|  |  |  |
| --- | --- | --- |

# Một số khái niệm

## Cơ sở dữ liệu

| | - Dữ liệu: là những sự kiện thô hoặc chưa được xử lý  + Ví dụ: 49, Con mèo, emoji,...  - Thông tin: là những dữ liệu đã được xử lý  + Vd: Có 49 emoji hình con mèo khác nhau được tích hợp trên hệ thống  - CSDL: là 1 tập hợp các dữ liệu liên quan tới nhau  + Vd: CSDL về các học sinh của trường Tiểu học Vân Tảo, CSDL về lượng sách bán ra của Nhã Nam trong tháng 12,...  - Siêu dữ liệu (meta data):  + Là mô tả đầy đủ của 1 CSDL  + Những thông tin như: định dạng lưu trữ, kiểu dữ liệu mà ta sẽ lưu trữ trong CSDL hoặc các ràng buộc đối với dữ liệu, tất cả những thông tin này được gọi là siêu dữ liệu |  | | --- | --- |   - Ưu điểm:  + Bảo đảm được tính nhất quán cho dữ liệu được lưu trữ: Khi thay đổi dữ liệu của 1 đối tượng, thông tin của các đối tượng liên quan sẽ được cập nhật theo.  + Hạn chế trùng lặp dữ liệu  + Dễ dàng truy cập, thay đổi, thống kê dữ liệu: Chúng ta có thể sử dụng các công cụ trực quan hoặc mã lệnh để truy cập, thay đổi dữ liệu trong CSDL, từ bất kỳ thiết bị nào, bất kỳ lúc nào. Việc tìm kiếm các thông tin trong CSDL cũng diễn ra nhanh chóng nhờ cơ chế cache và index.  + Dễ dàng chia sẻ, phân phối dữ liệu: Chúng ta có thể cấp quyền xem / chỉnh sửa dữ liệu trong CSDL cho các ứng dụng khác nhau, hoặc hủy việc cấp phát quyền này bất kỳ khi nào mà không ảnh hưởng đến dữ liệu đang có trên hệ thống.  - Nhược điểm:  + Dữ liệu được lưu trữ tại 1 nơi, vậy khi có sự cố, toàn bộ hệ thống đều bị ảnh hưởng. Để giảm thiểu thiệt hại này, ta cần sao lưu dữ liệu thường xuyên, hoặc xây dựng để chạy song song nhiều CSDL và đồng bộ chúng với nhau.  + Khi có quá nhiều lượng truy cập đồng thời, CSDL sẽ có khả năng bị quá tải và không phản hồi. Để tránh khả năng xảy ra quá tải, chúng ta cần tính toán để trang bị cấu hình tương xứng với phạm vi ứng dụng. Vd: ta không thể xây dựng 1 CSDL chạy trên máy tính RAM 2GB, CPU 4 lõi 1GHz để phục vụ 10.000(request/second).  + Tranh chấp dữ liệu (Vd: Nhiều người cùng muốn mua 1 món hàng với số lượng có hạn trên 1 trang TMĐT) |
| --- | --- | --- |

### 

## Hệ quản trị CSDL (DataBase Management System)

| Course về Hệ quản trị CSDL: [utube](https://www.youtube.com/watch?v=wClEbCyWryI&list=PLBlnK6fEyqRi_CUQ-FXxgzKQ1dwr_ZJWZ&index=2)  - Là 1 hệ thống (tập hợp các phần mềm) để quản lý các CSDL, cho phép: tạo, bảo trì, lưu trữ, truy xuất CSDL,..  + Ví dụ: MS Access, SQL Server, MySQL,...  - Chức năng:  + Định nghĩa: Xác định kiểu dữ liệu, cấu trúc và ràng buộc (ràng buộc: ví dụ lưu trữ danh sách người trên 18 tuổi thì biến tuổi trong CSDL sẽ có ràng buộc > 18) cho dữ liệu được lưu trữ  + Xây dựng: Quá trình lưu trữ dữ liệu trên một số phương tiện lưu trữ (khi dữ liệu được lưu trữ thì cũng có nghĩa CSDL đang được xây dựng)  + Kiểm soát: Bao gồm các chức năng như truy vấn CSDL để truy xuất dữ liệu cụ thể, cập nhật CSDL và tạo bản báo cáo  + Chia sẻ: Cho phép nhiều người dùng và chương trình có thể đồng thời truy cập vào cùng 1 CSDL  + Bảo vệ CSDL khỏi truy cập trái phép, lỗi phần cứng hoặc phần mềm, và bảo trì CSDL trong 1 thời gian dài  - Đặc trưng của DBMS:  + Bản chất tự mô tả của hệ thống CSDL:   | - Một hệ thống CSDL gồm: CSDL và Siêu dữ liệu (Metadata - Định nghĩa cho CSDL) | | --- |   + Sự phân cách giữa chương trình và dữ liệu, và trừu tượng hóa dữ liệu  + Hỗ trợ nhiều chế độ xem của dữ liệu  + Chia sẻ dữ liệu và xử lý giao dịch nhiều người dùng |
| --- | --- |

| Kiến trúc tổng quát của 1 CSDL  | - Mức trong (mức vật lý):  + Đây là mức lưu trữ CSDL. Tại mức này, vấn đề cần giải quyết là dữ liệu gì và được lưu trữ như thế nào? ở đầu (đĩa từ, băng từ, track, sector,... nào)?  - Mức khái niệm (Mức logic):  + Tại mức này sẽ giải quyết cho câu hỏi CSDL cần phải lưu trữ bao nhiêu loại dữ liệu? Đó là những dữ liệu gì? Mối quan hệ giữa các loại dữ liệu này như thế nào?  - Mức ngoài (Khung nhìn - view)  + Đó là mức của người sử dụng và các chương trình ứng dụng |  | | --- | --- |   CSDL lưu trữ điểm của sinh viên:  + Thông tin sinh viên (mã sv, tên,..)  + Học phần (Mã học phần, tên học phần, số tín chỉ,...)  + Lớp học phần (thời gian học, tên lớp học phần,...)  + Có thể thêm: Tên giảng viên,... |
| --- | --- | --- |

# Các mô hình dữ liệu (Data Models)

| - Mô hình dữ liệu là sự trừu tượng hóa môi trường thực, nó là sự biểu diễn dữ liệu ở mức quan niệm  - Mỗi loại mô hình dữ liệu đặc trưng cho một cách tiếp cận dữ liệu khác nhau của những nhà phân tích - thiết kế CSDL  - Phân loại mô hình dữ liệu:  + Mô hình dữ liệu khái niệm: còn được gọi là mô hình miền, thiết lập các khái niệm và ngữ nghĩa cơ bản của một miền nhất định cho nhiều đối tượng của các bên có liên quan. Các mô hình dữ liệu khái niệm được xây dựng dựa trên kiến trúc tổng thể, thông qua việc sử dụng mô hình mối quan hệ thực tế hoặc lớp UML.  + Mô hình dữ liệu logic: là phương pháp mô hình hóa cơ sở dữ liệu trừu tượng và khái niệm. Mô hình tạo ra một lược đồ hoặc mô hình dữ liệu ngữ nghĩa và các yêu cầu của nó. Thông qua chuyển đổi ngôn ngữ định nghĩa dữ liệu DDL, người dùng có thể dễ dàng chuyển mô hình dữ liệu logic sang mô hình dữ liệu vật lý.  + Mô hình dữ liệu vật lý: giúp người dùng dễ dàng hình dung cấu trúc cơ sở dữ liệu, tự động lấy ra được lược đồ cơ sở dữ liệu tương ứng. Theo đó, mô hình này cho pháp sử dụng cấu hình UML để mô hình hóa dữ liệu cụ thể. |
| --- |

## 

| Mô hình mạng - Được tạo ra để giải quyết những khuyết điểm của mô hình phân cấp  - Mô hình dữ liệu mạng (Network Data Model) - là mô hình được biểu diễn bởi một đồ thị có hướng:  + Mỗi đỉnh là một loại mẫu tin (Record Type)  + Mỗi cung có hướng thể hiện mối liên hệ (Set Type) giữa 2 mẫu tin  - Mô hình dữ liệu mạng vẫn còn tồn tại những hạn chế như số lượng con trỏ lớn, hạn chế trong việc biểu diễn ngữ nghĩa và móc nối giữa các mẫu tin với nhau. | Mô hình phân cấp - Mô hình dữ liệu phân cấp (Hierarchical Data Model) : Là 1 cây, trong đó mỗi nút của cây biểu diễn một loại mẫu tin  - Giữa 2 loại mẫu tin chỉ tồn tại một mối quan hệ duy nhất. Quan hệ giữa các nút cấp trên và nút con là 1-n và n-n  - Ưu:  + Dễ xây dựng và thao tác, phù hợp với các tổ chức phân cấp như tổ chức nhân sự trong các doanh nghiệp  - Khuyết:  + Thường xảy ra tình trạng lặp lại các bản ghi dư thừa, không nhất quán.  - Ví dụ: Xây dựng hệ thống quản lý sinh viên của các trường Đại học tại Việt Nam  + Có 4 loại bản ghi: Trường, Khoa, Ngành, Sinh Viên |
| --- | --- |
| Mô hình thực thể liên kết - Entity - RelationShip Model  - Mô hình này gồm:  + Thực thể (Entity)  + Thuộc tính của thực thể (Entity Attributes)  + Mối quan hệ giữa các thực thể (Entity Relationship) | Mô hình quan hệ (Relational Data Model) - Nền tảng cơ bản của nó là khái niệm lý thuyết tập hợp trên các quan hệ, tức là tập của các bộ giá trị (Value Tuples). Trong mô hình dữ liệu này những khái niệm sẽ được sử dụng gồm: thuộc tính (Attributes), quan hệ (Relation), lược đồ quan hệ (Relation Schema), bộ (Tuple), khóa (Key)  - Mô hình dữ liệu này cùng với mô hình dữ liệu thực thể kết hợp đang được sử dụng rộng rãi trong việc phân tích và thiết kế CSDL hiện nay    - Ưu:  + Khả năng tối ưu hóa đa dạng các xử lý nhờ dựa trên lý thuyết tập hợp và đại số quan hệ  - Nhược:  + Cấu trúc này vẫn chưa linh hoạt và hạn chế trong việc biểu diễn ngữ nghĩa phức tạp của các quan hệ thực tế. |
| Mô hình hướng đối tượng - Mô hình dữ liệu hướng đối tượng (Object Oriented Data Model)  - Dựa trên cách tiếp cận hướng đối tượng đã quen thuộc trong các phương pháp OOP, sử dụng các khái niệm: class, inheritance, multi - inheritance  - Đặc trưng cơ bản của cách tiếp cận này: encapsulation, polymorphism, Reusability  - Ưu:  + Cho phép định nghĩa các kiểu đối tượng phức tạp, nhờ dùng các tích chất đặc trưng của OOP  - Nhược:  + Còn tồn tại là cấu trúc lưu trữ còn phức tạp, có thể cần sử dụng đến nhiều con trỏ. Khả năng tối ưu hóa chưa tốt, còn bị hạn chế trong một vài trường hợp. |  |

# Bài tập

|  |
| --- |

|  |
| --- |

|  |
| --- |

|  |
| --- |

|  |
| --- |

|  |
| --- |

|  |
| --- |