Chương 3

KHÔI PHỤC DỮ LIỆU & AN TOÀN DỮ LIỆU

GV: Phạm Thị Bạch Huệ

Email: ptbhue@fit.hcmus.edu.vn

Nội dung

Phần 1: Khôi phục dữ liệu

1. Mục đích.
2. Các lọai sự cố.
3. Các khái niệm liên quan khôi phục dữ liệu
   - Tập tin nhật ký GT (transaction log).
   - Điểm lưu trữ (checkpoint).
4. Các kỹ thuật khôi phục dữ liệu.

Ví dụ

- 2 tài khỏan A = 1000 đ, B = 2000 đ.
- T chuyển 50đ từ A sang B.
- Hệ thống ở tình trạng:
  - Ghi A:= A-50
  - Chưa ghi B:= B+50
  - Khi đó mất điện!
- Khi có điện trở lại
  - Nếu cho T thực hiện lại thì A = 900
  - Nếu không cho T thực hiện lại thì A = 950 và B =2000.
  - Vậy phải thực hiện khôi phục hệ thống như thế nào?

Mục tiêu khôi phục DL

- Khôi phục cơ sở dữ liệu là tiến trình phục hồi cơ sở dữ liệu về tình trạng nhất quán cuối cùng trước khi có sự cố xảy ra.
- Việc khôi phục dữ liệu được thực hiện bởi bộ quản lý khôi phục dữ liệu (RM – Recovery Manager).
- Khôi phục dữ liệu tự động giúp giảm thiểu việc yêu cầu người sử dụng thực hiện lại công việc (*khi xảy ra sự cố*).

Mục tiêu khôi phục DL

- Giao tác là đơn vị cơ bản khi khôi phục CSDL.
- Trong 4 tính chất của GT(ACID), RM bảo đảm 2 tính chất, đó là tính nguyên tố (Atomic) và tính bền bỉ (Durability).

Các loại sự cố

- Sự cố của giao tác
  - Giao tác bị rollback do deadlock hay do bộ lập lịch yêu cầu (thực hiện lại).
  - Khi xảy ra sự cố giao tác, hệ thống vẫn bình thường.
  - Tần suất: vài lần/phút.
- Sự cố hệ thống
  - Hệ thống không thể tiếp tục thực hiện được nữa. Nguyên nhân có thể do lỗi trong bộ xử lý, bị mất điện hay do lỗi của phần mềm.
  - Khi xảy ra sự cố hệ thống, chỉ mất những thông tin trên bộ nhớ chính.
  - Tần suất: vài lần/tháng.

Các loại sự cố

- Sự cố thiết bị lưu trữ (media failure)
  - Ví dụ: đĩa bị hư, đầu đọc bị hư,...
  - Khi xảy ra sự cố thiết bị lưu trữ, có thể bị mất 1 phần hay toàn bộ dữ liệu
  - Tần suất: vài lần/năm
- Lỗi phần mềm ứng dụng
  - Lỗi logic của chương trình truy cập CSDL, làm cho việc thực hiện giao tác không thành công.

Nhận xét

- Dù bất cứ nguyên nhân gì, ta cần xem xét 2 khả năng khi phục hồi sự cố:

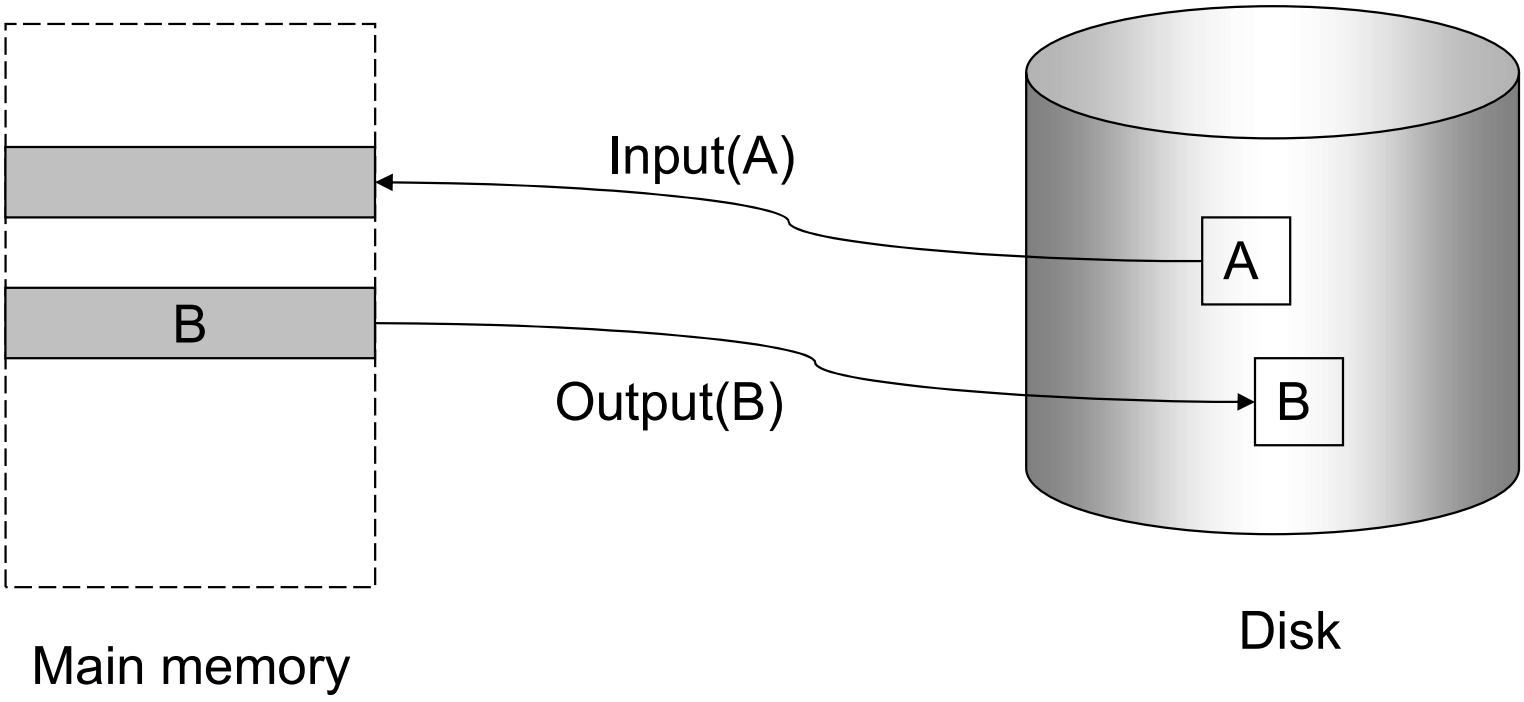
1. Mất dữ liệu trên bộ nhớ chính (database buffer).
2. Mất dữ liệu trên bộ nhớ phụ.

Sao chép dữ liệu

- DBMS cung cấp cơ chế cho phép sao chép cơ sở dữ liệu phòng khi CSDL bị sự cố.
  - Sao chép toàn bộ hoặc chỉ sao chép những thay đổi kể từ lần sao chép cuối.
  - Giải quyết được trường hợp mất dữ liệu trên bộ nhớ phụ.

Việc truy xuất dữ liệu

- Dữ liệu được đọc từ đĩa hoặc ghi vào đĩa tính theo đơn vị là khối (block).
  - Physical block: khối dữ liệu được lưu trên đĩa.
  - Buffer block: khối dữ liệu được lưu tạm thời trên main memory.



Việc truy xuất dữ liệu

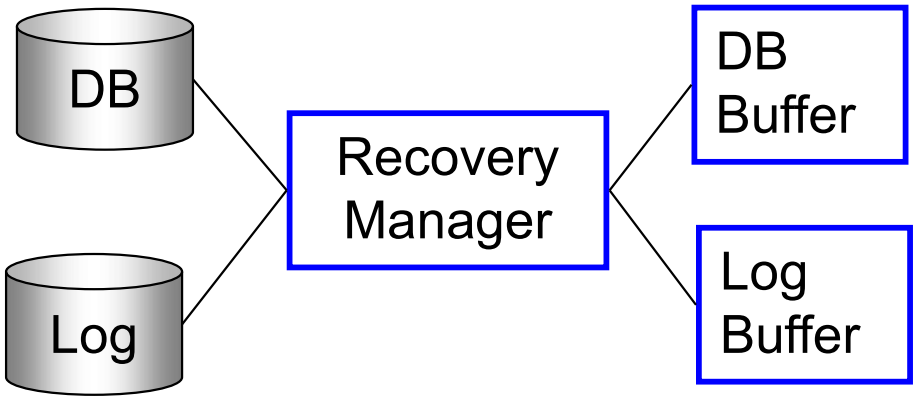
- Read (X): gán X cho biến cục bộ xi
  - Nếu khối DL có chứa X chưa có trong buffer, thực hiện Input (X).
  - Gán giá trị X (chứa trên buffer block) cho biến cục bộ xi.
- Write (X): gán xi cho X
  - Nếu khối DL có chứa X chưa có trong buffer, thực hiện Input(X).
  - Gán giá trị xi cho X (trên buffer block có chứa X).
- **Việc Đọc/ Ghi trên dữ liệu được thực hiện gián tiếp qua buffer.**

Quản lý buffer

- Buffer:
  - Dữ liệu mất khi có sự cố hệ thống.
  - Không gian hạn chế.
- Chiến lược thay thế để định ra vùng trống trên buffer dùng để nạp dữ liệu mới.
  - FIFO.
  - LRU.

Nhận xét

- Database gồm có 2 phần:
  - Database Vật lý và
  - Buffer cho Database
- Log gồm có 2 phần:
  - Log Vậl lý và
  - Buffer cho Log

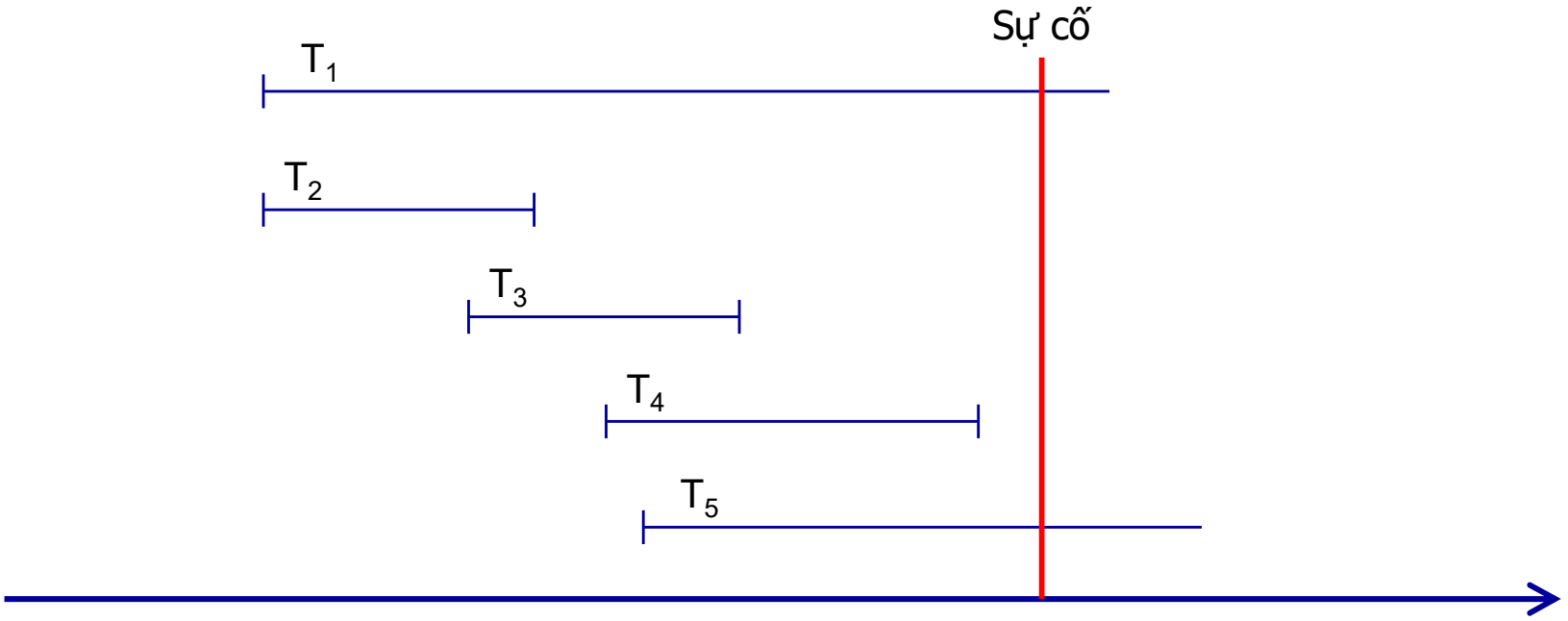


- Khi có sự cố ⇒ mất thông tin trên buffer cho database ⇒ phải có thể dựa vào log file để phục hồi dữ liệu.
- Khi có sự cố ⇒ mất thông tin trên buffer cho log, nghĩa là thông tin về những thao tác cập nhật lên CSDL (chưa được thực hiện thật sự lên CSDL Vật lý) sẽ bị mất và cần phải thực hiện lại các thao tác này.

Nhận xét

- Dữ liệu trên buffer được ghi (flush) bền bỉ xuống đĩa dựa vào một số sự kiện, ví dụ:
  - Một câu lệnh cụ thể được thực hiện (ví dụ commit).
  - Hoặc ghi tự động mỗi khi buffer đầy.
- Việc ghi tường minh từ buffer xuống đĩa gọi là force-writing.
- Trong khoảng thời gian từ khi thao tác ghi trên buffer xảy ra đến khi thao tác flush từ buffer xuống bộ nhớ phụ, nếu sự cố xảy ra thì RM phải xác định trạng thái của GT thực hiện thao tác ghi tại thời điểm xảy ra sự cố.
  - Nếu GT đã commit, RM phải redo những cập nhật của GT (rollforward) để đảm bảo tính bền bỉ.
  - Nếu GT chưa commit, RM phải undo (rollback) những thay đổi của GT trên CSDL nhằm đảm bảo tính nguyên tố.

Ví dụ



Khi sự cố xảy ra, T2 , T3, T4 đã commit nên RM phải ghi nhận những thay đổi của chúng lên CSDL khi ht khởi động lại.

T1 và T5 phải được undo

- DBMS thường cung cấp các tiện ích sau để hỗ trợ cho quá trình phục hồi dữ liệu:
  - RM.
  - Cơ chế backup.
  - Ghi nhật ký (log).
  - Checkpoint.

Steal & No-force

- RM dùng hai giải pháp sau để ghi dữ liệu từ buffer xuống đĩa:
  - Steal policy: buffer manager ghi từ buffer xuống đĩa trước khi GT commit. Ngược lại là no-steal, nghĩa là không ghi gì cả trước khi GT commit.
  - Force policy: dữ liệu từ buffer do 1 GT cập nhật lập tức được ghi xuống đĩa khi GT commit. Ngược lại là no-force.
- Với no-steal, không phải undo những thay đổi do giao tác bị hủy thực hiện vì những thay đổi chưa được ghi xuống đĩa.
- Với force, không phải redo những thay đổi thực hiện bởi các GT đã commit.
- Steal policy tránh tình trạng còn quá nhiều dữ liệu trên buffer lẽ ra nên ghi xuống đĩa.
- No force có lợi khi 2 GT cùng làm việc trên 1 block, GT sau không phải nạp lại block từ đĩa lên buffer.
- Hầu hết các DBMS dùng chính sách steal, no-force.

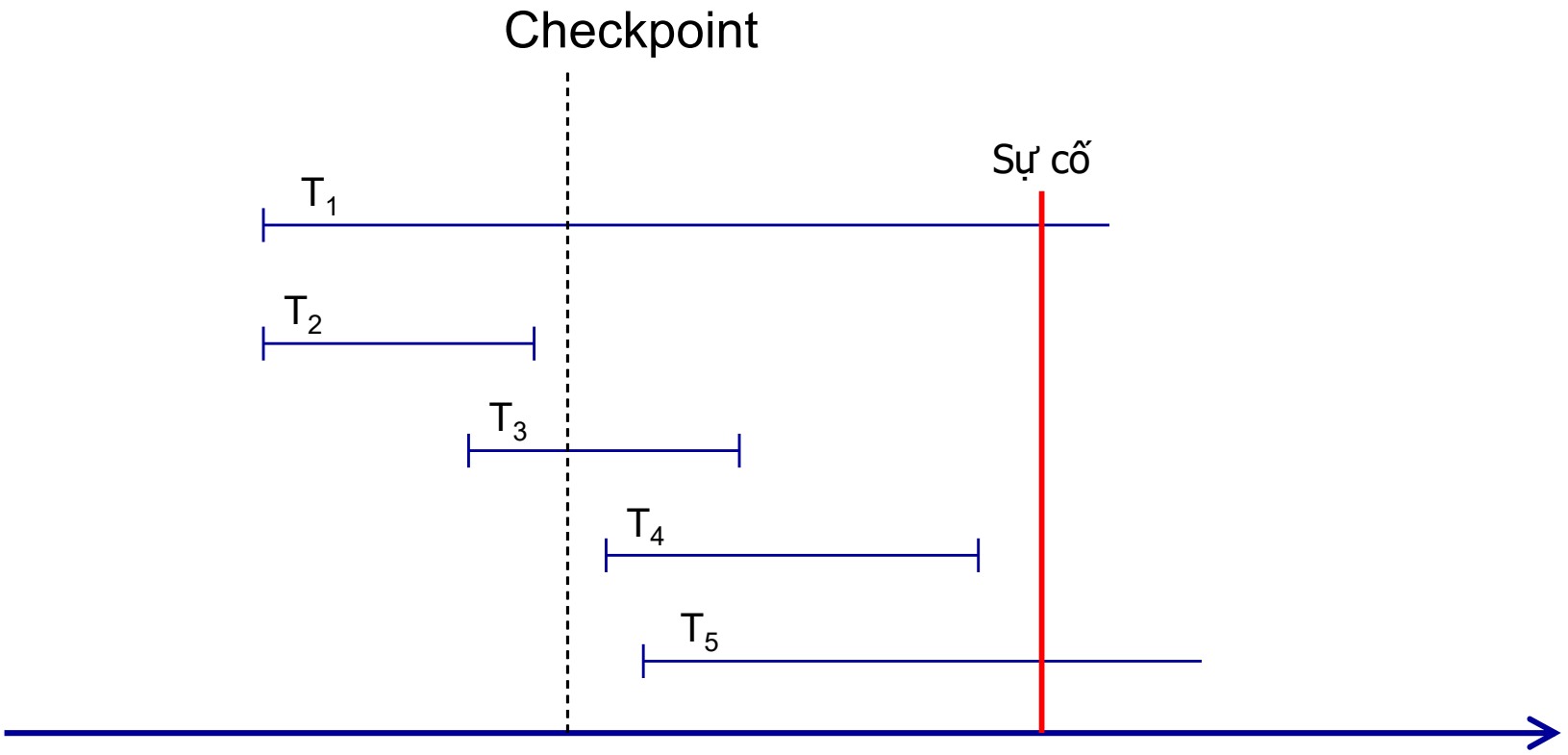
Log file

- Log file chứa các thông tin về mọi sự thay đổi trên CSDL.
- Log file được dùng cho xử lý phục hồi dữ liệu. Log file chứa các thông tin:
  - Transaction record:
    1. ID của giao tác.
    2. Kiểu mẩu tin nhật ký (GT bắt đầu, insert, update, delete, abort, commit)
    3. ID của đơn vị dữ liệu bị cập nhật.
    4. Giá trị cũ của đơn vị dữ liệu ⇔ Before Image
    5. Giá trị mới của đơn vị dữ liệu ⇔ After Image
    6. Con trỏ để quản lý các mẫu tin trong tập tin log.
  - Checkpoint record.
- Vì log file là rất quan trọng, thường có 2 hoặc 3 tập tin log được tạo ra.

Checkpoint

- Hạn chế của pp khôi phục dùng log
  - Quét toàn bộ log → thời gian tìm kiếm tăng.
  - Không cần thực hiện lại những giao tác đã ghi chắc chắn lên CSDL.
  - Checkpoint dùng để cải thiện quá trình phục hồi.
- Các checkpoint định kỳ xảy ra:
  - Ghi tất cả log record có nội dung thay đổi từ bộ nhớ chính xuống bộ nhớ phụ.
  - Ghi xuống CSDL tất cả những gì thay đổi trên buffer CSDL.
  - Ghi checkpoint record vào log file.
- RM quyết định thời gian định kỳ thực hiện checkpoint, sau m phút hoặc sau t giao tác commit kể từ lần checkpoint trước.

Ví dụ dùng checkpoint



T2 đã được ghi xuống bộ nhớ phụ nên không cần redo T2.

Các kỹ thuật phục hồi

- Kỹ thuật phục hồi dùng cách cập nhật trì hoãn (Recovery techniques using deferred update)
- Kỹ thuật phục hồi dùng cách cập nhật tức thì (Recovery techniques using immediate update)

Kỹ thuật phục hồi dùng cập nhật trì hoãn

- CSDL không được cập nhật cho đến khi GT đã commit. (Dựa trên chính sách no-steal.)
- Nếu GT gặp sự cố trước khi commit thì không cần thực hiện undo gì cả.
- Nhưng cần redo những cập nhật cho các GT commit.
- PP này sử dụng log như sau:
  - GT bắt đầu, ghi nhận lại.
  - Không thay đổi database buffer hoặc CSDL.
  - Ghi log record vào đĩa, ghi dòng commit của GT.
  - Dùng log record để cập nhật thật sự.
  - Nếu 1 GT abort, bỏ qua các thao tác của GT đó và không làm gì cả.

Kỹ thuật phục hồi dùng cách cập nhật tức thì

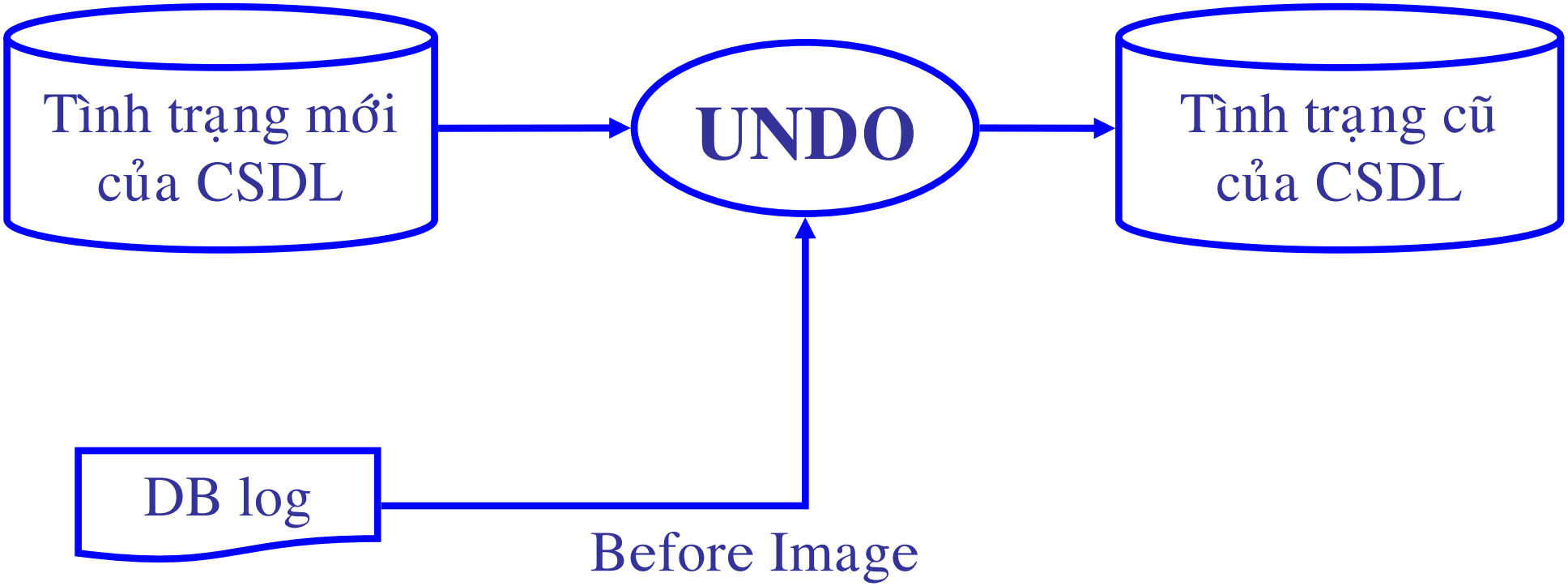
- Cập nhật trên CSDL mỗi khi có GT thực hiện thay đổi CSDL, không đợi đến khi GT kết thúc.
- Khi sự cố xảy ra:
  - Vừa redo các thao tác cập nhật của GT commit.
  - Vừa undo các thao tác cập nhật của GT chưa commit.
- Dùng log file như sau:
  - GT bắt đầu, ghi nhận lại.
  - Ghi nhận lại các thao tác ghi của GT vào log file.
  - Ghi nhận lại thao tác ghi trên database buffer.
  - Database buffer được ghi xuống đĩa khi đến lúc phải ghi.
  - Ghi nhận commit, nếu GT commit.

Nghi thức WAL (Write Ahead Log)

- Dữ liệu từ DB Buffer có thể được ghi nhận lên DB Vật lý trước khi giao tác được thật sự commit hay rollback. Vậy nếu giao tác phải rollback thì cần tiến hành việc undo dựa vào thông tin ghi trên log file (vật lý) => Cần phải cập nhật trên log file vật lý trước khi cập nhật lên CSDL vật lý.
- Phải viết vào log file vật lý trước khi viết vào CSDL vật lý (Write Ahead Log Protocol).

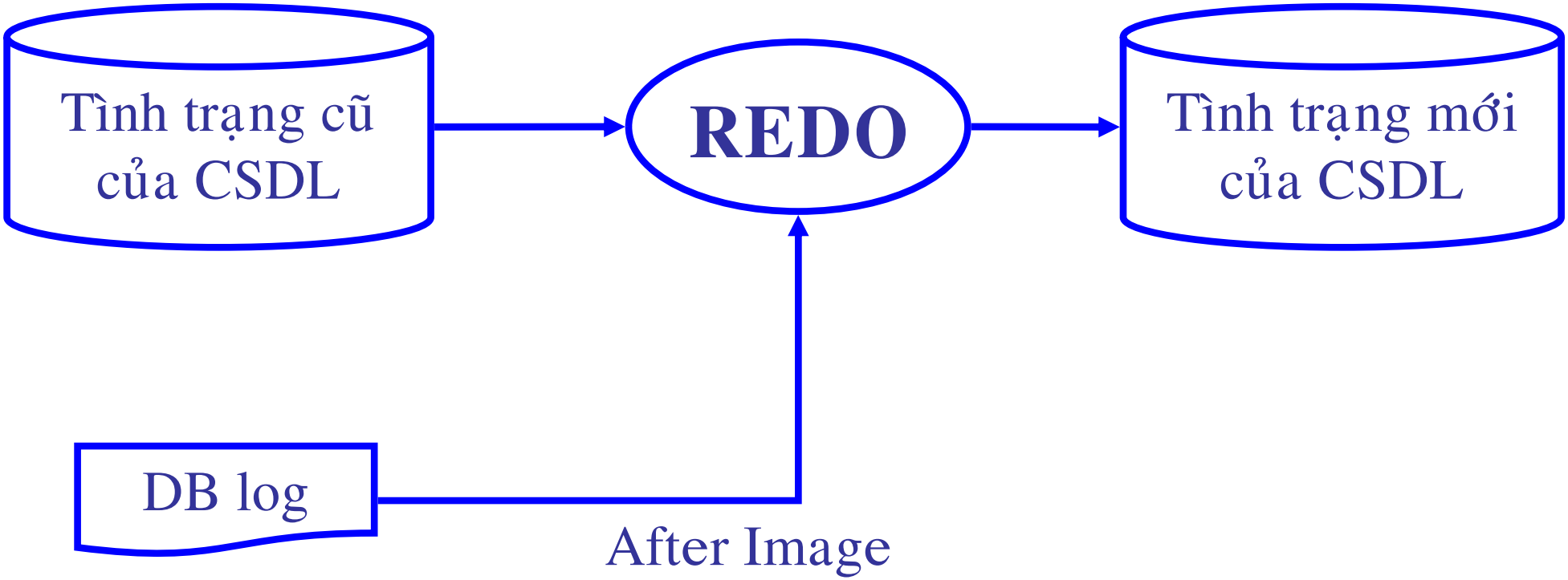
Nghi thức UNDO

- Thực hiện cho những giao tác chưa kết thúc nhưng xảy ra sự cố hay những giao tác bị rollback.



Nghi thức REDO

- Thực hiện cho những giao tác đã commit nhưng chưa được ghi nhận trên CSDL Vật lý.



Phục hồi bình thường

- Sau 1 cái dừng bình thường của hệ thống, 1 điểm checkpoint được ghi vào log file như là mẫu tin cuối cùng của log file.
- Khi hệ thống được khởi động lại, nếu mẫu tin cuối cùng trong log file là checkpoint thì thủ tục phục hồi bình thường được gọi (*nói chung là không phải thực hiện thao tác undo hay redo nào cả*).

Phục hồi khi có sự cố

- Nếu mẫu tin cuối cùng của log file là checkpoint thì không cần xét tiếp.
- Ngược lại, xác định checkpoint cuối cùng trong log file.
- Xác định 2 nhóm giao tác:
  - Nhóm 1: Giao tác đã commit trước khi xảy ra sự cố hệ thống.
  - Nhóm 2: gồm 2 loại
    - Giao tác chưa được commit trước khi xảy ra sự cố hệ thống.
    - Giao tác bị rollback trước khi xảy ra sự cố hệ thống.
- Với các giao tác thuộc nhóm 1: Áp dụng nghi thức Redo.
- Với các giao tác thuộc nhóm 2: Áp dụng nghi thức Undo.

Một số quy ước

- Undo các thao tác có dấu ↑
  - ↑ Cần undo thật sự trên CSDL vật lý dựa vào before image.
  - [↑] Không cần undo trên CSDL vật lý vì thao tác này sau checkpoint cuối cùng, những thay đổi chỉ mới trên log file.
- Redo các thao tác có dấu ×

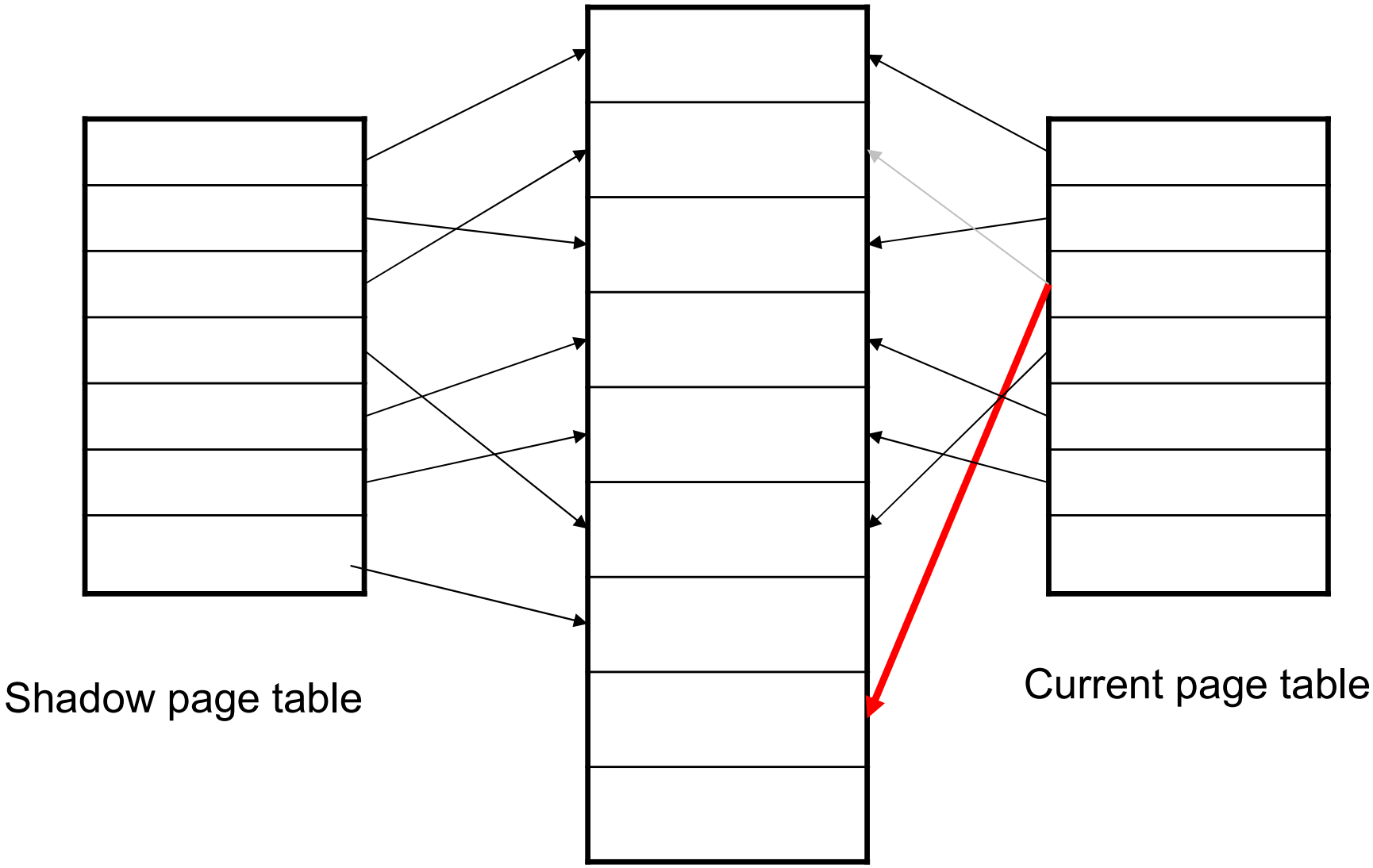
Ví dụ

BOT_i	**Bắt đầu giao tác T_i**
U1(i)	**Cập nhật lần 1 của Ti**
BOTi+1	**Bắt đầu giao tác Ti+1**
U1(i+1) ↑	**Thao tác cập nhật thứ 1 của giao tác Ti+1**
Checkpoint	
BOTi+2	**Bắt đầu giao tác Ti+2**
U1(i+2) ×	**Thao tác cập nhật thứ 1 của giao tác Ti+2**
U2(i) ×	**Thao tác cập nhật thứ 2 của giao tác Ti**
Commit Ti	**Commit Ti**
U2(i+1) [↑]	**Thao tác cập nhật thứ 2 của giao tác Ti+1**
BOTi+3	**Bắt đầu giao tác Ti+3**
U1(i+3) [↑]	**Thao tác cập nhật thứ 1 của giao tác Ti+3**
U2(i+3) [↑]	**Thao tác cập nhật thứ 2 của giao tác Ti+3**
U2(i+2) ×	**Thao tác cập nhật thứ 2 của giao tác Ti+2**
Commit Ti+2	**Commit Ti+2**
U3(i+1) [↑]	**Thao tác cập nhật thứ 3 của giao tác Ti+1**
	Sự cố hệ thống xảy ra

Shadow paging

- Một phương pháp phục hồi khác (pp log) là dùng trang bóng.
- Suốt một quá trình sống của 1 giao tác, có 2 bảng được duy trì:
  - Bảng trang hiện hành (current page table): sẽ bị thay đổi khi T thực hiện ghi.
  - Bảng trang bóng (shadow page table): bảng sao của table trước khi T thực hiện.
  - Khi giao tác khởi động, 2 page table này giống nhau.

Shadow paging



Hết chương 3 phần 1.