

**VIEW-SERIALIZABLE**

**NỘI DUNG:**

**I. View-Equivalent:**

**II. View-Serializable:**

**III. Xác định lịch có View-Serializable không?**

## I. View-Equivalent:

### Định nghĩa:

- Hai lịch S và S' được gọi là tương đương theo chuẩn View-Equivalent nếu 2 lịch đó thỏa mãn 3 yêu cầu sau :
  1. Nếu trong S có  $W_j(A) \rightarrow R_i(A)$  thì trong S' cũng phải là  $W_j(A) \rightarrow R_i(A)$
  2. Nếu trong S kết thúc bằng  $W_i(A)$  thì trong S' kết thúc cũng bằng  $W_i(A)$
  3. Nếu trong S, Ti bắt đầu đọc  $R_i(A)$  thì trong S' , Ti cũng bắt đầu đọc  $R_i(A)$  .

**Ví dụ:** Xét 2 lịch sau có View-Equivalent không?

S1		
T1	T2	T3
R(A)		
	W(A)	
W(A)		
		W(A)

S2		
T1	T2	T3
R(A)		
W(A)		
	W(A)	
		W(A)

### Hướng dẫn:

*Xét ĐK 1:* Trong trường hợp này 2 lịch thỏa điều kiện này, vì sau khi T1 thực hiện W(A) không có bất kì 1 Ti (i=2,3) thực hiện R(A) → Không cần xét . (Tương tự với các W khác)  
→ *Thỏa ĐK 1.*

*Xét ĐK 2:* Trong S1, T1 bắt đầu thực hiện R(A) và trong S2 cũng vậy → *Thỏa ĐK 2.*

*Xét ĐK 3:* Trong S1, T3 kết thúc thực hiện W(A) và trong S2 cũng vậy → *Thỏa ĐK 3.*

*Từ trên, ta rút ra được kết luận là S1 và S2 tương đương theo chuẩn View-Equivalent.*

## II. View-Serializable :

**Định nghĩa:** 2 lịch S được gọi là View-Serializable khi tồn tại 1 lịch S' tuần tự tương đương với S theo chuẩn View-Equivalent.

**Ví dụ:** Lấy lại ví dụ trên

Ta có S1 và S2 tương đương theo chuẩn View-Equivalent (1)

Và S2 tuần tự (2)

Từ (1) và (2) → S1 là View-Serializable

### III. Cách xác định 1 lịch có View-Serializable không?

**Bước 1:** Chèn 2 giao tác  $T_b$  và  $T_f$  vào lịch với:

- $T_b$  (viết tắt Transaction for beginning) thực hiện Ghi (Write) tất cả các đơn vị dữ liệu và thực hiện trước tất cả các giao tác khác có trong lịch.
- $T_f$  (viết tắt Transaction for finalization) thực hiện Đọc (Read) trên tất cả các đơn vị dữ liệu.

**Ví dụ:** Cho lịch S như sau,

T1	T2	T3
	R(B)	
	W(A)	
R(A)		
		R(A)
W(B)		
	W(B)	

Sau khi thêm 2 giao tác  $T_b$  và  $T_f$  vào ta có:

$T_b$	T1	T2	T3	$T_f$
W(A)				
W(B)				
		R(B)		
		W(A)		
	R(A)			
			R(A)	
	W(B)			
		W(B)		
			W(B)	
				R(A)
				R(B)

**Bước 2:** Lập sơ đồ ưu tiên,

Thành lập các cung theo điều kiện sau:

Nếu trong lịch S có  $W_i(A) \rightarrow R_j(A)$  thì ta lập được cung :  $T_i \rightarrow T_j$  trên đơn vị dữ liệu A

hay  $T_i$  là nguồn của  $R_j(A)$  trên  $T_j$

hay  $T_j$  có nguồn của  $R_j(A)$  là  $T_i$

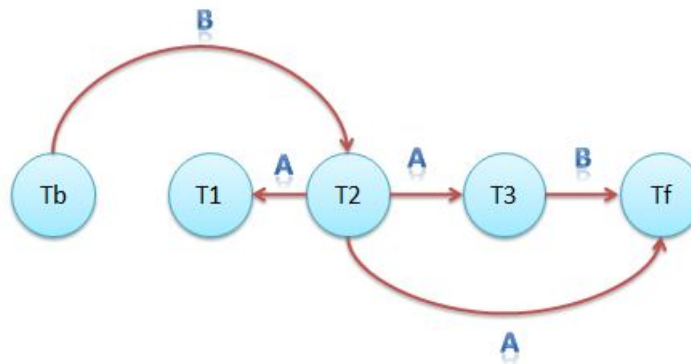
**Ví dụ:** Trở lại ví dụ trên ta suy luận như sau:

- T2 có nguồn của R2( B) là Tb (vì  $Wb(B) \rightarrow R2(b)$ ) nên ta có **Tb  $\rightarrow$  T2 trên A.** <sup>(1)</sup>
- Nguồn của R1(A) trên T1 là T2 (vì T2 ghi lên A trước khi T1 đọc A,  $W2(A) \rightarrow R1(A)$  ) nên suy ra **T2  $\rightarrow$  T1 trên A.** <sup>(2)</sup>

Tương tự ta có:

- Nguồn của R3(A) là T2 nên **T2  $\rightarrow$  T3 trên A.** <sup>(3)</sup>
- Nguồn của Rf(B) trong Tf là T3 nên **T3  $\rightarrow$  Tf trên B.** <sup>(4)</sup>
- Nguồn của Rf(A) trong Tf là T2 nên **T2  $\rightarrow$  Tf trên A.** <sup>(5)</sup>

Từ <sup>(1), (2), (3), (4), (5)</sup> ta có sơ đồ sau:



**Bước 3:** Xét trên từng cặp  $T_i \rightarrow T_j$  trên ĐVDL X:

Ta xét tất cả các thao tác trong giao tác Tk thực hiện Ghi(Write) trên ĐVDL X sao cho Wk phải nằm trước Ti hoặc nằm sau Tj .

Có 3 trường hợp xảy ra:

- $T_i = Tb$  và  $T_j \neq Tf$ : ta thực hiện chèn cung  $T_j \rightarrow Tk$ .
- $T_i \neq Tb$  và  $T_j = Tf$ : ta thực hiện chèn cung  $Tk \rightarrow Ti$ .
- $T_i \neq Tb$  và  $T_j \neq Tf$ : ta thực hiện chèn 2 cung  $Tk \rightarrow Ti$  và  $Tj \rightarrow Tk$ .

**Ví dụ:** Lấy lại ví dụ trên ta thực hiện tiếp, sau bước 2 ta đã có các cung sau:

1. Tb  $\rightarrow$  T2 trên ĐVDL (B) .
2. T2  $\rightarrow$  T1 trên ĐVDL (A) .
3. T2  $\rightarrow$  T3 trên ĐVDL (A).
4. T2  $\rightarrow$  Tf trên ĐVDL (A) .
5. T3  $\rightarrow$  Tf trên ĐVDL (B) .

Xét từng cung ta có:

1.  $T_b \rightarrow T_2$  trên ĐVDL (B).

Tb	T1	T2	T3	Tf
W(A)				
W(B)				
		R(B)		
		W(A)		
	R(A)			
			R(A)	
	W(B)			
		W(B)		
			W(B)	
				R(A)
				R(B)

Xét  $T_b \rightarrow T_2$ , nên ta có  $T_i = T_b$  và  $T_j = T_2$ .

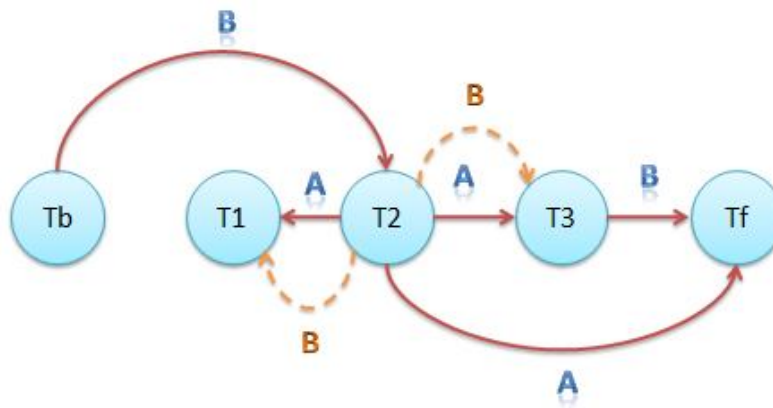
Trên đơn vị dữ liệu B ta có T1 và T3 cũng thực hiện Ghi (Write) B nên suy ra  $k = 1, 3$ .

Ta giả sử  $T_{k1}$  là T1 và  $T_{k2}$  là T3.

Chú ý T1 và T3 đều thực hiện W(B) sau  $T_j$  (ở đây là T2).

$\Rightarrow$  Ta có thêm 2 cung từ  $T_2 (T_j) \rightarrow T_1 (T_{k1})$  và  $T_2 (T_j) \rightarrow T_3 (T_{k2})$ .

Sơ đồ của chúng ta trở thành như sau:



2.  $T2 \rightarrow T1$  trên ĐVDL (A).

Tb	T1	T2	T3	Tf
W(A)				
W(B)				
		R(B)		
		W(A)		
	R(A)			
			R(A)	
	W(B)			
		W(B)		
			W(B)	
				R(A)
				R(B)

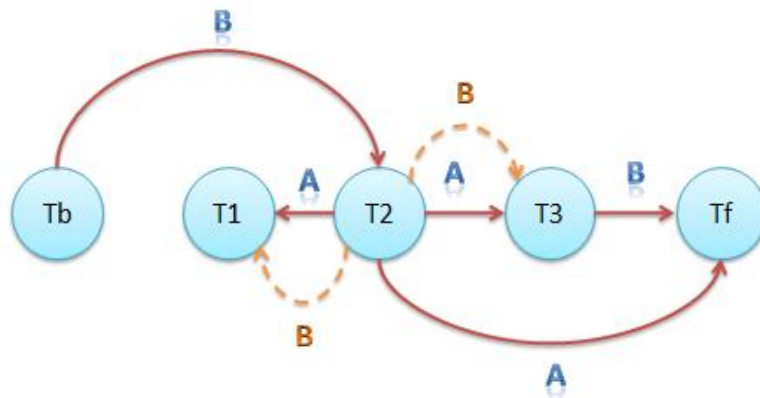
Ta có  $Ti = T2$  và  $Tj = T1$ .

Trên dữ liệu là A ngoài Tb thực hiện W(A) thì không còn giao tác nào khác ghi A nữa.

Xét Tb (Tk), dễ dàng thấy Tb luôn trước Ti và Tj trong mọi trường hợp nên ta lập được cung  $Tb \rightarrow Ti$  ở đây là  $Tb \rightarrow T2$ . Nhìn trong lược đồ hiện tại ta đã có  $Tb \rightarrow Ti$  ở đây là  $Tb \rightarrow T2$ .

Nên ta rút ra 1 suy luận nhỏ: đối với các thao tác ghi thực hiện trên Tb thì ta không cần xét đến trong bước 3 này, hay không thêm bất kì cung nào vào nữa cả.

Cho nên lược đồ hiện tại của chúng ta vẫn là:



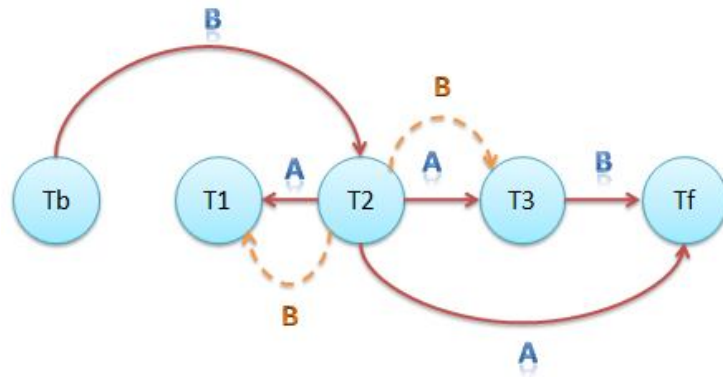
3.  $T2 \rightarrow T3$  trên ĐVDL (A).

Tương tự như trên với ĐVDL là A, ta cũng không thêm được cung nào vào.

4.  $T2 \rightarrow Tf$  trên ĐVDL (A).



Tương tự như trên với ĐVDL là A, ta cũng không thêm được cung nào vào.  
 Lược đồ vẫn là :



5.  $T3 \rightarrow Tf$  trên ĐVDL (B).

Tb	T1	T2	T3	Tf
W(A)				
W(B)				
		R(B)		
		W(A)		
	R(A)			
			R(A)	
	W(B)			
		W(B)		
			W(B)	
				R(A)
				R(B)

Ta có  $T_i = T3$  và  $T_j = Tf$ .

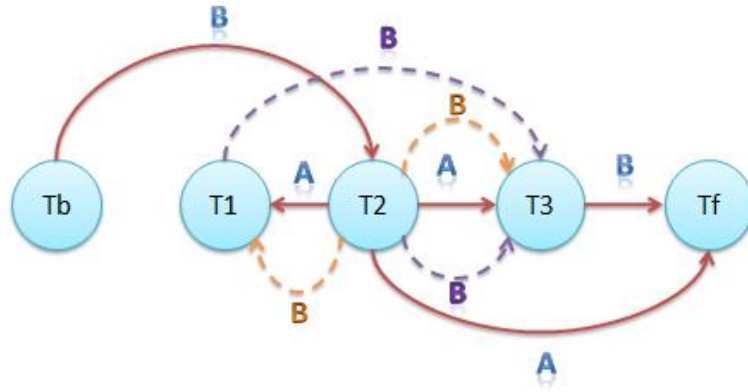
Xét trên ĐVDL B, ta có T1 và T2 cùng ghi trên B và thực hiện W(B) trước  $T_i$  ở đây là T3.

Nên ta lập được thêm 2 cung là:

$T1 \rightarrow T3$

$T2 \rightarrow T3$

Lược đồ của chúng ta như sau:



Như vậy ta đã xét xong 5 cung chính và thêm được 4 cung mới.

Thực hiện xong bước 3 nhé.

**Bước 4:** Xét đồ thị sau khi thực hiện bước 3 có chu trình không.

Nếu không thì xác định các bước tuần tự của đồ thị và kết luận lịch  $S$  khả tuần tự theo View-Serializable .

Nếu có chu trình ta dẫn đến kết luận  $S$  không khả tuần tự theo chuẩn View-Serializable.

**Ví dụ:** Lại dùng ví dụ trên 1 lần nữa (lần cuối cùng :D) .

Xét thấy đồ thị ưu tiên ta lập được sau bước 3 không có chu trình mà tuần tự theo  $T2 \rightarrow T1 \rightarrow T3$

Nên ta kết luận  $S$  khả tuần tự theo chuẩn View-Serializable và lịch tuần tự tương đương với  $S$  theo chuẩn View-Equivalent là  $T2 \rightarrow T1 \rightarrow T3$ :

T1	T2	T3
	R(B)	
	W(A)	
	W(B)	
R(A)		
W(B)		
		R(A)

Hết !!!