Course code   : **CSE2007**
Course title   : **Database Management System**
Module   : **5**
Topic   : **6**

Shadow Paging

Objectives

This session will give the knowledge about

- Shadow Paging

Shadow Paging

No log required in a single-user environment

- Log may be needed in a multiuser environment for the concurrency control method

Shadow paging considers disk to be made of n fixed-size disk pages

- Directory with n entries is constructed

- When transaction begins executing, directory copied into shadow directory to save while current directory is being used

- Shadow directory is never modified

Shadow Paging

New copy of the modified page created and stored elsewhere

- Current directory modified to point to new disk block

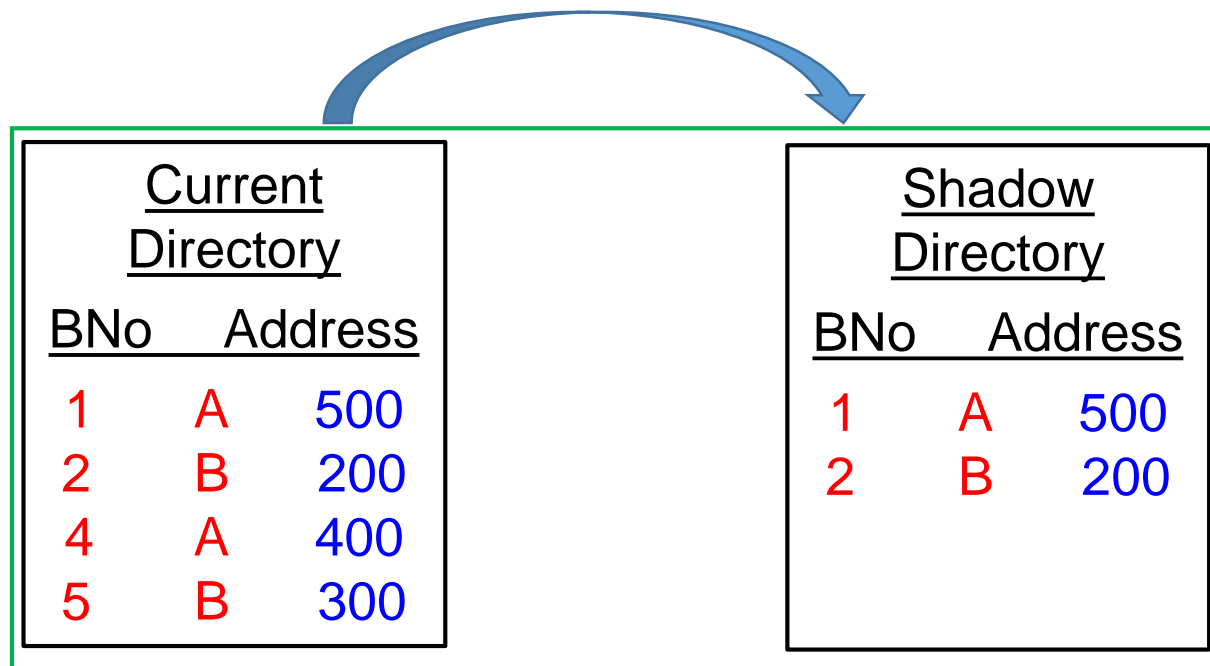- Shadow directory still points to old disk block

Failure recovery

- Discard current directory

- Free modified database pages
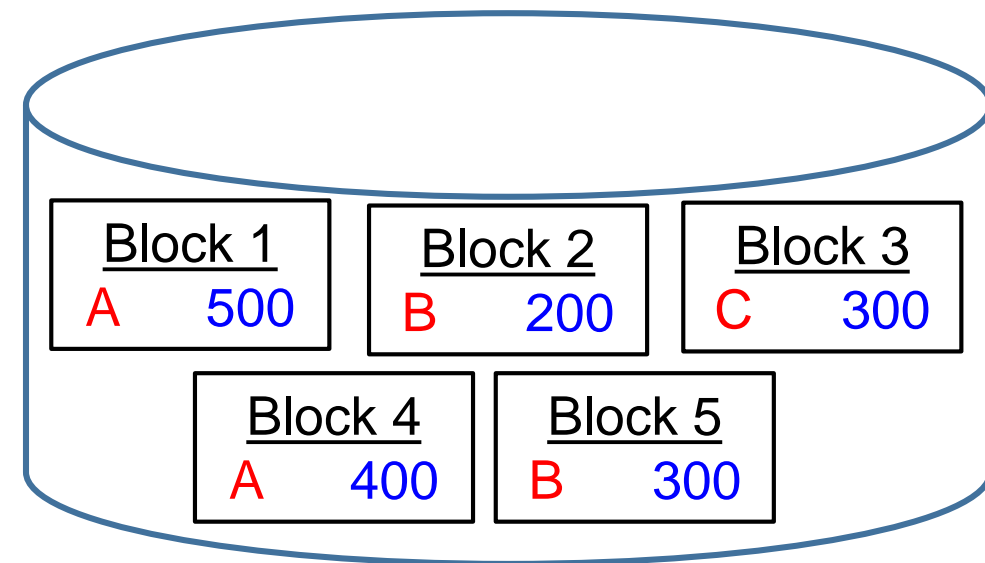
- NO-UNDO/NO-REDO technique

Steps in Shadow Paging

- Construct a Directory with every entry in the directory pointing to a database block.

- Create a Shadow copy of the directory known as Shadow Directory

- Transaction should modify only pages pointed bu the current directory and write it into a new location on the disk.

- If transaction commits, discard the shadow directory and its pointing old disk blocks

- If transaction fails, discard the current directory and the and its newly created disk blocks
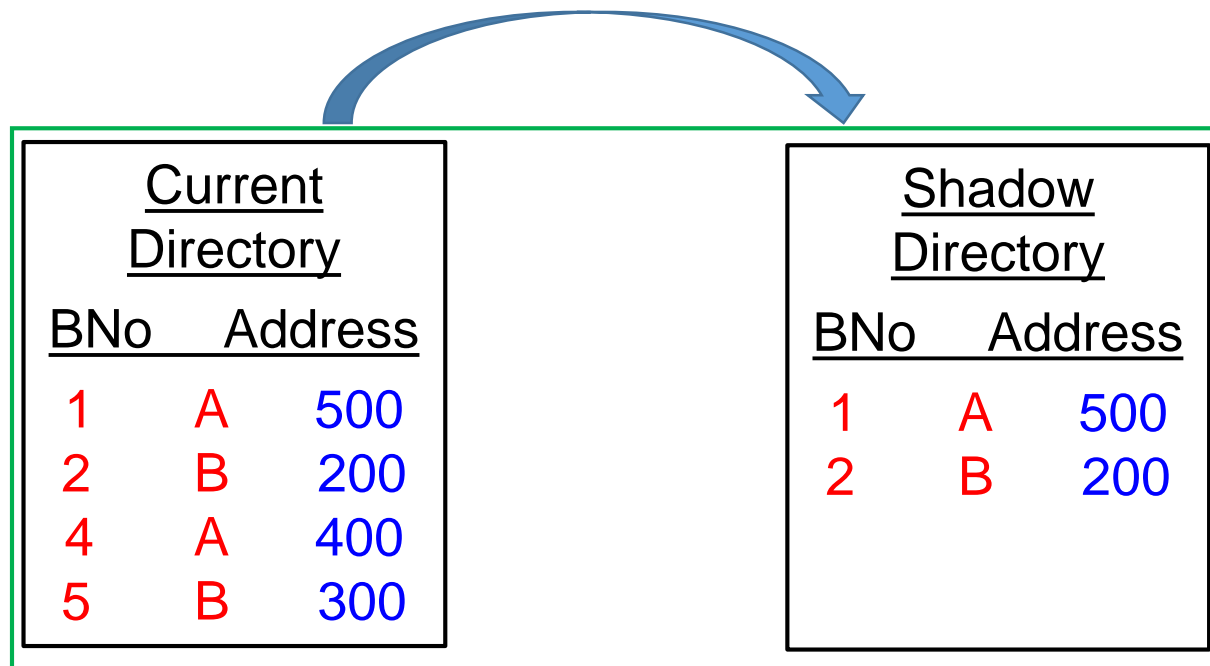
Shadow Paging

Current Directory

BNo	Address	
1	A	500
2	B	200
4	A	400
5	B	300

Shadow Directory

BNo	Address	
1	A	500
2	B	200

2 READ(A)

3 A=A-100

4 WRITE(A)

5 READ(B)

6 B=B+100

7 WRITE(B)

8 COMMIT

Block 1

A	500

Block 2

B	200

Block 3

C	300

Block 4

A	400

Block 5

B	300

Shadow Paging

Current Directory

BNo		Address
1	A	500
2	B	200
4	A	400
5	B	300

Shadow Directory

BNo		Address
1	A	500
2	B	200

2 READ(A)
3 A=A-100
4 WRITE(A)
5 ABORT

Block 1
A   500

Block 2
B   200

Block 3
C   300

Block 4
A   400

Block 5
B   300

Shadow Paging

2 READ(A)

3 A=A-100

4 WRITE(A)

5 READ(B)

6 B=B+100

7 WRITE(B)

8 ABORT

Current Directory		
BNo	Address	
1	A	500
2	B	200
4	A	400
5	B	300

Shadow Directory		
BNo	Address	
1	A	500
2	B	200

Block 1	
A	500

Block 2	
B	200

Block 3	
C	300

Block 4	
A	400

Block 5	
B	300
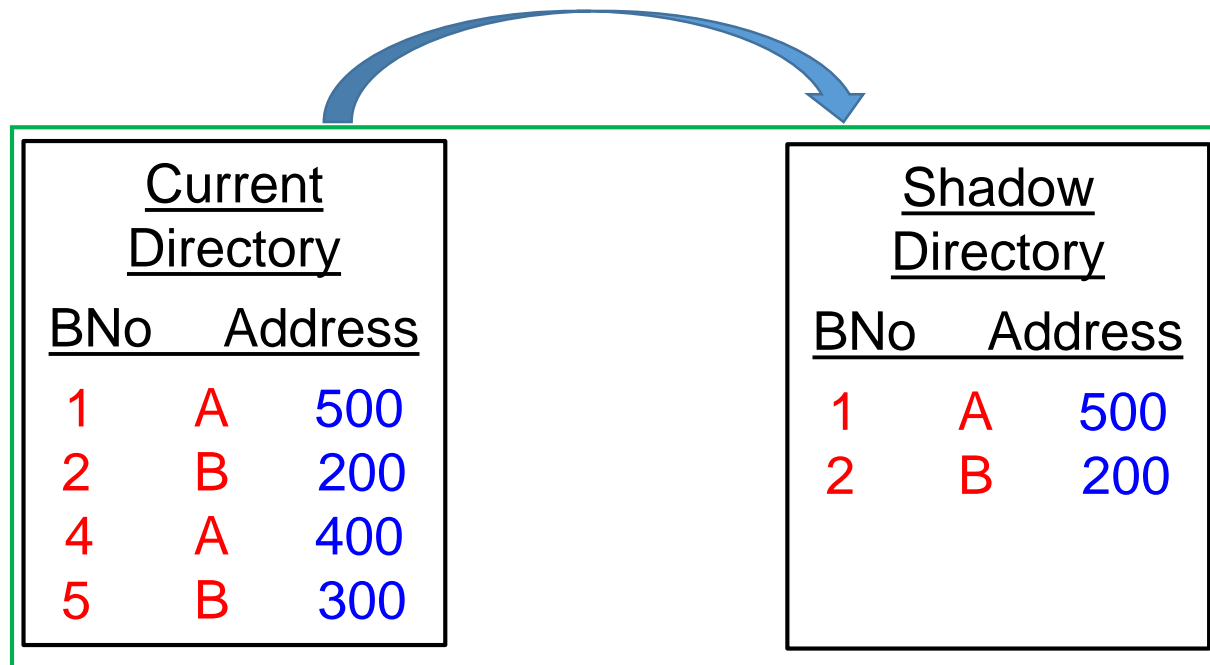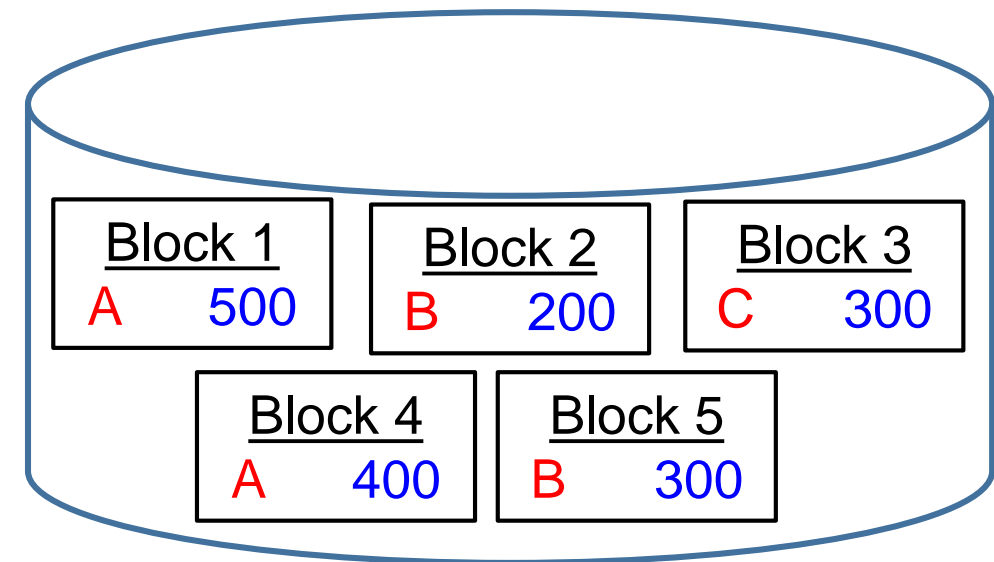
[start_transaction, T_1]
[read_item, T_1, A]
[read_item, T_1, D]
[write_item, T_1, D, 20, 25]
[commit, T_1]
[checkpoint]
[start_transaction, T_2]
[read_item, T_2, B]
[write_item, T_2, B, 12, 18]
[start_transaction, T_4]
[read_item, T_4, D]
[write_item, T_4, D, 25, 15]
[start_transaction, T_3]
[write_item, T_3, C, 30, 40]
[read_item, T_4, A]
[write_item, T_4, A, 30, 20]
[commit, T_4]
[read_item, T_2, D]
[write_item, T_2, D, 15, 25] ◄——— System crash

Figure 22.6 shows the log corresponding to a particular schedule at the point of a system crash for four transactions T_1, T_2, T_3, and T_4. Suppose that we use the *immediate update protocol* with checkpointing. Describe the recovery process from the system crash. Specify which transactions are rolled back, which operations in the log are redone and which (if any) are undone, and whether any cascading rollback takes place.

Summary

This session will give the knowledge about

- Shadow Paging