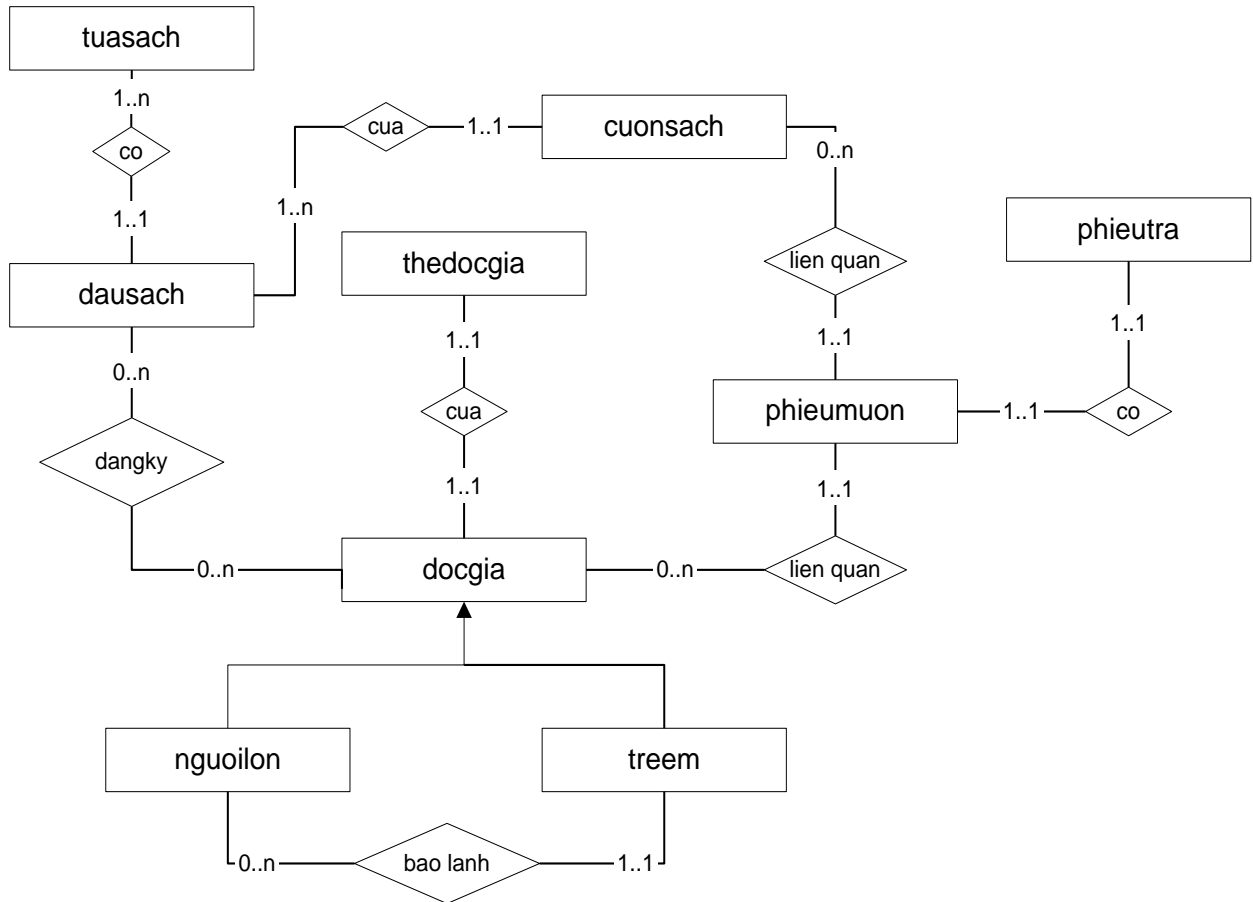


## CHUYỂN ĐỔI TỪ MÔ HÌNH ER SANG MÔ HÌNH DỮ LIỆU QUAN HỆ

### I. Mô hình dữ liệu ER



### II. Sự chuyển đổi từ mô hình dữ liệu ER sang mô hình dữ liệu quan hệ

Dựa theo 6 bước trong phương pháp chuyển đổi từ một mô hình dữ liệu ER sang mô hình quan hệ dữ liệu, chúng ta có mô hình quan hệ dữ liệu như sau:

**docgia**(ma\_docgia, ho, tenlot, ten, hình).

**thedocgia**(ma\_docgia, ngaylapthe, ngayhethan)

**nguoilon**(ma\_docgia, sonha, duong, quan, dienthoai, ngaysinh)

**treem**(ma\_docgia, ma\_docgia\_nguoilon, ngaysinh)

**tuasach**(ma\_tuasach, tuasach, tacgia, tomtat)

**dausach**(isbn, ma\_tuasach, ngonngu, bia, trangthai)

**cuonsach**(ma\_cuonsach, isbn, tinhtrang)

**dangky**(isbn, ma\_docgia, ngay\_dk, ghichu)

**phieumuon**(isbn, ma\_cuonsach, ma\_docgia, ngaymuon, ngaytra)

**phieutra**(isbn, ma\_cuonsach, ma\_docgia, ngaymuon, ngaytrathats, tienphat)

### III. Nhận xét và đề nghị một số mô hình quan hệ dữ liệu khác

- **Nhận xét 1 :** Vì theo yêu cầu của đề bài mã thể độc giả chính là mã độc giả, do vậy khóa chính của quan hệ **thedocgia** chính là **ma\_docgia**. Nhưng câu hỏi đặt ra ở đây là chúng ta có nên có quan hệ **thedocgia** này hay không? Nếu có thì mục đích để làm gì? Chúng ta nhận thấy thông tin cần thiết cho một độc giả đã thể hiện đầy đủ trên quan hệ **docgia(nguoiilon,treem)**, như vậy chức năng chính của quan hệ **thedocgia** là chỉ cung cấp thông tin *ngaylaple* và *ngayhethan*, mà hai thuộc tính này được truy xuất thường xuyên để kiểm tra tình trạng thẻ độc giả có còn hợp lệ không khi độc giả mượn sách. Vậy rõ ràng khi có yêu cầu hệ thống kết xuất thông tin độc giả cùng với thông tin thẻ độc giả thì chúng ta phải thực hiện một phép kết giữa hai quan hệ **docgia** và **thedocgia** mới lấy được đầy đủ thông tin, nhưng giả sử hệ thống chúng ta rất quan tâm đến tốc độ của việc truy xuất thông tin, thì việc thực hiện phép kết này là không nên chút nào. Do vậy chúng ta cần nên gộp quan hệ **thedocgia** vào trong quan hệ **docgia**, vì khi chúng ta để lại quan hệ **thedocgia** thì cũng không có ích lợi gì hơn cho hệ thống. Do đó quan hệ **docgia** và **thedocgia** được gộp chung thành quan hệ **docgia** như sau:

**docgia**(*ma\_docgia*, *ho*, *tenlot*, *ten*, *hinh*, *ngaylapthe*, *ngayhethan*).

- **Nhận xét 2:** Tiếp theo chúng ta thấy việc tồn tại của quan hệ **phieumuon** và **phieutra** là có nên hay không? Rõ ràng là chúng ta thấy không nên vì phải chấp nhận một sự trùng lặp dữ liệu quá lớn, muốn truy xuất thông tin thì phải kết giữa hai bảng (công việc được khuyến cáo là cần hạn chế càng tốt vì phép kết làm chậm tốc độ truy xuất thông tin), đồng thời với việc lưu trữ dư thừa như vậy sẽ làm cho kích thước lưu trữ trong hệ thống bị phình to.

Dữ liệu minh họa cho hai bảng **phieumuon** và **phieutra** (dữ liệu trùng lặp quá nhiều, lưu trữ thừa làm cho kích thước hệ thống bị phình to)

#### 1. Phieumuon

isbn	ma_cuonsach	ma_docgia	ngaymuon
1	1	1	01/01/2004
1	1	1	10/2/2004
1	1	1	10/6/2004
1	1	1	10/8/2004
...	...	...	...
...	...	...	...

#### 2. Phieutra

isbn	ma_cuonsach	ma_docgia	ngaytra	ngaytrathatsu	tienphat
1	1	1	15/01/2004	10/01/2004	0
1	1	1	24/02/2004	10/03/2004	50.000
1	1	1	24/06/2004	20/06/2004	0
1	1	1	24/08/2004	20/8/2004	0
...	...	...	...	...	...
...	...	...	...	...	...

Do vậy ý tưởng đầu tiên là chúng ta nên gộp hai quan hệ **phieumuon** và **phieutra** thành quan hệ **muontra** như sau:

**muontra**(*isbn*, *ma\_cuonsach*, *ma\_docgia*, *ngaymuon*, *ngaytra*, *ngaytrathatsu*, *tienphat*)

Dữ liệu minh họa cho quan hệ **muontra**

isbn	ma_cuonsach	ma_docgia	ngaymuon	ngaytra	Ngaytrathats	tienphat
1	1	1	01/01/2004	15/01/2004	10/01/2004	0
1	1	1	10/2/2004	24/02/2004	10/03/2004	50.000
1	1	1	10/6/2004	24/06/2004	20/06/2004	0
1	1	1	10/8/2004	24/08/2004	20/8/2004	0
...	...	...	...	...	...	...
...	...	...	...	...	...	...

- **Nhận xét 3:** với quan hệ **muontra** mới chúng ta nhận thấy rằng vẫn còn sự trùng lặp dữ liệu nhưng chúng ta dễ dàng truy xuất nhanh các thông tin mượn trả sách cho thư viện. Với quan hệ này rất phù hợp cho một hệ thống yêu cầu về mặt tốc độ truy xuất thông tin nhanh kể cả chấp nhận sự trùng lặp dữ liệu. Nhưng nếu hệ thống yêu cầu không được trùng lặp thông tin thì sao? Lúc này chắc chắn chúng ta không nên để nguyên quan hệ **muontra** mà phải nên phân thành hai quan hệ **muon** và **qtrinhmuon** như sau:

**muon**(*isbn, ma\_cuonsach, ma\_docgia*)

**qtrinhmuon**(*isbn, ma\_cuonsach, ngaymuon, ma\_docgia, ngay tra, ngaytrathats, tienphat*)

Dữ liệu minh họa cho hai quan hệ **muon** và **qtrinhmuon**

1. **muon**

isbn	ma_cuonsach	ma_docgia
1	1	1

2. **qtrinhmuon**

isbn	ma_cuonsach	ngaymuon	ma_docgia	ngaytra	ngaytrathats	tienphat
1	1	01/01/2004	1	15/01/2004	10/01/2004	0
1	1	10/2/2004	1	24/02/2004	10/03/2004	50.000
1	1	10/6/2004	1	24/06/2004	20/06/2004	0
1	1	10/8/2004	1	24/08/2004	20/8/2004	0
...	...	...	...	...	...	...
...	...	...	...	...	...	...

- **Nhận xét 4:** với dữ liệu minh họa trên ta thấy có giảm bớt sự trùng lặp thông tin và kích thước cho hệ thống. Nhưng nếu chúng ta muốn hệ thống lưu trữ tốt nhất, không bị trùng lặp thông tin, cũng như kích thước hệ thống càng nhỏ càng tốt thì sao? Lúc đó ta vẫn chọn phương án trên, nhưng để tránh việc lưu trữ nhiều và phép kết xảy ra giữa hai bảng thực hiện nhanh hơn ta chọn lại cách đặt khóa chính trên bảng **muon**, nghĩa là ta không chọn 3 thuộc tính *isbn, ma\_cuonsach, ma\_docgia* làm khóa chính mà chúng ta thêm vào một thuộc tính *ma\_phieumuon* để làm khóa chính. Như vậy lúc đó hai quan hệ **muon**, và **qtrinhmuon** có sự thay đổi như sau:

**muon**(*ma\_phieumuon, isbn, ma\_cuonsach, ma\_docgia*)

**qtrinhmuon**(*ma\_phieumuon, ngaymuon, ngay tra, ngaytrathats, tienphat*)

Dữ liệu minh họa cho hai bảng

1. **muon**

<b>ma_phieumuon</b>	<b>isbn</b>	<b>ma_cuonsach</b>	<b>ma_docgia</b>
<b>PM01</b>	<b>1</b>	<b>1</b>	<b>1</b>

2. **qtrinhmuon**

<b>Ma_phieumuon</b>	<b>ngaymuon</b>	<b>ngaytra</b>	<b>Ngaytrathatsu</b>	<b>tienphat</b>
<b>PM01</b>	<b>01/01/2004</b>	<b>15/01/2004</b>	<b>10/01/2004</b>	<b>0</b>
<b>PM01</b>	<b>10/2/2004</b>	<b>24/02/2004</b>	<b>10/03/2004</b>	<b>50.000</b>
<b>PM01</b>	<b>10/6/2004</b>	<b>24/06/2004</b>	<b>20/06/2004</b>	<b>0</b>
<b>PM01</b>	<b>10/8/2004</b>	<b>24/08/2004</b>	<b>20/8/2004</b>	<b>0</b>
...	...	...	...	...
...	...	...	...	...

#### IV. Kết luận

Như vậy dựa trên sự phân tích và nhận xét ở trên chúng ta có thể có những mô hình quan hệ có thể sử dụng:

- a. **Mô hình 1:** trường hợp xấu nhất(trùng lặp, kết xuất chậm, kích thước hệ thống lớn) ☹

**docgia**(ma\_docgia, ho, tenlot, ten, hình).  
**thedocgia**( ma\_docgia, ngaylapthe, ngayhethan)  
**nguoiilon**(ma\_docgia, sonha, duong, quan, dienthoai, ngaysinh)  
**treem**(ma\_docgia, ma\_docgia\_nguoiilon, ngaysinh)  
**tuasach**(ma\_tuasach, tuasach, tacgia, tomtat)  
**dausach**(isbn, ma\_tuasach, ngonngu, bia, trangthai)  
**cuonsach**(ma\_cuonsach, isbn, tinhtrang)  
**dangky**(isbn, ma\_docgia, ngay\_dk, ghichu)  
**phieumuon**(isbn, ma\_cuonsach, ma\_docgia, ngaymuon, ngaytra)  
**phieutra**(isbn, ma\_cuonsach, ma\_docgia, ngaymuon, ngaytrathatsu, tienphat)

- b. **Mô hình 2:** (hệ thống truy xuất nhanh, chấp nhận sự trùng lặp thông tin)

**docgia**(ma\_docgia, ho, tenlot, ten, hình, ngaylapthe, ngayhethan)  
**nguoiilon**(ma\_docgia, sonha, duong, quan, dienthoai, ngaysinh)  
**treem**(ma\_docgia, ma\_docgia\_nguoiilon, ngaysinh)  
**tuasach**(ma\_tuasach, tuasach, tacgia, tomtat)  
**dausach**(isbn, ma\_tuasach, ngonngu, bia, trangthai)  
**cuonsach**(ma\_cuonsach, isbn, tinhtrang)  
**dangky**(isbn, ma\_docgia, ngay\_dk, ghichu)  
**muontra**( isbn, ma\_cuonsach, ma\_docgia, ngaymuon, ngaytra, ngaytrathatsu, tienphat)

- c. **Mô hình 3:** hệ thống vẫn còn sự trùng lặp thông tin, khi thực hiện phép kết vẫn còn phải kết trên cả ba thuộc tính khóa chính của quan hệ muon

```
docgia(ma_docgia, ho, tenlot, ten, hình, ngaylapthe, ngayhethan)
nguoiilon(ma_docgia, sonha, duong, quan, dienthoai, ngaysinh)
treem(ma_docgia, ma_docgia_nguoiilon, ngaysinh)
tuasach(ma_tuasach, tuasach, tacgia, tomtat)
dausach(isbn, ma_tuasach, ngonngu, bìa, trangthai)
cuonsach(ma_cuonsach, isbn, tinhtrang)
dangky(isbn, ma_docgia, ngay_dk, ghichu)
muon(isbn, ma_cuonsach, ma_docgia)
qtrinhmuon(isbn, ma_cuonsach, ngaymuon, ma_docgia, ngay tra, ngaytrathatsutienphat)
```

- d. **Mô hình 4:** hệ thống không còn sự trùng lặp thông tin

```
docgia(ma_docgia, ho, tenlot, ten, hình, ngaylapthe, ngayhethan)
nguoiilon(ma_docgia, sonha, duong, quan, dienthoai, ngaysinh)
treem(ma_docgia, ma_docgia_nguoiilon, ngaysinh)
tuasach(ma_tuasach, tuasach, tacgia, tomtat)
dausach(isbn, ma_tuasach, ngonngu, bìa, trangthai)
cuonsach(ma_cuonsach, isbn, tinhtrang)
dangky(isbn, ma_docgia, ngay_dk, ghichu)
muon(ma_phieumuon, isbn, ma_cuonsach, ma_docgia)
qtrinhmuon(ma_phieumuon, ngaymuon, ngay tra, ngaytrathatsutienphat)
```

- Như vậy: tùy theo yêu cầu của hệ thống mà chúng ta lựa chọn một mô hình dữ liệu quan hệ thích hợp để cài đặt.
- ❖ Bài tập sinh viên:
- ✓ Phân tích và tìm ra nhiều mô hình quan hệ khác (có đánh giá ưu và khuyết điểm của chúng)
  - ✓ Tìm và mô tả tất cả các ràng buộc toàn vẹn cho từng mô hình.