Tên : Nguyễn Minh Quang

MSSV: 20143481

**Câu 1**. Tại sao bạn lựa chọn hệ thống CSDL thay vì lưu dữ liệu trong file quản lý bởi hệ điều hành? Khi nào bạn không nên dùng hệ CSDL?

Bởi vì khi lưu trữ trên file thì sẽ phát sinh một số vấn đề :

* Bộ nhớ sẽ không đủ để lưu trữ nếu dữ liệu quá lớn
* Dữ liệu được chuyển qua lại giữa bộ nhớ chính và bộ nhớ phụ nên nếu dùng hệ điều hành 32 bit thì không đủ.
* Với lưu trữ trên file thì mỗi câu truy vấn phải viết phải viết 1 chương trình đặc biệt riêng.
* Sẽ không đảm bảo được tính nhất quán.
* Khi gặp sự cố thì khôi phục lại dữ liệu một cách khó khăn.
* Gặp vấn đề trong bảo mật và kiểm soát truy cập.

Lợi ích khi dùng hệ thống cơ sở dữ liệu:

* Độc lập dữ liệu và sẽ truy cập hiệu quả hơn.
* Giảm thiểu được thời gian phát triển ứng dụng.
* Hợp nhất được dữ liệu và bảo mật.
* Quản trị dữ liệu một cách thống nhất.
* Truy cập đồng thời và khôi phục khi gặp sự cố.

Không nên dùng cơ sở dữ liệu nếu các lợi ích không quan trọng với ứng dụng ta sử dụng, việc sử dụng tập các file sẽ là giải pháp tốt hơn vì việc duy trì và thực hiện trên hệ thống CSDL sẽ phải trả chi phí cao hơn.

**Câu 2**. Độc lập dữ liệu mức logic (Logical data independence) là gì? Tại sao nó quan trọng?

Độc lập dữ liệu logic là người quản trị có thể thay đổi các cấu trúc trong mức khái niệm mà không ảnh hưởng đến mức vật lý.

Vì những thay đổi này do người quản trị thay đổi cho phù hợp với sự thay đổi của dữ liệu thực tế liên quan đến hệ thống CSDL, view ra cho người dùng xem nên nó rất quan trọng hơn độc lập vật lý.

**Câu 3**. Giải thích sự khác nhau giữa độc lập dữ liệu mức logic và độc lập dữ liệu mức vật lý (physical data independence)? Hãy cho ví dụ minh họa.

Độc lập dữ liệu vật lý là các thay đổi trong mức vật lý để nâng cao hiệu năng lưu trữ mà không ảnh hưởng đến mức khái niệm (người quản trị dữ liệu sẽ không cần quan tâm đến mức vật lý cũng như sự thay đổi của mức này).

Độc lập dữ liệu logic là người quản trị có thể thay đổi các cấu trúc trong mức khái niệm mà không ảnh hưởng đến mức vật lý.

**Câu 4**. Giải thích sự khác biệt giữa lược đồ ý niệm/logic (conceptual/logical schema), lược đồ vật lý/bên trong (Physical/internal schema) và lược đồ ngoài (external schema)

Lược đồ ngoài là những bàn ghi cụ thể, người dùng cũng như người quản trị CSDL sẽ thao tác với mô hình ngoài dùng ngôn ngữ truy vấn dữ liệu.

Lược đồ ý niệm/logic là định nghĩa các mô hình quan hệ.

Lược đồ trong là mô hình, cách thức lưu dữ liệu vật lý của hệ thống CSDL.

**Câu 5**. Trách nhiệm của DBA. Giả sử là DBA không cần quan tâm đến việc thực hiện các câu truy vấn của chính DBA, thì DBA có cần hiểu về tối ưu hóa câu truy vấn không? Tại sao?

Trách nhiệm của DBA:

* Thiết kế lược đồ logic và lược đồ vật lý.
* Điều khiển bảo mật và phân quyền, dữ liệu luôn được toàn vẹn.
* Đảm bảo cho dữ liệu luôn trong tình trạng sẵn sàng và phục hồi lại khi có sự cố xảy ra.
* Sự điều chỉnh dữ liệu cần phải tối ưu hơn.

DBA cần phải hiểu về tối ưu hóa câu truy vấn vì tầm quan trọng DBA rất lớn. Cần phải am hiểu trong nhiều mảng liên quan đến lập trình để đảm bảo cho hệ thống CSDL luôn được hoạt động tốt và am hiểu về tối ưu câu truy vấn để có thể tối ưu hệ thống CSDL 1 cách tốt nhất khi cần thiết.

**Câu 6**. Ông A cần mua một hệ CSDL. Để tiết kiệm chi phí, ông A chỉ mua một hệ CSDL với số tính năng ít nhất có thể. Ông ta lập kế hoạch chỉ chạy nó một mình trên máy PC của ông ấy và không share thông tin với ai cả. Hãy cho biết tính năng nào trong các tính năng dưới đây của DBMS ông A mua nên có và tại sao:

+ Tiện ích bảo mật

+ Kiểm soát đồng thời

+ Khôi phục dữ liệu sau sự cố

+ Cơ chế khung nhìn

+ Ngôn ngữ truy vấn

Ông A nên mua DBMS có các tính năng sau:

* Tiện ích bảo mật, vì không có chức năng này dữ liệu của ông A có thể sẽ bị tấn công.
* Khôi phục dữ liệu sau sự cố, vì để tránh mất dữ liệu khi gặp sự cố.
* Cơ chế khung nhìn, để dễ sử dụng cho ông A.

Không cần “kiểm soát đồng thời” và “ngôn ngữ truy vấn” vì chỉ có một mình ông A sử dụng và ngôn ngữ truy vấn ông A cũng không cần biết vì đó là công việc của DBA quản lý CSDL.

**Câu 7**. Mô tả cấu trúc của một DBMS. Giả sử hệ điều hành của bạn được nâng cấp để hỗ trợ thêm một số chức năng về file (ví dụ khả năng cho phép lưu một chuỗi các bytes lên đĩa). Hãy cho biết lớp nào của DBMS bạn cần phải viết lại để có thể tận dụng ưu điểm của các tính năng mới đó.

Cấu trúc của một DBMS gồm 4 phần:

* Mức nội (interval level): mô tả cấu trúc lưu trữ vật lý của CSDL
* Mức ý niệm (conceptual level):
* Mô tả toàn bộ cấu trúc của cơ sở dữ liệu
* Che dấu chi tiết vật lý
* Tập trung vào mô tả dữ liệu, mối quan hệ, các ràng buộc.
* Mức ngoại(external level): mô tả một phần CSDL ứng với góc nhìn của một nhóm người dùng.

Để có thể tận dụng ưu điểm của tính năng cho phép lưu một chuỗi các byte lên đĩa chúng ta cần phải viết lại mức ngoại.

**Câu 8**. Trả lời các câu hỏi sau:

1. Giao tác (transaction) là gì?

Transaction là một tập hợp các hành động thực hiện một chức năng và chứng chỉ có thể cùng thành công hoặc thất bại.

1. Tại sao một DBMS thực hiện xen kẽ các hành động của các giao dịch khác nhau thay vì thực hiện lần lượt từng giao dịch một ?

Nhằm mục đích tận dụng tối đa thời gian trống của các chương trình trong hệ thống. Nếu chỉ thực hiện từng giao dịch một thì một số chương trình sẽ rảnh nhưng không có công việc làm, từ đó hiệu suất kém.

1. Một user phải chắc chắn điều gì để đảm bảo tính nhất quán giữa một giao dịch và CSDL ? Một DBMS nên chắc chắn điều gì để đảm bảo tính nhất quán giữa thực hiện đồng thời nhiều giao dịch và CSDL.

DBMS phải đảm bảo rằng giao dịch được thực thi toàn bộ và độc lập với các giao dịch khác. Một tính chất quan trọng của giao dịch trong DBMS là tính nguyên tử và có tính cô lập. Các giao dịch hoặc là thành công hoặc thất bại hoàn toàn. Điều này đảm bảo tính nhất quán dữ liệu.

1. Giải thích về nghi thức khóa 2 giai đoạn nghiêm ngặt (the strict two-phase locking protocol).

Khóa hai pha nghiêm ngặt sử dụng các khóa chia sẻ và độc quyền để bảo vệ dữ liệu. Một giao dịch phải nắm giữ tất cả các khóa cần thiết trước khi thực thi và không giải phóng bất cứ khóa nào trước khi giao dịch kết thúc hoàn toàn.

1. Tính chất WAL là gì và tại sao nó quan trọng ?

Đặc tính WAL ảnh hưởng đến chiến lược logging trong DBMS. WAL là viết tắt Write – Ahead Log, có nghĩa là mỗi thao tác viết lên cơ sở dữ liệu đều phải được ghi vào log(trên đĩa) trước khi nó thực sự xảy ra trong cơ sở dữ liệu. Điều này bảo vệ cơ sở dữ liệu nếu có sự cố xảy ra trong quá trình giao dịch đang thực hiện. Bằng việc ghi lại những thay đổi vào trong log trước khi những thay đổi này diễn ra, CSDL biết được phải làm lại những thao tác nào để khôi phục sự cố. Ngược lại, nếu việc ghi vào log diễn ra sau khi CSDL sẽ không thể phát hiện được có những thay đổi nào đã xảy ra trong quá trình khôi phục sự cố.

1. Định nghĩa Cơ sở dữ liệu (database)?

Cơ sở dữ liệu là một bộ sưu tập dữ liệu có hệ thống, được lưu trữ bằng điện tử. Nó có thể chứa bất kỳ loại dữ liệu nào, bao gồm từ, số, hình ảnh, video và tệp. Bạn có thể sử dụng phần mềm được gọi là hệ thống quản lý cơ sở dữ liệu (DBMS) để lưu trữ, truy xuất và chỉnh sửa dữ liệu. Trong các hệ thống máy tính, cơ sở dữ liệu từ cũng có thể tham khảo bất kỳ DBMS, đến hệ thống cơ sở dữ liệu, hoặc một ứng dụng liên kết với cơ sở dữ liệu.

1. Các chức năng của hệ quản trị CSDL

Chức năng của hệ quản trị cơ sở dữ liệu là Cung cấp cách tạo lập CSDL, Cung cấp cách cập nhật dữ liệu, tìm kiếm và kết xuất thông tin và Cung cấp công cụ kiểm soát điều khiển việc truy cập vào CSDL.

1. Ngôn ngữ định nghĩa dữ liệu là gì? Cho ví dụ.

Ngôn ngữ định nghĩa dữ liệu là ngôn ngữ máy tính được sử dụng để tạo và sửa đổi cấu trúc của các đối tượng cơ sở dữ liệu trong cơ sở dữ liệu. Các đối tượng cơ sở dữ liệu này bao gồm các khung nhìn, lược đồ, bảng, chỉ mục…

Ví dụ: Tạo bảng DIEMTHI với khoá chính là tập bao gồm hai cột MAMONHOC và MASV CREATE TABLE diemthi ( mamonhoc NVARCHAR(10) NOT NULL , masv NVARCHAR(10) NOT NULL , diemlan1 NUMERIC(4, 2), diemlan2 NUMERIC(4, 2), CONSTRAINT pk\_diemthi PRIMARY KEY(mamonhoc,masv) )

1. Ngôn ngữ thao tác dữ liệu là gì? Cho ví dụ

Ngôn ngữ thao tác dữ liệu là gì? Ngôn ngữ thao tác dữ liệu được viết tắt là DML là ngôn ngữ để người dùng diễn tả yêu cầu cập nhật hay khai thác thông tin. Trong đó, thao tác dữ liệu gồm: Cập nhật (nhập, sửa, xóa dữ liệu) và khai thác (sắp xếp, tìm kiếm, kết xuất báo cáo,…).

Ví dụ: INSERT INTO CONGTY (MCT, tencty, nuoc) VALUES (1, 'Air France', 'Phap');

INSERT INTO CONGTY VALUES (3, 'Qantas', 'Uc');

UPDATE CONGTY SET tencty = 'VIETNAM AIRLINE'

WHERE tencty ='Easy Jet';

DELETE FROM CONGTY WHERE tencty = 'Easy Jet' AND nuoc = 'EU';

1. Ngôn ngữ kiểm soát dữ liệu và ngôn ngữ kiểm soát transaction? cho ví dụ.

Ngôn ngữ kiểm soát dữ liệu (DCL) là một tập hợp con của Ngôn ngữ truy vấn có cấu trúc (SQL) và cho phép các quản trị viên cơ sở dữ liệu cấu hình truy cập bảo mật cho các cơ sở dữ liệu quan hệ. Nó bổ sung cho Ngôn ngữ định nghĩa dữ liệu (DDL), được sử dụng để thêm và xóa các đối tượng cơ sở dữ liệu và Ngôn ngữ thao tác dữ liệu (DML) được sử dụng để truy xuất, chèn và sửa đổi nội dung của cơ sở dữ liệu.

DCL là tập con của SQL đơn giản nhất, vì nó chỉ bao gồm ba lệnh: GRANT, REVOKE và DENY. Kết hợp, ba lệnh này cung cấp cho quản trị viên sự linh hoạt để thiết lập và loại bỏ quyền cơ sở dữ liệu theo kiểu cực kỳ chi tiết.

Ví dụ: **GRANT SELECT ON employee TO user1**.

Lệnh GRANT trên cấp quyền SELECT cho user1 trong bảng danh sách nhân viên

TCL (Ngôn ngữ kiểm soát giao dịch) được sử dụng để kiểm soát trạng thái của giao dịch mà từ đó chúng được khởi chạy. Kiểm soát một giao dịch đòi hỏi khả năng thực hiện giao dịch đó, khôi phục các sửa đổi, xác định SAVEPOINT để quản lý ROLLBACK, v.v.

Ví dụ: SAVEPOINT savepoint\_name;

ROLLBACK TO savepoint\_name;

1. Nhiệm vụ của công cụ kiểm soát, điều khiển truy cập vào CSDL?

+ Duy trì tính nhất quán của CSDL

+ Khôi phục CSDL khi có sự cố

+ Phát hiện và ngăn chặn sự truy cập không được phép

+ Tổ chức và điều khiển các truy cập đồng thời

+ Quản lí các mô tả dữ liệu.

1. Các loại đối tượng sử dụng CSDL?

Có nhiều người tham gia vào sự hoạt động của cơ sở dữ liệu với các vai trò khác nhau, với những mục đích khác nhau, thực hiện những nhiệm vụ, những công việc khác nhau. Ta có thể chia các đối tượng này làm hai nhóm.

Nhóm thứ nhất là những người tương tác thường xuyên với CSDL (actors on the scene). Phần lớn công việc của nhóm này liên quan đến bản thân cơ sở dữ liệu. Thuộc nhóm này ta có những đối tượng chính sau:

* Quản trị cơ sở dữ liệu (database administrator hay DBA) có nhiệm vụ cấp quyền truy cập CSDL cho những người có liên quan và giám sát hoạt động của những người này, quản lý các phần cứng và phần mềm tương ứng, chịu trách nhiệm về tình trạng vận hành, về hỏng hóc, về an toàn của hệ CSDL.
* Thiết kế cơ sở dữ liệu (database designer hay DBD) có nhiệm vụ xác định những dữ liệu nào cần được lưu trữ, cấu trúc của những dữ liệu ấy, phương pháp thể hiện và lưu trữ các dữ liệu này. Những nhân viên này cũng cần nắm bắt yêu cầu của những đối tượng khác (chủ yếu thuộc nhóm thứ nhất) để thiết kế cơ sở dữ liệu đáp ứng được một cách tốt nhất những yêu cầu ấy, thí dụ tạo ra những khung hiển thị dễ sử dụng, phù hợp, đáp ứng nhanh nhu cầu của người dùng, tạo ra các chương trình xử lý dữ liệu (tính toán, vẽ biểu đồ). Vai trò của những người thiết kế dữ liệu đặc biệt quan trọng trong giai đoạn đầu, khi thiết kế và bắt đầu triển khai. Khi CSDL đã hoạt động ổn định, họ có thể chuyển sang các bộ phận khác.
* Người dùng cuối (end user) là những người cần tìm kiếm thông tin và cập nhật thông tin. Đây là đối tượng phục vụ chủ yếu của CSDL. Họ có thể là một người bên ngoài tổ chức như khách hàng, hay bên trong tổ chức như nhân viên phòng kế toán cần dữ liệu để lập bảng lương, hay nhân viên kho cần cập nhật nguyên liệu hay hàng hóa tồn kho. Họ cũng có thể là những chuyên gia, sử dụng dữ liệu để phân tích tình hình kinh doanh hay năng lực tài chính chẳng hạn.

Nhóm thứ hai thường được gọi là những nhân vật hậu trường (worker behind scene), làm việc chủ yếu với HQTCSDL và các chương trình ứng dụng. Nhóm này gồm có:

* Những người thiết kế và xây dựng các thành phần chính của HQTCSDL như các mođun để thực hiện các chức năng như truy vấn, tạo từ điển, truy xuất dữ liệu, thực hiện các giao tiếp với hệ điều hành, với các trình biên dịch của các ngôn ngữ, xây dựng các chương trình ứng dụng.
* Những người tạo nên các công cụ, các tiện ích bổ sung nhằm mở rộng, hay nâng cao các tính năng của hệ CSDL. Các công cụ, tiện ích này có thể là những thành phần bổ sung được cung cấp kèm theo phần mềm, hoặc ở dạng tùy chọn và được mua và/hoặc cài đặt riêng biệt.
* Những người cung cấp các dịch vụ có liên quan như bảo trì phần cứng, sửa chữa các hỏng hóc.

1. Hệ quản trị CSDL (DBMS) là gì?

DBMS là phần mềm được thiết kế để có thể xác định, tiến hành các thao tác, truy xuất và quản lý dữ liệu trong Cơ sở dữ liệu. DBMS thường có khả năng tự thao tác với dữ liệu, định dạng dữ liệu, tên trường, cấu trúc bản ghi và cấu trúc tệp. Nó cũng xác định các quy tắc để xác nhận và thao tác với dữ liệu.

Với DBMS, người dùng có thể thao tác sửa/xóa/thêm dữ liệu mà không còn cần các chương trình khung. Các ngôn ngữ lập trình truy vấn như SQL thường đi kèm với DBMS để lập trình viên dễ dàng tương tác với dữ liệu họ cần.

1. Kể tên một vài DBMS thông dụng

* MySQL
* SQL
* Server
* Oracle
* dBASE
* FoxPro

1. Một hệ CSDL?

Một Hệ CSDL gồm CSDL và hệ quản trị CSDL và khai thác CSDL đó. Ngoài ra, còn có các phần mềm ứng dụng được xây dựng dựa trên hệ quản trị CSDL để việc khai thác CSDL trở lên thuận tiện hơn, đáp ứng các yêu cầu đa dạng của người dùng.

1. Siêu dữ liệu (meta-data)?

Siêu dữ liệu (metadata) là dạng dữ liệu mô tả thông tin chi tiết về dữ liệu. Trong cơ sở dữ liệu, metadata là các sửa đổi dạng biểu diễn khác nhau của các đối tượng trong cơ sở dữ liệu. Trong cơ sở dữ liệu quan hệ thì metadata là các định nghĩa của bảng, cột, cơ sở dữ liệu, view và nhiều đối tượng khác. Trong kho dữ liệu, metadata là dạng định nghĩa dữ liệu như: bảng, cột, một báo cáo, các luật doanh nghiệp hay những quy tắc biến đổi. Metadata bao quát tất cả các phương diện của kho dữ liệu. Trong quản lý tập tin, metadata chứa các thông tin thuộc tính của tập tin đó như: tên tập tin, mô tả tóm tắt, kích cỡ, ngày tạo ra,...

1. Ưu điểm của DBMS

* Cung cấp nhiều phương thức để lưu trữ và truy xuất các dữ liệu có nhiều định dạng khác nhau bằng cách sử dụng ngôn ngữ truy vấn (Query Language).
* Dễ dàng được duy trì.
* DBMS hỗ trợ nhiều ứng dụng có thể sử dụng trong cùng một hệ cơ sở dữ liệu với thời gian phát triển và duy trì được tối ưu nhất.
* Bảo mật và toàn vẹn dữ liệu với hoạt động cập nhật và sao lưu dữ liệu tối thiểu.
* DBMS còn tương thích với các ngôn ngữ lập trình ứng dụng như Java và Python nhằm mục đích hỗ trợ người dùng kết nối cơ sở dữ liệu với bất kỳ ứng dụng hay một trang web nào. Có chức năng sao lưu tự động và phục hồi khi cần.
* Cho phép người dùng có thể truy cập và chia sẻ dữ liệu.

1. Mô hình dữ liệu (data model) là gì?

Data model (mô hình dữ liệu) là sơ đồ về cách thức tổ chức, lưu trữ dữ liệu trong doanh nghiệp và các mối liên kết giữa các thông tin đó.

Mục đích của data modeling là tạo ra phương pháp hiệu quả nhất để lưu trữ thông tin, đông thời vẫn vẫn cung cấp các quy trình truy cập và báo cáo hoàn chỉnh. Các data model (mô hình dữ liệu) thường mang tính kỹ thuật nhưng (hầu hết) cũng được thiết kế đơn giản và trực quan, giúp những người dùng am hiểu kỹ thuật và cả những người không am hiểu về kỹ thuật dễ dàng nắm bắt các thuật ngữ cơ bản nhất. Nhờ các data model, mọi người trong doanh nghiệp của bạn đều có thể hiểu và làm việc với dữ liệu của bạn một cách hiệu quả hơn. Các mô hình dữ liệu được xây dựng dựa trên nhu cầu của doanh nghiệp. Không có một mô hình dữ liệu cuối cùng, bởi chúng sẽ thay đổi tuỳ theo những biến động trong nhu cầu kinh doanh và quản lý vận hành. Các quy tắc và yêu cầu khi xây dựng mô hình sẽ được thống nhất thông qua phản hồi từ các bên liên quan, sau đó chuẩn hoá thành thước đo để thiết kế mô hình mới hoặc điều chỉnh mô hình sẵn có.

1. Database designer có nhiệm vụ?

Thiết kế cơ sở dữ liệu (database designer hay DBD) có nhiệm vụ xác định những dữ liệu nào cần được lưu trữ, cấu trúc của những dữ liệu ấy, phương pháp thể hiện và lưu trữ các dữ liệu này. Những nhân viên này cũng cần nắm bắt yêu cầu của những đối tượng khác (chủ yếu thuộc nhóm thứ nhất) để thiết kế cơ sở dữ liệu đáp ứng được một cách tốt nhất những yêu cầu ấy, thí dụ tạo ra những khung hiển thị dễ sử dụng, phù hợp, đáp ứng nhanh nhu cầu của người dùng, tạo ra các chương trình xử lý dữ liệu (tính toán, vẽ biểu đồ). Vai trò của những người thiết kế dữ liệu đặc biệt quan trọng trong giai đoạn đầu, khi thiết kế và bắt đầu triển khai. Khi CSDL đã hoạt động ổn định, họ có thể chuyển sang các bộ phận khác.

1. System Analyst có nhiệm vụ?

* Tư vấn với quản lý và người sử dụng để xác định nhu cầu của hệ thống.
* Thiết kế một hệ thống để đáp ứng các mục tiêu và mục đích kinh doanh.
* Xác định đầu vào và đầu ra để đáp ứng nhu cầu của người sử dụng.
* Sử dụng các kỹ thuật như lấy mẫu, xây dựng mô hình và phân tích cấu trúc, cùng với các nguyên tắc tính toán, để đảm bảo giải pháp hiệu quả và khả thi về mặt tài chính.
* Xây dựng sơ đồ và biểu đồ quy trình kỹ thuật cho các lập trình làm theo.
* Giám sát việc thực hiện, phối hợp các bài kiểm tra và quan sát sự khởi đầu của hệ thống để kiểm tra hiệu suất.

1. Application Programmer có nhiệm vụ?

Lập trình viên ứng dụng (application programmer) là một loại người dùng cơ sở dữ liệu (database user), họ là các chuyên gia máy tính viết các chương trình ứng dụng. Họ là các lập trình viên và phát triển các chương trình như C, C++, Java,... Họ cung cấp các giao diện người dùng (user interface) để người dùng khác có thể tương tác với cơ sở dữ liệu.

1. Lược đồ (schema) là gì?

Lược đồ(schema) là tập hợp các đối tượng cơ sở dữ liệu bao gồm table, view, trigger, strored procedure, index,..Một lược đồ được liên kết với username được gọi là chủ sở hữu lược đồ, chủ sở hữu của các đối tượng cơ sở dữ liệu liên quan về mặt logic.

1. Phân loại mô hình dữ liệu

Chủ yếu có ba loại mô hình dữ liệu khác nhau: mô hình dữ liệu khái niệm (conceptual data models), mô hình dữ liệu logic (logical data models) và mô hình dữ liệu vật lý, (physical data models). Mỗi loại sẽ có một mục đích cụ thể khác nhau. Các mô hình dữ liệu được sử dụng để đại diện cho dữ liệu và cách nó được lưu trữ trong cơ sở dữ liệu và để thiết lập mối quan hệ giữa các mục dữ liệu.

1. Mô hình dữ liệu mức khái niệm?

Conceptual data models – Mô hình Dữ liệu Khái niệm: Mô hình dữ liệu này xác định những gì hệ thống chứa. Mô hình này thường được tạo bởi các bên liên quan của doanh nghiệp và Kiến trúc sư dữ liệu. Mục đích là để tổ chức, phạm vi và xác định các khái niệm và quy tắc kinh doanh.

1. Mô hình dữ liệu mức vật lý ?

Physical data models – Mô hình Dữ liệu Vật lý: Mô hình dữ liệu này mô tả cách hệ thống sẽ được triển khai bằng cách sử dụng một hệ thống quản lý cơ sở dữ liệu cụ thể. Mô hình này thường được tạo bởi chuyên viên quản trị dữ liệu và các nhà phát triển. Mục đích là triển khai thực tế cơ sở dữ liệu.

1. Mô hình dữ liệu mức logic ?

Logical data models – Mô hình dữ liệu logic: Xác định cách hệ thống sẽ được triển khai bất kể hệ quản lý cơ sở dữ liệu. Mô hình này thường được tạo bởi kiến trúc sư dữ liệu và nhà phân tích kinh doanh. Mục đích là phát triển bản đồ kỹ thuật của các quy tắc và cấu trúc dữ liệu.

1. Kiến trúc 3 mức trừu tượng của một hệ CSDL? Công dụng của nó?

* Mức vật lý.
* Mức logic
* Mức khung nhìn

Công dụng:

- Với 1 csdl, mỗi người dùng có một khung nhìn riêng. Người này thay đổi khung nhìn không ảnh hưởng đến người khác đang dùng chung csdl này.

- Những tương tác của người dùng với csdl không phụ thuộc vào những vấn đề chi tiết trong lưu trữ dữ liệu.

- Người quản trị csdl ( database administator-DBA) có thể thay đổi cấu trúc lưu trữ của csdl mà không ảnh hưởng đến khung nhìn của người sử dụng.

- Thay đổi về khía cạnh vật lý trong lưu trữ như thay thế thiết bị nhớ thứ cấp có thể không ảnh hưởng đến cấu trúc bên trong csdl.

- Người quản trị csdl có thể thay đổi cấu trúc tổng quát hay cấu trúc khái niệm của csdl mà không làm ảnh hưởng đến tất cả người dùng.

1. Mức vật lý trong kiến trúc 3 mức của 1 hệ CSDL?

Mức vật lý: Một tập hợp các tệp dữ liệu, các chỉ mục hoặc những cấu trúc lưu trữ khác dùng để truy xuất dữ liệu một cách có hiệu quả gọi là CSDL vật lý. CSDL vật lý tồn tại thường xuyên trong thiết bị lưu trữ như đĩa từ, nhiều CSDL có thể được quản lý bởi cùng một hệ quản trị CSDL.

1. Mức logic trong kiến trúc 3 mức trừu tượng của 1 hệ CSDL?

Mức logic: Là mức mô tả những dữ liệu nào được lưu trữ trong CSDL và có những mối quan hệ nào giữa chúng. Mức logic biểu diễn các thực thể (trong thế giới nhỏ), các thuộc tính và các mối quan hệ giữa các thực thể đó; cho thấy các ràng buộc trên dữ liệu, các thông tin về ngữ nghĩa, an ninh và toàn vẹn của dữ liệu. Mức này chỉ quan tâm đến cái gì được lưu trữ trong CSDL chứ không quan tâm đến cách thức để lưu trữ.

1. Mức view trong kiến trúc 3 mức trừu tượng của 1 hệ CSDL?

Mức khung nhìn(view): Là mức mô tả chỉ một phần của toàn bộ CSDL, phần tích hợp với một người sử dụng nhất định. Mỗi người dùng có thể không quan tâm đến toàn bộ thông tin của hệ CSDL mà chỉ một phần thông tin nào đó. Khung nhìn dành cho người sử dụng đó chỉ gồm những thực thể cùng những thuộc tính, những mối quan hệ của những thực thể mà họ quan tâm. Các khung nhìn khác nhau cũng có thể trình bày cùng một dữ liệu nhưng ở những khuôn dạng khác nhau.

Ví dụ: Người sử dụng có thể nhìn thấy thông tin ngày theo kiểu (ngày/tháng/năm), người sử dụng khác thấy thông tin ngày theo kiểu (tháng/ngày/năm).

1. Phân loại Tính độc lập dữ liệu ?

|  |  |
| --- | --- |
| Độc lập dữ liệu logic | Độc lập dữ liệu vật lý |
| Chủ yếu quan tâm đến cấu trúc hoặc thay đổi định nghĩa dữ liệu. | Chủ yếu quan tâm đến việc lưu trữ dữ liệu. |
| Khó truy xuất dữ liệu vì chủ yếu phụ thuộc vào cấu trúc logic của dữ liệu. | Dễ dàng truy xuất. |
| Rất khó để đạt được độc lập dữ liệu logic. | Dễ dàng đạt được sự độc lập dữ liệu vật lý. |
| Cần thực hiện các thay đổi trong lập trình ứng dụng nếu các trường hoặc thuộc tính mới được thêm vào hay xóa khỏi cơ sở dữ liệu. | Sự thay đổi ở mức vật lý thường không cần thay đổi ở mức lập trình ứng dụng. |
| Việc sửa đổi ở các mức logic là điều quan trọng khi cấu trúc logic của cơ sở dữ liệu bị thay đổi. | Các sửa đổi được thực hiện ở các mức vật lý có thể cần hoặc không để cải thiện hiệu suất của cấu trúc. |
| Quan tâm đến lược đồ khái niệm | Quan tâm đến lược đồ vật lý |
| Ví dụ về các thao tác liên quan độc lập dữ liệu logic: Thêm, Sửa đổi, Xóa một thuộc tính mới | Ví dụ về các thao tác liên quan độc lập dữ liệu vật lý: Thay đổi về phương pháp nén, phương pháp băm, thiết bị lưu trữ. |

1. Tính độc lập vật lý?

Tính độc lập dữ liệu vật lý giúp tách các mức khái niệm khỏi các mức nội bộ hay vật lý. Nó cung cấp mô tả logic về cơ sở dữ liệu mà không cần xác định cấu trúc vật lý. So với tính độc lập logic, thì độc lập dữ liệu vật lý dễ dàng để có thể có được.

Với sự độc lập về vật lý, bạn có thể dễ dàng thay đổi cấu trúc hoặc các thiết bị lưu trữ vật lý có ảnh hưởng đến lược đồ khái niệm. Bất kỳ thay đổi nào được thực hiện sẽ được tiếp nhận bởi việc ánh xạ giữa mức khái niệm và mức vật lý. Sự độc lập dữ liệu mức vật lý đạt được nhờ sự tồn tại của mức vật lý của cơ sở dữ liệu và sau đó là sự chuyển đổi từ mức khái niệm của cơ sở dữ liệu sang mức vật lý.

Ví dụ về các thay đổi trong độc lập dữ liệu vật lý:

Do tính độc lập về vật lý, bất kỳ thay đổi nào dưới đây cũng sẽ không ảnh hưởng đến lớp khái niệm:

- Sử dụng thiết bị lưu trữ mới như ổ cứng hoặc băng từ.

- Sửa đổi các phương thức tổ chức tệp trong cơ sở dữ liệu.

- Chuyển sang các cấu trúc dữ liệu khác nhau.

- Thay đổi phương thức truy cập.

- Sửa đổi các chỉ mục.

- Các thay đổi sang các phương pháp nén hoặc phương pháp băm.

- Thay đổi vị trí của cơ sở dữ liệu từ ổ C sang ổ D.

1. Tính độc lập logic là gì?

Độc lập dữ liệu logic là khả năng thay đổi sơ đồ mức khái niệm mà không làm ảnh hưởng tới các mức khác.

* Các khung nhìn bên ngoài.
* API hoặc chương trình bên ngoài.

Bất kỳ thay đổi nào được thực hiện sẽ được tiếp nhận bởi ánh xạ giữa mức bên ngoài và mức khái niệm. So với tính độc lập dữ liệu vật lý, việc đạt được sự độc lập về dữ liệu logic là sẽ khó hơn.

Ví dụ về các thay đổi trong độc lập dữ liệu logic:

Do tính độc lập về logic, bất kỳ thay đổi nào dưới đây sẽ không ảnh hưởng đến mức bên ngoài.

* Có thể thêm, sửa đổi hoặc xóa một thuộc tính, thực thể hoặc mối quan hệ mới mà không cần viết lại các chương trình ứng dụng hiện có.
* Hợp nhất hai bản ghi thành một.
* Tách các bản ghi hiện có thành hai hoặc nhiều bản ghi khác nhau.