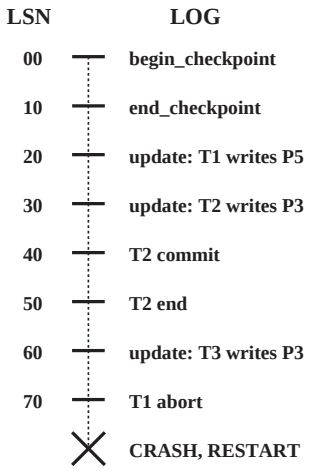

Figure 18.1 Execution with a Crash

3. Redoable log records are update log records and compensation log records; executing the actions indicated by these records several times is equivalent to executing them once.

4. A compensation log record (CLR) C describes the action taken to undo the actions recorded in the corresponding update log record U. (This can happen during normal system execution when a transaction is aborted, or during recovery from a crash.) The compensation log record C also contains a field called undonextLSN which is the LSN of the next log record that is to be undone for the transaction that wrote update record U; this field in C is set to the value of prevLSN in U.

   Unlike an update log record, a CLR describes an action that will never be undone. An aborted transaction will never be revived, therefore once a CLR has properly returned the data its previous state, both transactions can be forgotten.

Exercise 18.3 Briefly answer the following questions:

1. What are the roles of the Analysis, Redo, and Undo phases in ARIES?

2. Consider the execution shown in Figure 18.1.

   (a) What is done during Analysis? (Be precise about the points at which Analysis begins and ends and describe the contents of any tables constructed in this phase.)

   (b) What is done during Redo? (Be precise about the points at which Redo begins and ends.)

   (c) What is done during Undo? (Be precise about the points at which Undo begins and ends.)

Answer 18.3 The answer to each question is given below.

1. The Analysis phase starts with the most recent begin_checkpoint record and proceeds forward in the log until the last log record. It determines

   (a) The point in the log at which to start the Redo pass

   (b) The dirty pages in the buffer pool at the time of the crash.

   (c) Transactions that were active at the time of the crash which need to be undone.

   The Redo phase follows Analysis and redoes all changes to any page that might have been dirty at the time of the crash. The Undo phase follows Redo and undoes the changes of all transactions that were active at the time of the crash.

2. (a) For this example, we will assume that the Dirty Page Table and Transaction Table were empty before the start of the log. Analysis determines that the last begin_checkpoint was at LSN 00 and starts at the corresponding end_checkpoint (LSN 10).

   We will denote Transaction Table records as (transID, lastLSN) and Dirty Page Table records as (pageID, recLSN) sets.

   Then Analysis phase runs until LSN 70, and does the following:

LSN 20	Adds (T1, 20) to TT and (P5, 20) to DPT
LSN 30	Adds (T2, 30) to TT and (P3, 30) to DPT
LSN 40	Changes status of T2 to "C" from "U"
LSN 50	Deletes entry for T2 from Transaction Table
LSN 60	Adds (T3, 60) to TT. Does not change P3 entry in DPT
LSN 70	Changes (T1, 20) to (T1, 70)

   The final Transaction Table has two entries: (T1, 70), and (T3, 60). The final Dirty Page Table has two entries: (P5, 20), and (P3, 30).

   (b) Redo Phase: Redo starts at LSN 20 (smallest recLSN in DPT).

LSN 20	Changes to P5 are redone.
LSN 30	P3 is retrieved and its pageLSN is checked. If the page had been written to disk before the crash (i.e. if $pageLSN >= 30$), nothing is re-done otherwise the changes are re-done.
LSN 40,50	No action
LSN 60	Changes to P3 are redone
LSN 70	No action

   (c) Undo Phase: Undo starts at LSN 70 (highest lastLSN in TT). The Loser Set consists of LSNs 70 and 60. LSN 70: Adds LSN 20 to the Loser Set. Loser Set = (60, 20). LSN 60: Undoes the change on P3 and adds a CLR indicating this

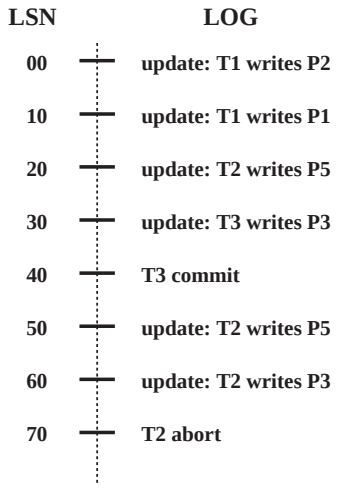

Figure 18.2 Aborting a Transaction

Undo. Loser Set = (20). LSN 20: Undoes the change on P5 and adds a CLR indicating this Undo.

Exercise 18.4 Consider the execution shown in Figure 18.2.

1. Extend the figure to show prevLSN and undonextLSN values.
2. Describe the actions taken to rollback transaction $T2$.
3. Show the log after $T2$ is rolled back, including all prevLSN and undonextLSN values in log records.

Answer 18.4 The answer to each question is given below.

1. The extended figure is shown below:

LSN	prevLSN	undonextLSN(of a CLR corresponds to the ULR)
00	–	–
10	00	00
20	–	–
30	–	–
40	30	– (not an update log record)
50	20	20
60	50	50
70	60	– (not an update log record)

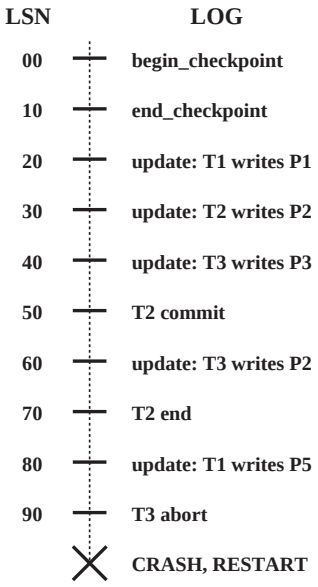

Figure 18.3 Execution with Multiple Crashes

2. Step i) Restore P3 to the before-image stored in LSN 60.
   Step ii) Restore P5 to the before-image stored in LSN 50.
   Step iii) Restore P5 to the before-image stored in LSN 20.

3. The log tail should look something like this:

LSN	prevLSN	transID	type	pageID	undonextLSN
80	70	T2	CLR	P3	50
90	80	T2	CLR	P5	20
100	90	T2	CLR	P5	–
110	100	T2	END	–	–

Exercise 18.5 Consider the execution shown in Figure 18.3. In addition, the system crashes during recovery after writing two log records to stable storage and again after writing another two log records.

1. What is the value of the LSN stored in the master log record?
2. What is done during Analysis?
3. What is done during Redo?
4. What is done during Undo?