Failure Recovery

Berikut ini adalah sebuah *schedule* yang dihasilkan dari eksekusi konkuren 3 buah transaksi.
R1(A); R2(B); R3(C); R1(B); R2(D); R3(A); W2(D); A2; W3(C); C3; W1(A); C1; R2(B); R2(D); W2(D); W2(A); C2;

a. Tuliskan isi log hasil eksekusi *schedule* tersebut. Asumsikan bahwa start dari setiap transaksi dimulai tepat sebelum eksekusi instruksi pertama transaksi tersebut. Nilai A, B, C, dan D sebelum eksekusi *schedule* adalah A_0, B_0, C_0, dan D_0, dan setiap penulisan nilai item X yang semula X_i akan mengubah nilainya menjadi X_{i+1}.
b. Apabila *checkpoint* terakhir terjadi tepat setelah T_2 menyelesaikan proses *rollback* dan terjadi *system failure* tepat sebelum T_2 commit (saat dijalankan kembali setelah *abort*), jelaskan proses *recovery* yang dilakukan setelah sistem pulih kembali.

Solution Alternative

Problem a

	Content of Log
R1(A);	<T_1 start>
R2(B);	<T_2 start>
R3(C);	<T_3 start>
R1(B); R2(D); R3(A);	
W2(D);	<T_2, D, D_0, D_1>
A2;	<T_2, D, D_0>
	<T_2 abort>
W3(C);	<T_3, C, C_0, C_1>
C3;	<T_3 commit>
W1(A);	<T_1, A, A_0, A_1>
C1;	<T_1 commit>
R2(B);	<T_2 start>
R2(D);	
W2(D);	<T_2, D, D_0, D_1>
W2(A);	<T_2, A, A_1, A_2>
C2;	<T_2 commit>

Problem b

Content of Log

<T_1 start>
<T_2 start>
<T_3 start>
<T_2, D, D_0, D_1>
<T_2, D, D_0>
<T_2 abort>
<checkpoint {T_1, T_3}>
<T_3, C, C_0, C_1>
<T_3 commit>
<T_1, A, A_0, A_1>
<T_1 commit>
<T_2 start>
<T_2, D, D_0, D_1>
<T_2, A, A_1, A_2>
--system failure--

Proses recovery (*recovery process*):

1. Scan backward dari entri log terakhir hingga menemukan checkpoint record.
   Scan backward from the last log entry to find checkpoint record
   UL = {T_1, T_3}
2. Mulai dari checkpoint, scan forward untuk melakukan proses redo sambil mengupdate isi UL.
   Start from checkpoint, scan forward to redo transactions while updating UL.

<T_3, C, C_0, C_1>	C diisi C_1.
<T_3 commit>	UL = {T_1}
<T_1, A, A_0, A_1>	A is updated to A_1.
<T_1 commit>	UL = {}
<T_2 start>	UL = {T_2}
<T_2, D, D_0, D_1>	D diisi D_1.
<T_2, A, A_1, A_2>	A diisi A_2.

3. Scan backward dari entri log terakhir untuk melakukan undo terhadap transaksi yang masih ada di UL.
   Scan backward from the last log entry to undo transcations remaining in UL.

<T_2 start>	UL = {}, undo finishes
<T_2, D, D_0, D_1>	Write redo log record, update D to D_0.
<T_2, A, A_1, A_2>	Write redo log record, update A to A_1.
<T_2, A, A_1>	
<T_2, D, D_0>	
<T_2 abort>	

<T_1 start>
<T_2 start>
<T_3 start>
<T_2, D, D_0, D_1>
<T_2, D, D_0>
<T_2 abort>
<checkpoint {T_1, T_3}>
<T_3, C, C_0, C_1>
<T_3 commit>
<T_1, A, A_0, A_1>
<T_1 commit>
<T_2 start>
<T_2, D, D_0, D_1>
<T_2, A, A_1, A_2>

<T_2, A, A_1>
<T_2, D, D_0>
<T_2 abort>

KESALAHAN UMUM:

- TIDAK MENULISKAN ACTIVE TRANSACTION PADA CHECKPOINT ⇒ HARUSNYA DITULIS

- OPERASI READ DITULISKAN DI LOG ⇒ HARUSNYA TIDAK PERLU