

Homework2

Phạm Ngọc Hải

March 27, 2024

1 Mô hình thực thể quan hệ (ER model)

1.1 Minh họa mô hình ER cho 1 bài toán

Dưới đây là mô hình Entity-Relationship (E-R) cho một cửa hàng sách trực tuyến với nhiều kho sách bán sách.

Các ý nghĩa thông thường cho các tác giả đã viết sách, khách hàng mua sách và giỏ hàng chứa đơn hàng của khách hàng.

Lưu ý rằng một giỏ hàng có thể chứa nhiều sách nhưng được liên kết với một khách hàng duy nhất. Bao gồm chỉ các thuộc tính cần thiết để hỗ trợ giao dịch mua bán (giá của một cuốn sách và số lượng sách đã mua, và kho sách cần giao từ) và các khóa phù hợp cho các thực thể và mối quan hệ. Minh họa các khóa.

Trong mô hình này, chúng ta sẽ có các thực thể sau:

1. **Tác giả (Authors):**
 - **author_id** (khóa chính): Mã định danh cho mỗi tác giả.
 - **author_name**: Tên của tác giả.
2. **Sách (Books):**
 - **book_id** (khóa chính): Mã định danh cho mỗi cuốn sách.
 - **title**: Tiêu đề của cuốn sách.
 - **author_id** (khóa ngoại): Mã định danh của tác giả của cuốn sách.
3. **Khách hàng (Customers):**
 - **customer_id** (khóa chính): Mã định danh cho mỗi khách hàng.
 - **customer_name**: Tên của khách hàng.
4. **Giỏ hàng (Shopping Carts):**
 - **cart_id** (khóa chính): Mã định danh cho mỗi giỏ hàng.
 - **customer_id** (khóa ngoại): Mã định danh của khách hàng đặt hàng.
5. **Sách trong giỏ hàng (Books in Shopping Carts):**
 - **cart_id** (khóa chính, khóa phụ): Mã giỏ hàng.
 - **book_id** (khóa chính, khóa phụ): Mã sách.
 - **price**: Giá của cuốn sách.
 - **quantity**: Số lượng cuốn sách trong giỏ hàng.
6. **Kho sách (Warehouses):**
 - **warehouse_id** (khóa chính): Mã định danh cho mỗi kho sách.
 - **warehouse_name**: Tên của kho sách.
7. **Kho sách chứa sách (Books in Warehouses):**
 - **book_id** (khóa chính, khóa phụ): Mã sách.
 - **warehouse_id** (khóa chính, khóa phụ): Mã kho sách.

Các mối quan hệ giữa các thực thể được minh họa bằng các mũi tên với các hình chữ nhật ở đầu mỗi mũi tên, đại diện cho các khóa ngoại. Các thuộc tính khóa chính được khoanh tròn và ghi đậm.

1.2 Chuyển đổi ER model thành relational model và xác định khóa

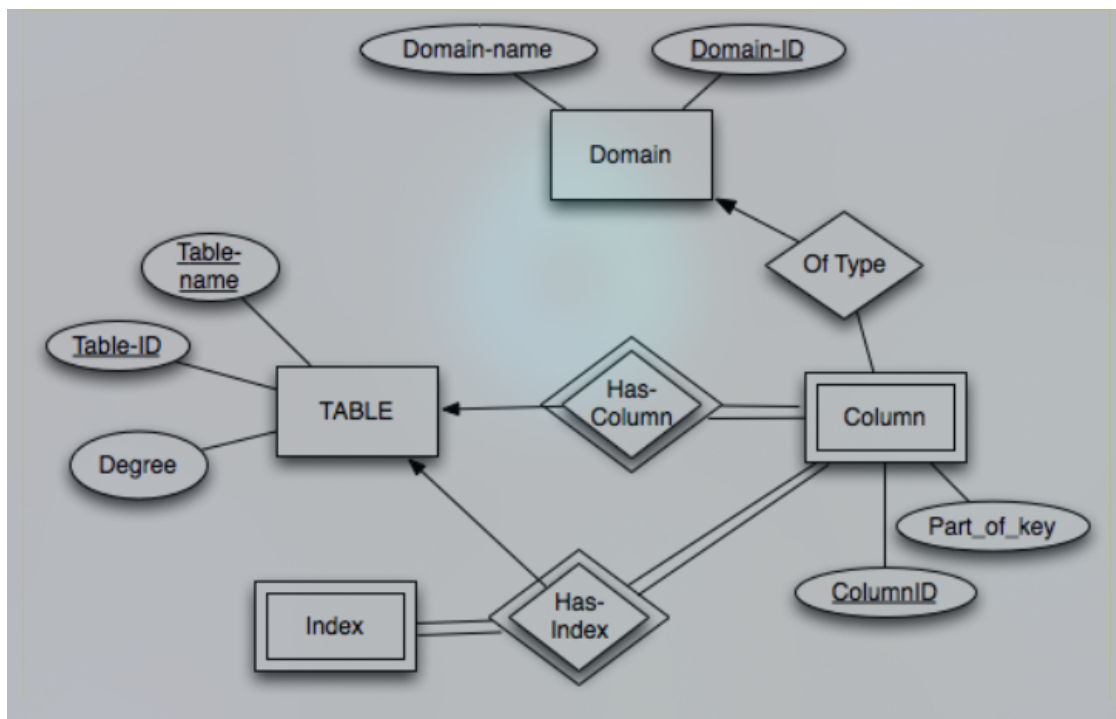
Dựa trên mô hình E-R mà chúng ta đã tạo, chúng ta có thể ánh xạ nó thành một schema quan hệ. Dưới đây là cách ánh xạ:

1. **Tác giả (Authors):**
 - **author** (**author_id**: primary key, **author_name**)
2. **Sách (Books):**
 - **book** (**book_id**: primary key, **title**, **author_id**: foreign key references author(author_id))
3. **Khách hàng (Customers):**
 - **customer** (**customer_id**: primary key, **customer_name**)
4. **Giỏ hàng (Shopping Carts):**
 - **cart** (**cart_id**: primary key, **customer_id**: foreign key references customer(customer_id))
5. **Sách trong giỏ hàng (Books in Shopping Carts):**
 - **cart_book** (**cart_id**: foreign key references cart(cart_id), **book_id**: foreign key references book(book_id), **price**, **quantity**)
6. **Kho sách (Warehouses):**
 - **warehouse** (**warehouse_id**: primary key, **warehouse_name**)
7. **Kho sách chứa sách (Books in Warehouses):**
 - **warehouse_book** (**warehouse_id**: foreign key references warehouse(warehouse_id), **book_id**: foreign key references book(book_id))

Trong schema này:

- Mỗi thực thể trong mô hình E-R được ánh xạ thành một bảng quan hệ.
- Mỗi thuộc tính trong thực thể trở thành một cột trong bảng.
- Các khóa chính của mỗi thực thể trở thành các khóa chính trong bảng tương ứng.
- Các khóa ngoại của mỗi thực thể được ánh xạ thành các cột trong bảng có tham chiếu đến các khóa chính của các bảng khác.

1.3 Mô tả mô hình thực thể ER sau



Mô hình ER trong ảnh thể hiện cấu trúc cơ sở dữ liệu với các bảng, cột và chỉ mục. Các bảng được kết nối với nhau thông qua các cột và chỉ mục, và các cột được kết nối với nhau thông qua chỉ mục.

Thành phần chính:

- **Bảng:**
 - **Domain:** Lưu trữ thông tin về tên miền (Domain-name) và mã định danh (Domain-ID).
 - **Table:** Lưu trữ thông tin về các bảng trong cơ sở dữ liệu, bao gồm tên bảng (Table-name), mã định danh (Table-ID) và kiểu dữ liệu (Of Type).
 - **Column:** Lưu trữ thông tin về các cột trong bảng, bao gồm tên cột (Column-name) và mã định danh (ColumnID).
 - **Index:** Lưu trữ thông tin về các chỉ mục trong cơ sở dữ liệu, bao gồm tên chỉ mục (Index-name) và mã định danh (IndexID).
- **Mối quan hệ:**
 - **Domain** có mối quan hệ 1-n với **Table**. Mỗi tên miền có thể được sử dụng bởi nhiều bảng.
 - **Table** có mối quan hệ 1-n với **Column**. Mỗi bảng có thể chứa nhiều cột.
 - **Table** có mối quan hệ 1-n với **Index**. Mỗi bảng có thể có nhiều chỉ mục.
 - **Column** có mối quan hệ n-n với **Index**. Một cột có thể tham gia vào nhiều chỉ mục, và một chỉ mục có thể bao gồm nhiều cột.

Thuộc tính:

- **Domain:**
 - **Domain-name:** Tên miền (ví dụ: `com`, `org`).
 - **Domain-ID:** Mã định danh duy nhất cho tên miền.

- **Table:**
 - **Table-name:** Tên bảng (ví dụ: `users`, `products`).
 - **Table-ID:** Mã định danh duy nhất cho bảng.
 - **Of Type:** Kiểu dữ liệu của bảng (ví dụ: `int`, `varchar`).
- **Column:**
 - **Column-name:** Tên cột (ví dụ: `username`, `email`).
 - **ColumnID:** Mã định danh duy nhất cho cột.
- **Index:**
 - **Index-name:** Tên chỉ mục (ví dụ: `primary_key`, `unique_index`).
 - **IndexID:** Mã định danh duy nhất cho chỉ mục.

Chú thích:

- Mô hình ER này không đầy đủ và có thể được mở rộng để bao gồm các thành phần khác như khóa chính, khóa ngoại, v.v.
- Hình ảnh chỉ hiển thị một phần của mô hình ER. Có thể có thêm các bảng và mối quan hệ không được hiển thị.

=> Tóm tắt:

Mô hình ER trong ảnh mô tả cấu trúc cơ sở dữ liệu với các bảng, cột và chỉ mục. Các thành phần được kết nối với nhau thông qua các mối quan hệ 1-n và n-n. Mô hình này có thể được sử dụng để thiết kế và quản lý cơ sở dữ liệu.

2 Phụ thuộc hàm (Functional Dependencies)

Consider the following relation and the functional dependencies on it:

* $R(A,B,C,D,E)$:

* $A \rightarrow BC$

* $CD \rightarrow E$

* $B \rightarrow D$

* $E \rightarrow A$

a) Find all candidate keys.

Then argue why there cannot be a candidate key with three attributes.

b) Is R in BCNF? Explain.

c) Is R in 3NF?

d)

- Is the decomposition $R_1(A,B,C)$ and $R_2(A,D,E)$ of R lossless or lossy? Justify your answer.
- Is this decomposition FD preserving? If your answer is NO, what is not preserved?
- Is the decomposition $R_3(A,B,C,D)$ and $R_4(C,D,E)$ of R lossless or lossy? Justify your answer.
- Is this decomposition FD preserving? If your answer is NO, what is not preserved?

e)

- What is R1? Circle your answer BCNF 3NF NONE OF THESE
 - What is R2? Circle your answer BCNF 3NF NONE OF THESE
 - What is R3? Circle your answer BCNF 3NF NONE OF THESE
 - What is R4? Circle your answer BCNF 3NF NONE OF THESE
- f) Using the decomposition algorithm lossless-ly decompose R into BCNF relations. Then check if it is FD preserving.

Bài làm

a) Khóa ứng viên:

1. **A** (thỏa mãn $A \rightarrow BC$)
2. **CD** (thỏa mãn $CD \rightarrow E$)

Tại sao không có khóa ứng viên với ba thuộc tính?

- Khóa ứng viên dùng để định dạng duy nhất một bản ghi trong một quan hệ.
- Nếu có khóa ứng viên với ba thuộc tính (ví dụ: BCD), nó cần xác định tất cả các thuộc tính khác (E và A trong trường hợp này).
- Tuy nhiên, các phụ thuộc hàm cho chúng ta biết:
 - A xác định BC (không xác định E)
 - CD xác định E (không xác định A)
- Không có tập hợp ba thuộc tính nào có thể đáp ứng yêu cầu này, do đó không có khóa ứng viên với ba thuộc tính.

b) R có ở dạng chuẩn BCNF không? Không.

- BCNF (dạng chuẩn Boyce-Codd) yêu cầu mọi định danh (bên trái của một phụ thuộc hàm) phải là một khóa ứng viên.
- Trong quan hệ này:
 - B xác định D, nhưng B không phải là khóa ứng viên.

c) R có ở dạng chuẩn 3NF không? Có.

- 3NF yêu cầu:
 - Không có thuộc tính phụ thuộc (các thuộc tính không nằm trong bất kỳ khóa ứng viên nào) phụ thuộc vào một tập hợp con hợp lý của một khóa ứng viên.
- Trong trường hợp này:
 - E phụ thuộc vào CD, là một khóa ứng viên.
 - D phụ thuộc vào B, không phải là khóa ứng viên, nhưng $B \rightarrow D$ là một phụ thuộc hàm đầy đủ (B xác định duy nhất D).
 - Do đó, tất cả các phụ thuộc hàm đều thỏa mãn yêu cầu của 3NF.

d) Phân rã quan hệ:

R1(A,B,C) và R2(A,D,E):

- **Mất mát dữ liệu:** Không. Chúng ta có thể xây dựng lại quan hệ R ban đầu bằng cách nối R1 và R2 trên thuộc tính A.

- **Bảo toàn phụ thuộc hàm:** Không. Phụ thuộc hàm $B \rightarrow D$ không được bảo toàn vì B không phải là khóa ứng viên trong R1 hoặc R2.

R3(A,B,C,D) và R4(C,D,E):

- **Mất mát dữ liệu:** Không. Chúng ta có thể xây dựng lại R bằng cách nối R3 và R4 trên (C, D).
- **Bảo toàn phụ thuộc hàm:** Không. Phụ thuộc hàm $A \rightarrow BC$ không được bảo toàn vì A không phải là khóa ứng viên trong R3 hoặc R4.

e) Trạng thái chuẩn hóa của các phân rã:

- R1: Không BCNF, 3NF (**3NF**)
- R2: Không BCNF, 3NF (**3NF**)
- R3: Không BCNF, 3NF (**3NF**)
- R4: Không BCNF, 3NF (**3NF**)

f) Phân rã R thành các quan hệ BCNF không mất mát dữ liệu:

Chúng ta có thể phân rã R thành các quan hệ BCNF theo các bước sau:

1. Phân rã R dựa trên $A \rightarrow BC$:
 - R1(A, BC)
2. Phân rã R1(A, BC) thêm dựa trên $B \rightarrow D$:
 - R2(A, B)
 - R3(B, C)

Phân rã này (R2, R3) thỏa mãn BCNF vì tất cả các định danh đều là khóa ứng viên trong các quan hệ tương ứng.

Tuy nhiên, phân rã này **không bảo toàn phụ thuộc hàm** vì phụ thuộc hàm $B \rightarrow D$ bị mất. B không phải là khóa ứng viên trong R2 hoặc R3, do đó chúng ta không thể suy ra D từ B nữa.

Tóm tắt:

- Quan hệ R ban đầu có các khóa ứng viên là A và CD.
- Nó không ở dạng chuẩn BCNF do $B \rightarrow D$ không có B là khóa ứng viên.
- Nó ở dạng chuẩn 3NF, nhưng việc phân rã thành R1/R2 hoặc R3/R4 làm mất thông tin về phụ thuộc hàm.
- Một phân rã BCNF là R2(A, B) và R3(B, C), nhưng nó không bảo toàn phụ thuộc hàm.