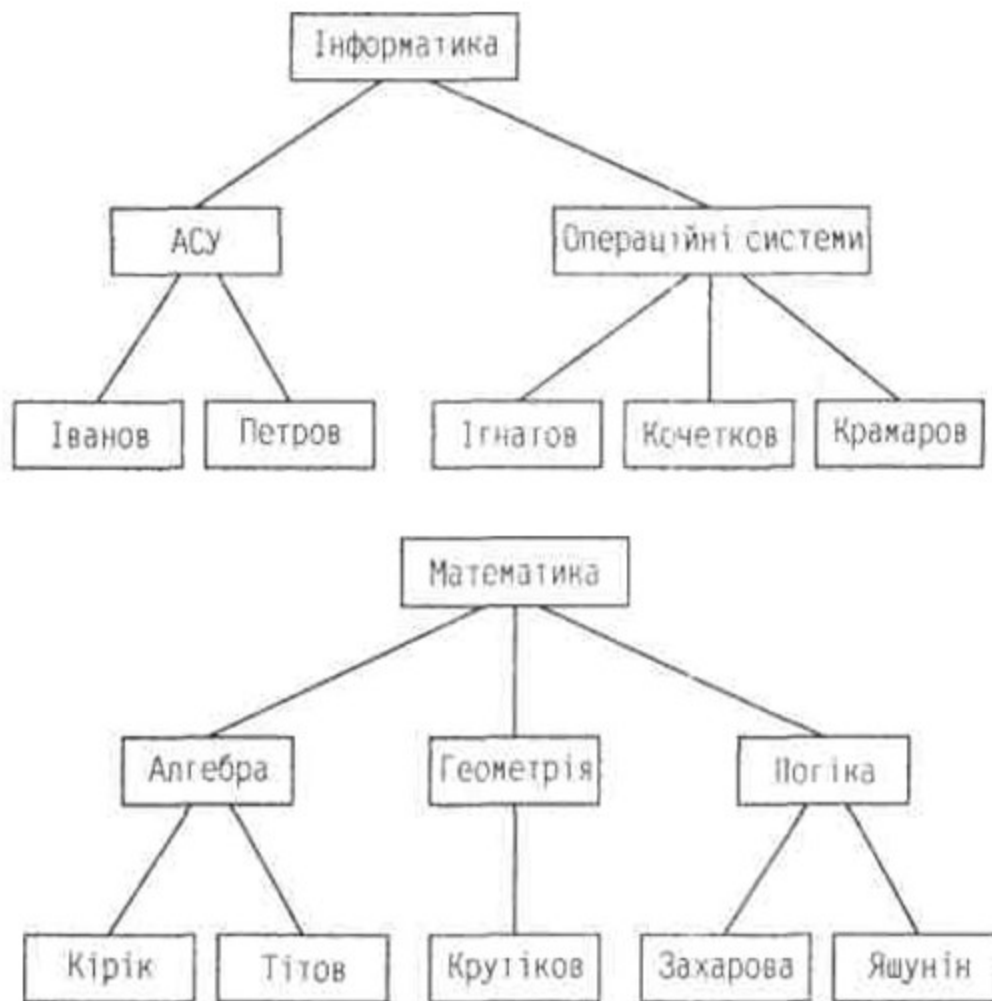
Ієрархічна структура даних

- кожний екземпляр сегмента зв'язаний (підпорядковує собі) з усіма екземплярами дочірніх сегментів;
- екземпляри одного сегмента, зв'язані з одним екземпляром батьківського сегмента, можуть бути зв'язані між собою в ланцюжок, що дає змогу виконувати їхнє послідовне перебирання у межах усіх сегментів, породжених з одного початкового.

Схема (а) та її екземпляри (б) в ієрархічній структурі даних



а



б

Ієрархічна схема з розгалуженням



Ієрархічна структура даних

- *Ієрархічна структура даних* — це сукупність ієрархічної схеми даних та всіх можливих екземплярів цієї схеми.
- Сукупність ієрархічних структур, даних називається *ієрархічною базою даних*.

Операції над ієрархічною структурою

Операції над ієрархічною структурою

- Операції на групи:
- операції пошуку (чи вибирання)
- операції оновлення даних (або маніпулювання ними).

Операції над ієрархічною структурою

- У результаті успішного виконання пошукових операцій визначається поточний екземпляр сегмента, який може відігравати роль стартової позиції для операцій маніпулювання; спочатку поточним є кореневий екземпляр сегмента.
- Пошук необхідного сегмента відбувається за - «навігаційним» принципом.

Вибирання даних

Вибирання даних

- Необхідний екземпляр сегмента вибирається в ієрархічній структурі за допомогою команди «навігації» структурою.
- Екземпляр сегмента є одиницею навігації.
- Для визначення необхідного сегмента навігації може накладатися умова на значення полів сегмента.
- Відносно поточного сегмента можна переміщуватися ієрархічною структурою ввверх, вниз і вбік.

Вибирання даних

- Порядок переміщення встановлюється, починаючи з кореня або будь-якого первинного сегмента. Далі рухатись можна зверху донизу і зліва направо.
- Доступ до даних здійснюється за допомогою команди GET.
- GET UNIQUE – пряме вибирання. Вибирання першого сегмента вказаного типу, що відповідає умові пошуку. За допомогою цієї операції фіксується початкова позиція для подальшої навігації ієрархією.
- GET NEXT – послідовне вибирання. Вибирання сегмента, розташованого відразу за поточним, згідно з наявним порядком розташування сегментів у базі даних. Задаючи різні значення параметрів команди, можна отримати:

Вибирання даних

- сегмент, який розташований за поточним (при цьому тип сегмента може виявитися довільним);
- наступний сегмент заданого типу;
- наступний сегмент заданого типу, що відповідає заданій умові.
- GET NEXT WITHIN PARENT — послідовне вибирання в межах поточного батьківського сегмента.

Приклад

- ФАКУЛЬТЕТ (Назва. Декан)
- КАФЕДРА (Назва. Завідувач. Корпус)
- ВИКЛАДАЧ (Ім'я. Посада. Адреса)

Команда GET UNIQUE має такий синтаксис:

GET UNIQUE <тип сегмента>

[WHERE <умова>]

Приклад

Перший екземпляр кафедри в корпусі 5 знаходиться у такий спосіб:

- GET UNIQUE КАФЕДРА
- WHERE КАФЕДРА.Корпус = 5;

Знайдений сегмент стає поточним. Для знаходження професора кафедри АСУ факультету інформатики слід записати:

- GET UNIQUE ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва = "Інформатики" AND
- КАФЕДРА.Назва = "АСУ" AND
- ВИКЛАДАЧ.Посада = "професор";

Приклад

Перший екземпляр кафедри в корпусі 5 знаходиться у такий спосіб:

- GET UNIQUE КАФЕДРА
- WHERE КАФЕДРА.Корпус = 5;

Знайдений сегмент стає поточним. Для знаходження професора кафедри АСУ факультету інформатики слід записати:

- GET UNIQUE ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва = "Інформатики" AND
- КАФЕДРА.Назва = "АСУ" AND
- ВИКЛАДАЧ.Посада = "професор";

Приклад

- Якщо необхідно знайти кілька сегментів, що відповідають умові пошуку, слід скористатися командою GET NEXT (вона має синтаксис, аналогічний GET UNIQUE).
- GET UNIQUE ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва = "інформатики" AND
- КАФЕДРА.Назва = "математики" AND
- ВИКЛАДАЧ.Посада = "професор";
- NT: GET NEXT ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва = "інформатики" AND
- КАФЕДРА.Назва = "математики" AND
- ВИКЛАДАЧ.Посада = "професор";
- GOTO NT

Маніпулювання даними

Маніпулювання даними

- Маніпулювання даними передбачає додавання, заміну та видалення екземплярів сегментів специфікованих типів.
- Додати новий екземпляр сегмента можна лише за умови, що в ієрархічній структурі вже є екземпляр його батьківського сегмента. У команді додавання INSERT задається повний ієрархічний шлях до цього батьківського сегмента, тип сегмента, що підлягає додаванню, та дані екземпляра сегмента, який додається.

Маніпулювання даними

- приклад операції додавання нового екземпляра сегмента КАФЕДРА:
- КАФЕДРА.Назва - "САПР";
- КАФЕДРА.Завідуючий = "Іванов";
- КАФЕДРА.Корпус = 3;
- INSERT КАФЕДРА
- WHERE ФАКУЛЬТЕТ.Назва = "інформатики";
-
- Додати сегмент ВИКЛАДАЧ можна так:
- ВИКЛАДАЧ.ім'я = "Петров";
- ВИКЛАДАЧ.Посада - "асистент";
- ВИКЛАДАЧ.Адреса - "Проспект Миру 13/17";
- INSERT ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва="інформатики" AND КАФЕДРА.Назва = "САПР";

Маніпулювання даними

- Щоб замінити сегмент, його слід спочатку вибрати за допомогою команди GET HOLD, яка вказує, що шуканий сегмент буде модифікований, потім змінити у знайденого сегмента певні поля й **застосувати команду REPLACE**. Наприклад, у такий спосіб можна змінити сегмент «адреса викладача Петрова з кафедри САПР»:
 - GET HOLD UNIQUE ВИКЛАДАЧ
 - WHERE ФАКУЛЬТЕТ.Назва = "інформатики" AND
 - КАФЕДРА.Назва = "САПР" AND
 - ВИКЛАДАЧ.Ім'я = "Петров";
 - ВИКЛАДАЧ.Адреса = "вул. Теремківська 1/19";
 - REPLACE;

Маніпулювання даними

- Видалення сегмента здійснюється за тією ж схемою, що й заміна: потрібно віднайти сегмент за допомогою команди GET HOLD, потім видалити його командою DELETE.
- Наприклад, видалення інформації про викладача Петрова кафедри САПР факультету інформатики виконується так:
- GET HOLD UNIQUE ВИКЛАДАЧ
- WHERE ФАКУЛЬТЕТ.Назва = "інформатики" AND
- КАФЕДРА.Назва = "САПР" AND
- ВИКЛАДАЧ.Ім'я = "Петров";
- DELETE;

Маніпулювання даними

- Видалення екземпляра сегмента приводить до видалення всіх екземплярів його дочірніх сегментів, оскільки в ієрархічній структурі жодний екземпляр сегмента не може існувати без свого «батька».
- Отже, якщо ви видаляєте екземпляр сегмента КАФЕДРА, то автоматично видаляються всі екземпляри сегментів, що відповідають викладачам даної кафедри