



CƠ SỞ DỮ LIỆU

CHƯƠNG 5: RÀNG BUỘC TOÀN VỆN

ThS. LÊ NGÔ THỰC VI



MỤC TIÊU



1. **Nắm rõ khái niệm và mô tả ràng buộc toàn vẹn.**
2. **Xác định được các loại ràng buộc toàn vẹn**
3. **Nhận diện được các ràng buộc toàn vẹn trong một mô tả CSDL**



NỘI DUNG

1. Khái niệm
2. Các đặc trưng của Ràng buộc toàn vẹn (RBTV)
3. Phân loại
4. Cài đặt



KHÁI NIỆM

1



1. Khái niệm

- **RBTV (Integrety Constraints)** xuất phát từ những qui định hay điều kiện
 - Trong thực tế
 - Trong mô hình dữ liệu
 - Các thao tác làm thay đổi dữ liệu không nên được thực hiện một cách tùy tiện vì có thể đưa CSDL đến một tình trạng ‘xấu’
- RBTV là một điều kiện được định nghĩa trên một hay nhiều quan hệ khác nhau
- Các RBTV là những điều kiện **bất biến** mà mọi thể hiện của quan hệ đều phải thỏa ở bất kỳ thời điểm nào



1. Khái niệm

- Tại sao cần phải có RBTV?
 - Bảo đảm **tính kết dính** của các thành phần cấu tạo nên CSDL
 - Bảo đảm **tính nhất quán** của dữ liệu
 - Bảo đảm CSDL luôn biểu diễn **đúng ngữ nghĩa** thực tế
- Ví dụ
 - Mức lương của một người nhân viên không được vượt quá trưởng phòng (R1)
 - Người quản lý trực tiếp (của một nhân viên) phải là một nhân viên trong công ty (R2)



CÁC ĐẶC TRƯNG CỦA RBTV

Bối cảnh – Nội dung – Bảng tầm ảnh hưởng

2



2.1. Bối cảnh

- Là những **quan hệ có khả năng bị vi phạm RBTV** khi thực hiện các phép cập nhật (thêm, xóa, sửa)
- Ví dụ (R1): Mức lương của một người nhân viên không được vượt quá trưởng phòng
 - Các phép cập nhật
 - Thêm mới một nhân viên vào một phòng ban
 - Bổ nhiệm trưởng phòng cho một phòng ban
 - Cập nhật lương cho nhân viên
 - Bối cảnh: NHANVIEN, PHONGBAN



2.1. Bối cảnh (tt)

- Là những **quan hệ có khả năng bị vi phạm RBTV** khi thực hiện các phép cập nhật (thêm, xóa, sửa)
- Ví dụ (R2): Người quản lý trực tiếp phải là một nhân viên trong công ty
 - Các phép cập nhật
 - Cập nhật người quản lý trực tiếp của một nhân viên
 - Thêm mới một nhân viên
 - Bối cảnh: NHANVIEN



2.2. Nội dung

- Nội dung của một RBTV được phát biểu bằng
 - **Ngôn ngữ tự nhiên**
 - Dễ hiểu nhưng thiếu tính chặt chẽ
 - **Ngôn ngữ hình thức**
 - Cô đọng, chặt chẽ nhưng đôi lúc khó hiểu
 - Biểu diễn thông qua mã giả
 - Phép tính quan hệ

2.2. Nội dung (tt)

- Ví dụ (R1):
 - **Ngôn ngữ tự nhiên:** Mức lương của một người nhân viên không được vượt quá trưởng phòng
 - **Ngôn ngữ hình thức**
$$\forall t \in \text{NHANVIEN} (\exists u \in \text{PHONGBAN} (\exists v \in \text{NHANVIEN} ($$
$$u.\text{TRPHG} = v.\text{MANV} \wedge$$
$$u.\text{MAPHG} = t.\text{PHG} \wedge$$
$$t.\text{LUONG} \leq v.\text{LUONG})))$$



2.2. Nội dung (tt)

- Ví dụ (R2):
 - **Ngôn ngữ tự nhiên:** Người quản lý trực tiếp phải là một nhân viên trong công ty
 - **Ngôn ngữ hình thức**

$\forall t \in \text{NHANVIEN} (t.\text{MA_NQL} \neq \text{null} \wedge \exists s \in \text{NHANVIEN} (t.\text{MA_NQL} = s.\text{MANV}))$



2.3. Bảng tầm ảnh hưởng

- Xác định thao tác nào (thêm, xóa, sửa) trên dữ liệu cần phải kiểm tra RBTV khi được thực hiện trên quan hệ bối cảnh
- Có 2 loại:
 - Bảng tầm ảnh hưởng cho một RBTV
 - Bảng tầm ảnh hưởng tổng hợp

2.3. Bảng tầm ảnh hưởng (tt)

- Bảng tầm ảnh hưởng cho một RBTV

Tên_RB	Thêm	Xóa	Sửa
Quan hệ 1	+	–	+ (Thuộc tính)
Quan hệ 2	–	+	–
...			
Quan hệ n	–	+	–

(+) Vi phạm RBTV

(–) Không vi phạm RBTV

2.3. Bảng tầm ảnh hưởng (tt)

- Bảng tầm ảnh hưởng tổng hợp

	Ràng buộc 1			Ràng buộc 2						Ràng buộc m		
	T	X	S	T	X	S	T	X	S
Quan hệ 1	+	-	+	+	-	+				+	-	+
Quan hệ 2	-	+	-									
Quan hệ 3	-	-	+							-	+	-
...												
Quan hệ n				-	+	-				-	-	+



PHÂN LOẠI RBTV

3



3. Phân loại RBTV

- Một quan hệ
 - Miền giá trị
 - Liên bộ
 - Liên thuộc tính
- Nhiều quan hệ
 - Tham chiếu
 - Liên bộ, liên quan hệ
 - Liên thuộc tính, liên quan hệ
 - Thuộc tính tổng hợp
 - Chu trình

3.1. Miền giá trị

- Ràng buộc qui định các giá trị cho một thuộc tính

R	A	B	C	D
	α	α	1	1
	α	β	5	7
	β	β	12	3
	β	β	23	9

$$\beta \in \{\alpha, \beta, \gamma\}$$

$$9 \in \{1..10\}$$

- Miền giá trị: liên tục hoặc rời rạc



3.1. Miền giá trị (tt)

- Ví dụ (R3): Thời gian tham gia mỗi đề án của một nhân viên không quá 60 giờ
 - Bối cảnh: PHANCONG
 - Biểu diễn: $\forall t \in \text{PHANCONG} (t.\text{THOIGIAN} \leq 60)$
 - Bảng tầm ảnh hưởng:

R3	Thêm	Xóa	Sửa
PHANCONG	+	-	+ (THOIGIAN)



3.1. Miền giá trị (tt)

- **Ví dụ (R4): Giới tính của nhân viên là ‘Nam’ hoặc ‘Nu’**
 - Bối cảnh: NHANVIEN
 - Biểu diễn: $\forall t \in \text{NHANVIEN} (t.\text{PHAI} \in \{\text{‘Nam’}, \text{‘Nu’}\})$
hay $\text{DOM}(\text{PHAI}) = \{\text{‘Nam’}, \text{‘Nu’}\}$
 - Bảng tầm ảnh hưởng:

R4	Thêm	Xóa	Sửa
NHANVIEN	+	–	+ (PHAI)



3.2. Liên bộ

- Sự tồn tại của một hay nhiều bộ phụ thuộc vào sự tồn tại của một hay nhiều bộ khác trong cùng quan hệ
- Trường hợp đặc biệt
 - RB khóa chính
 - RB duy nhất (unique)



3.2. Liên bộ (tt)

- **Ví dụ (R5): Mỗi phòng ban có tên duy nhất**
 - Bối cảnh: PHONGBAN
 - Biểu diễn: $\forall t1, t2 \in \text{PHONGBAN} (t1 \neq t2 \Rightarrow t1.TENPHG \neq t2.TENPHG)$
 - Bảng tầm ảnh hưởng:

R5	Thêm	Xóa	Sửa
PHONGBAN	+	-	+ (TENPHG)



3.2. Liên bộ (tt)

- **Ví dụ (R6): Một nhân viên được tham gia tối đa 5 đề án**
 - Bối cảnh: PHANCONG
 - Biểu diễn:

$\forall t \in \text{PHANCONG} \text{ (card}(\{ s \in \text{PHANCONG} \mid s.\text{MA_NVIEN} = t.\text{MA_NVIEN}\}) \leq 5 \text{)}$
hay

$\forall t \in \text{PHANCONG: count}_{(s \in \text{PHANCONG: } s.\text{MA_NVIEN} = t.\text{MA_NVIEN})}(s.\text{MA_NVIEN}) \leq 5$

- **Bảng tầm ảnh hưởng:**

R6	Thêm	Xóa	Sửa
PHANCONG	+	-	+ (MA_NVIEN, SODA)



3.2. Liên bộ (tt)

- Ví dụ (R7): Cho lược đồ THIDAU(NGAY, GIO, DOI, SOBAN). Mỗi trận đấu là cuộc thi đấu của đúng 2 đội
 - Bối cảnh: THIDAU
 - Biểu diễn:
$$\forall t \in \text{THIDAU} (\exists ! s \in \text{THIDAU} (t \neq s \wedge t.\text{NGAY} = s.\text{NGAY} \wedge t.\text{GIO} = s.\text{GIO}))$$
 - Bảng tầm ảnh hưởng:

R7	Thêm	Xóa	Sửa
THIDAU	+	+	+ (NGAY, GIO, DOI)



3.3. Liên thuộc tính

- Là ràng buộc giữa các thuộc tính trong cùng quan hệ
- **Ví dụ (R8): Một nhân viên không quản lý trực tiếp chính mình**
 - Bối cảnh: NHANVIEN
 - Biểu diễn: $\forall t \in \text{NHANVIEN} (t.\text{MA_NQL} \neq t.\text{MANV} \vee t.\text{MA_NQL} = \text{null})$
 - Bảng tầm ảnh hưởng:

R8	Thêm	Xóa	Sửa
NHANVIEN	+ (*)	–	+ (MA_NQL)

Ở thời điểm thêm 1 bộ vào NHANVIEN, MA_NQL là null



3.3. Liên thuộc tính (tt)

- Ví dụ (R9): Cho lược đồ KHOAHOC(MAKH, TENKH, BDAU, KTHUC). Mỗi khóa học kéo dài ít nhất 3 tháng.
 - Bối cảnh: KHOAHOC
 - Biểu diễn: $\forall t \in \text{KHOAHOC} \ (t.\text{KTHUC} - t.\text{BDAU} \geq 3)$
 - Bảng tầm ảnh hưởng:

R9	Thêm	Xóa	Sửa
KHOAHOC	+	-	+ (BDAU, KTHUC)



3. Phân loại RBTV

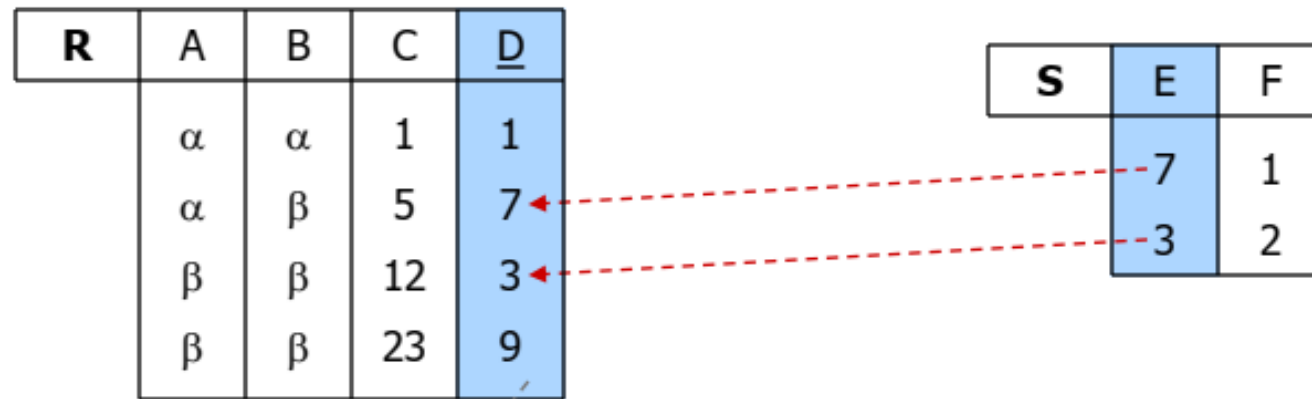
- Một quan hệ
 - Miền giá trị
 - Liên bộ
 - Liên thuộc tính
- Nhiều quan hệ
 - Tham chiếu
 - Liên bộ, liên quan hệ
 - Liên thuộc tính, liên quan hệ
 - Thuộc tính tổng hợp
 - Chu trình

3.4. Tham chiếu

- Giá trị xuất hiện tại các thuộc tính trong một quan hệ nào đó phải tham chiếu đến giá trị khóa chính của một quan hệ khác cho trước
- TH đặc biệt: Ràng buộc khóa ngoại

R	A	B	C	<u>D</u>
	α	α	1	1
	α	β	5	7
	β	β	12	3
	β	β	23	9

S	E	F
	7	1
	3	2



Bắt buộc phải tồn tại trước



3.4. Tham chiếu (tt)

- Ví dụ (R10): Mọi thân nhân phải có mối quan hệ gia đình với một nhân viên trong công ty.
 - Bối cảnh: THANNHAN, NHANVIEN
 - Biểu diễn: $\forall t \in \text{THANNHAN} (\exists s \in \text{NHANVIEN} (s.\text{MANV} = t.\text{MA_NVIENT}))$
hay $\text{THANNHAN.MA_NVIENT} \subseteq \text{NHANVIEN.MANV}$
 - Bảng tầm ảnh hưởng:

R10	Thêm	Xóa	Sửa
NHANVIEN	–	+	– (*)
THANNHAN	+	–	+ (MA_NVIENT)



3.4. Tham chiếu (tt)

- Còn gọi là phụ thuộc tồn tại
- Thường có bối cảnh là 2 quan hệ, nhưng có trường hợp suy biến thành 01 quan hệ
- **Ví dụ (R2): Người quản lý trực tiếp phải là một nhân viên trong công ty**

- **Bối cảnh:** NHANVIEN

- **Biểu diễn:** $\forall t \in \text{NHANVIEN} (t.\text{MA_NQL} \neq \text{null} \wedge$

$$\exists s \in \text{NHANVIEN} (t.\text{MA_NQL} = s.\text{MANV}))$$

- **Bảng tầm ảnh hưởng:**

R2	Thêm	Xóa	Sửa
NHANVIEN	+	+	+(MA_NQL)

3.5. Liên bộ - Liên quan hệ

- Là ràng buộc xảy ra giữa các bộ trên nhiều quan hệ khác nhau

- Ví dụ (R11): HOADON(SOHD, MAKH, NGÀYHD)

CTHD(SOHD, MAHH, DGIA, SLG)

Mỗi hóa đơn phải có ít nhất một chi tiết hóa đơn

- Bối cảnh: HOADON, CTHD
- Biểu diễn: $\forall t \in \text{HOADON} (\exists s \in \text{CTHD} (t.\text{SOHD} = s.\text{SOHD}))$
- Bảng tầm ảnh hưởng:

R11	Thêm	Xóa	Sửa
HOADON	– (*)	–	– (*)
CTHD	–	+	+ (SOHD)



3.6. Liên thuộc tính – Liên quan hệ

- Là ràng buộc xảy ra giữa các thuộc tính trên nhiều quan hệ khác nhau
- **Ví dụ (R12): Ngày sinh của trưởng phòng phải nhỏ hơn ngày nhận chức**
 - Bối cảnh: NHANVIEN, PHONGBAN
 - Biểu diễn:

$\forall t \in \text{PHONGBAN} (\exists s \in \text{NHANVIEN} (s.\text{MANV} = t.\text{TRPHG} \wedge t.\text{NG_NHANCHUC} > s.\text{NGSINH}))$

Bảng tầm ảnh hưởng: R12	Thêm	Xóa	Sửa
NHANVIEN	–	–	+ (NGSINH)
PHONGBAN	+	–	+ (NG_NHANCHUC, TRPHG)



3.7. Thuộc tính tổng hợp

- Thuộc tính tổng hợp: Là thuộc tính có giá trị được tính toán từ các thuộc tính khác
- Khi CSDL có thuộc tính tổng hợp, RBTV bảo đảm quan hệ giữa thuộc tính tổng hợp và các thuộc tính nguồn



3.7. Thuộc tính tổng hợp (tt)

- Ví dụ (R13): PHONGBAN(TENPHG, MAPHG, TRPHG, NG_NHANCHUC, SO_NV)
Số nhân viên của một phòng ban phải bằng tổng số lượng nhân viên thuộc phòng đó
 - Bối cảnh: NHANVIEN, PHONGBAN
 - Biểu diễn: $\forall t \in \text{PHONGBAN} \ (t.\text{SO_NV} = \text{card}\{s \in \text{NHANVIEN} \mid s.\text{PHG} = t.\text{MAPHG}\})$
hay $\forall t \in \text{PHONGBAN}: t.\text{SO_NV} = \text{count}_{(s \in \text{NHANVIEN}: s.\text{PHG} = t.\text{MAPHG})} (*)$
 - Bảng tầm ảnh hưởng

R13	Thêm	Xóa	Sửa
NHANVIEN	+	+	+(PHG)
PHONGBAN	–	–	+(SO_NV)

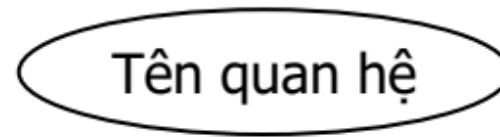
3.8. RBTV chu trình

- Lược đồ CSDL có thể được biểu diễn bằng đồ thị

- Đỉnh:

- Quan hệ

- Thuộc tính



- ● Tên thuộc tính (Attribute name)

- Cạnh: Đường nối một đỉnh quan hệ với một đỉnh thuộc tính

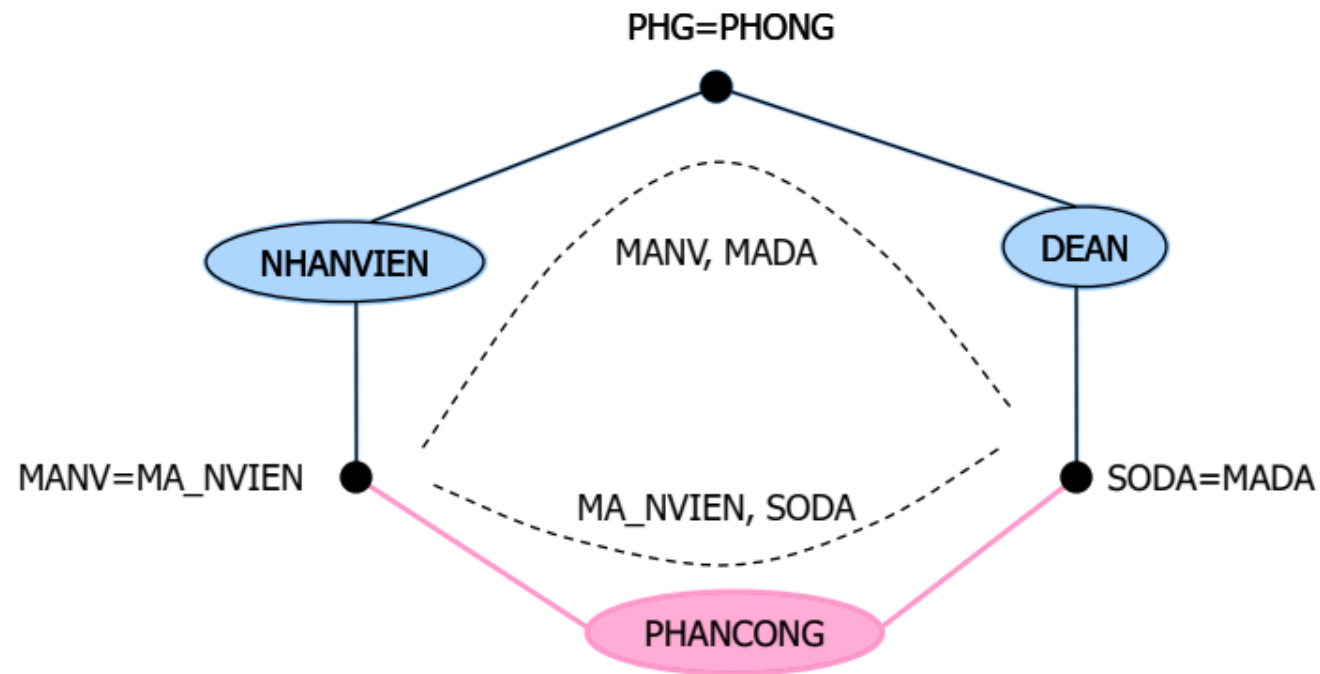


- Chu trình:

- Đồ thị xuất hiện đường đi khép kín ~ Lược đồ CSDL có chu trình

3.8. RBTV chu trình (tt)

- Ví dụ (R14): Nhân viên chỉ được phân công vào các đề án do phòng ban của mình phụ trách





3.8. RBTV chu trình (tt)

- Ví dụ (R14): Nhân viên chỉ được phân công vào các đề án do phòng ban của mình phụ trách

- Bối cảnh: NHANVIEN, DEAN, PHANCONG

- Biểu diễn:

$NVDA \leftarrow (NHANVIEN \bowtie_{PHG = MAPHG} PHONGBAN) \bowtie_{PHONG} DEAN$

$\forall t \in PHANCONG (\exists s \in NVDA (t.MA_NVIEN = s.MANV \wedge t.MADA = s.SODA))$

- Bảng tầm ảnh hưởng

R14	Thêm	Xóa	Sửa
NHANVIEN	–	–	+ (PHG)
DEAN	–	–	+ (PHONG)
PHANCONG	+	–	+ (MA_NVIEN, SODA)



CÀI ĐẶT RBTV

4



4. Cài đặt

- Các RBTV được cài đặt bởi
 - Primary key
 - Foreign key
 - Check constraint
 - Trigger
 - Transaction



4.1. Check constraint

- Ví dụ 15: Lương của trưởng phòng phải lớn hơn 50000

```
ALTER TABLE PHONGBAN (  
    TENPB VARCHAR(20) UNIQUE,  
    MAPHG INT NOT NULL,  
    TRPHG CHAR(9),  
    NG_NHANCHUC DATETIME,  
    CONSTRAINT CHK_PB_LUONGTRPHG CHECK (  
        TRPHG NOT IN (SELECT MANV FROM NHANVIEN  
                        WHERE LUONG < 50000 ))  
    )
```



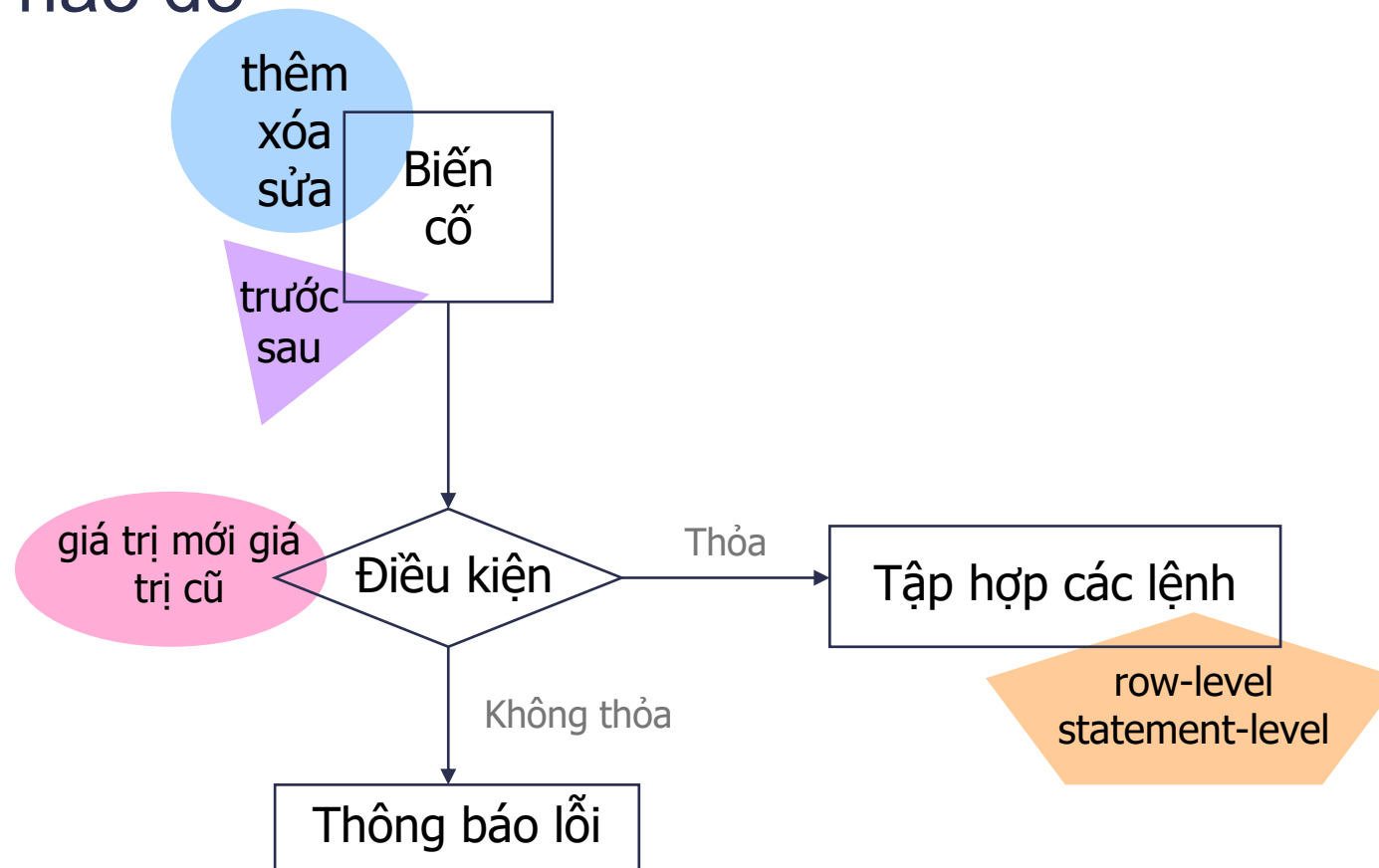
4.1. Check constraint (tt)

- Ví dụ 16: Số lượng nhân viên của mỗi phòng ban không quá 20 người

```
ALTER TABLE NHANVIEN ADD  
CONSTRAINT CHK_NV_SLNVPB CHECK (  
    20 >= ALL ( SELECT COUNT(MANV) FROM NHANVIEN  
                GROUP BY PHG ))
```

4.2. Trigger

- Là tập hợp các lệnh được **thực hiện tự động** khi xuất hiện một **biến cố** nào đó





4.2. Trigger (tt)

- Cú pháp

cú pháp gốc

```
CREATE TRIGGER <Tên_trigger>  
AFTER|BEFORE INSERT|UPDATE|DELETE ON <Tên_bảng>  
REFERENCING  
    NEW ROW|TABLE AS <Tên_1>  
    OLD ROW|TABLE AS <Tên_2>  
FOR EACH ROW | FOR EACH STATEMENT  
WHEN (<Điều kiện>)  
    <Tập_lệnh_SQL>
```

biến dữ liệu mới

biến dữ liệu để trả về trạng
thái trước đó nếu đk ko hợp lệ

```
DROP TRIGGER <Tên_trigger>
```



4.2. Trigger (tt)

- Ví dụ 15: Lương của trưởng phòng phải lớn hơn 50000

```
CREATE TRIGGER TR_PB_UPD  
AFTER UPDATE OF TRPHG ON PHONGBAN  
REFERENCING  
    NEW ROW AS NewTuple  
FOR EACH ROW  
WHEN (50000 >= (SELECT LUONG FROM NHANVIEN  
                WHERE MANV=NewTuple.TRPHG))
```

[Thông báo lỗi cho người dùng]



4.2. Trigger (tt)

- Ví dụ 15: Lương của trưởng phòng phải lớn hơn 50000

```
CREATE TRIGGER TR_PB_UPD
AFTER UPDATE OF TRPHG ON PHONGBAN
REFERENCING
    NEW ROW AS NewTuple
    OLD ROW AS OldTuple
FOR EACH ROW
WHEN (50000 >= (SELECT LUONG FROM NHANVIEN
                WHERE MANV=NewTuple.TRPHG))
UPDATE PHONGBAN
SET TRPHG=OldTuple.TRPHG
WHERE MAPHG=NewTuple.MAPHG
```



4.2. Trigger (tt)

- Ví dụ 15: Lương của trưởng phòng phải lớn hơn 50000

```
CREATE TRIGGER TR_PB_UPD
AFTER UPDATE OF LUONG ON NHANVIEN
REFERENCING
    NEW ROW AS NewTuple
    OLD ROW AS OldTuple
FOR EACH ROW
WHEN (NewTuple.LUONG <= 50000 AND NewTuple.MANV IN (
    SELECT TRPHG FROM PHONGBAN ))
UPDATE NHANVIEN
SET LUONG=OldTuple.LUONG
WHERE MANV=NewTuple.MANV
```



4.2. Trigger (tt)

cú pháp sử dụng thông thường

- Cú pháp

for=after
instead of=before

```
CREATE TRIGGER <Tên_trigger> ON <Tên_bảng>  
FOR INSERT|UPDATE|DELETE  
AS  
BEGIN  
    <Khai báo biến>  
    <Phần kiểm tra>  
    IF <không thỏa điều kiện>  
        begin  
            <thông báo lỗi>  
            Rollback transaction  
        end  
END
```

các biến phải có cùng kiểu giá trị :
+gán 1 thuộc tính cho biến:
@x = select y
+gán nhiều thuộc tính cho nhiều
hoặc 1 biến:
select @x = y

biến của hệ thống
có @@ đằng trước
và datatype đằng
sau
biến thường tương
tự biến của hệ
thống khác ở chỗ
chỉ có 1 @



4.2. Trigger (tt)

- Ví dụ 12: Ngày sinh của trưởng phòng phải nhỏ hơn ngày nhận chức

```
CREATE TRIGGER TRPHG_NS ON PHONGBAN
FOR INSERT
AS BEGIN
    declare @NNC datetime, @NS datetime
    select @NNC = i.NG_NHANCHUC, @NS =
nv.NGSINH
    from Inserted i, NHANVIEN nv
    where i.TRPHG = nv.MANV
    if (@NNC <= @NS) begin
        print (N'Lỗi....')
        --Raiserror(N'Lỗi',16,1)
        Rollback transaction
    end
END
```

transaction học
phía sau



4.3. Transaction

- Là tập các lệnh thực hiện một xử lý nào đó trong một ứng dụng CSDL, sao cho
 - Hoặc là tất cả các lệnh đều được thực hiện thành công
 - Hoặc là không có lệnh nào được thực hiện
- Ví dụ: xử lý chuyển tiền trong ngân hàng

Giao tác Chuyển_tiền

Giảm tiền trong tài khoản người gửi

Tăng tiền trong tài khoản người nhận

Nếu tất cả đều thành công thì hoàn tất giao tác

Ngược lại quay lui giao tác

Cuối giao tác



4.3. Transaction (tt)

- Giao tác phải đảm bảo
 - Tính nguyên tố (atomicity)
 - Tính nhất quán của CSDL (consistency): các RBTV không bị vi phạm
 - Trong khi thực hiện giao tác
 - Trước và sau khi thực hiện giao tác



4.3. Transaction (tt)

- Ví dụ 7: Mỗi trận đấu là cuộc thi đấu của đúng 2 đội

Giao tác Thêm_trận_đấu(t, s)

Thêm t vào THIDAU

Thêm s vào THIDAU

Nếu có một thao tác thất bại thì

Quay lui giao tác

Ngược lại

Hoàn tất giao tác

Cuối lệnh nếu

Cuối giao tác



4.3. Transaction (tt)

- Ví dụ 7: Mỗi trận đấu là cuộc thi đấu của đúng 2 đội

Giao tác Xóa_trận_đấu(ngay, gio)

Với mọi $s \in \text{THIDAU}$ ($s.\text{NGAY} = \text{ngay} \wedge$
 $s.\text{GIO} = \text{gio}$)

Xóa s khỏi THIDAU

Cuối với mọi

Nếu có một thao tác thất bại thì

Quay lui giao tác

Ngược lại

Hoàn tất giao tác

Cuối nếu

Cuối giao tác



4.3. Transaction (tt)

- Ví dụ 11: Mỗi hóa đơn phải có ít nhất một chi tiết hóa đơn

Giao tác Thêm_hóa_đơn

Thêm HOADON

Thêm chi tiết thứ 1 vào CTHD

Thêm chi tiết thứ 2 vào CTHD

...

Nếu có một thao tác thêm thất bại thì

Quay lui giao tác

Ngược lại

Hoàn tất giao tác

Cuối lệnh nếu

Cuối giao tác



4.3. Transaction (tt) – Store procedure

- Các DBMS thương mại cung cấp cách thức lưu trữ các hàm hay thủ tục
- **Cú pháp**

```
CREATE PROCEDURE <Tên_thủ_tục>  
<DS_tham_số>
```

```
AS
```

Khai báo biến cục bộ

Thân chương trình

```
GO
```

```
EXEC <Tên_thủ_tục> <DS_tham_số>
```



4.3. Transaction (tt) – Store procedure (tt)

- Ví dụ 7: Mỗi trận đấu là cuộc thi đấu của đúng 2 đội

```
CREATE PROCEDURE Thêm_trận_đấu
t THIDAU , s THIDAU
AS
    begin tran
        Thêm t vào THIDAU
        If @@error<>0 rollback tran

        Thêm s vào THIDAU
        If @@error<>0 rollback tran
    commit tran
GO

EXEC Thêm_trận_đấu x, y
```




4.3. Transaction (tt)

- DBMS sẽ kiểm tra RBTV
 - Sau khi một thao tác cập nhật diễn ra trên CSDL
 - Cuối mỗi giao tác
- Nên cài đặt RBTV ở đâu ???
 - DBMS
 - Application
- Nên cài đặt loại nào?
 - Trigger quá nhiều → hệ thống chậm
 - Stored Procedure → hiệu quả cao



THẢO LUẬN

