

answer - no. 1

1605023

Site 3 has initiated transaction T to transfer Hc. 1000 from account P at site 4 to account Q at site 2. Write down the tasks of TC3.

Solution

tasks of TC3

- i. TC3 starts the execution of the transaction T.
- ii. There are two sub-transactions  $T_1$  and  $T_2$  here:
  - ( $T_1$ ) update account P (withdraw Hc. 1000)
  - ( $T_2$ ) update account Q (deposit Hc. 1000).

Here, TC3 assigns these two sub-transactions  $T_1$  and  $T_2$  to  $TM_4$  and  $TM_2$  respectively.

- iii. TC3 coordinates the termination of all parallel sub-transactions (commitment or abortion of sub-transactions at sites 4 and 2).

Ans.

answer - 20.2

1605023

Date :

site 1 has initiated transaction T to transfer tk. 1000 from account A at site 3 to account B at site 2 and transaction T1 to add tk. 5000 to account C at site 3. Write down the tasks of TM3.

solution

tasks of TM3

i. There are, in total, two sub-transactions in which TM3 is involved:

(T<sub>A</sub>) update account A at site 3 (withdraw tk. 1000)

(T<sub>C</sub>) update account C at site 3 (deposit tk. 5000).

TM3 maintains logs for these sub-transactions for recovery purpose.

ii. TM3, along with TC1, coordinates the execution and termination of the transactions at site 3.

Ans.

answer - 21.1

1605023

Why is Commit protocol needed to commit the transaction T initiated by TC1 in given scenario? Explain.

Solution

If the entire system, considered here, was a centralized one, then a commit at site 1 would be enough to terminate the transaction. But, we are dealing with a distributed system. Hence, transaction may fail, or problems may arise at different sites at different points of time during execution. Therefore, commit protocols are necessary to ensure atomicity across sites.

Ans.

answer - 21.2

1605023

Write down the activities of the coordinator of site 1 in phase 1 and phase 2 for the case when all the sites including coordinator wants to commit.

Solution

activities of TC1

phase - 1

- i. TC1 asks TM2 and TM3 to prepare to commit transaction T by adding record  $\langle \text{prepare } T \rangle$  to log, forcing log for T to stable storage, and sending  $\langle \text{prepare } T \rangle$  message to TM2 and TM3.

phase - 2

- i. Upon receiving  $\langle \text{ready } T \rangle$  messages from TM2 and TM3, TC1 adds a decision record  $\langle \text{commit } T \rangle$  to log and forces log for T to stable storage.
- ii. TC1, then, sends  $\langle \text{commit } T \rangle$  message to TM2 and TM3 informing them of the decision.

Ans.



answer - 21.3

1605023

Write down the activities of the site3 in phase 1 and coordinator in phase 2 in the case when site3 does not want to commit.

Solution

activities of TM3 in phase 1

- i. Upon receiving  $\langle \text{prepare } T \rangle$  message from TC1, TM3 adds  $\langle \text{no } T \rangle$  record to the log and sends  $\langle \text{abort } T \rangle$  message back to TC1.

activities of TC1 in phase 2

- i. Upon receiving  $\langle \text{abort } T \rangle$  message from TM3, TC1 adds a decision record  $\langle \text{abort } T \rangle$  to the log and forces log for T to stable storage.
- ii. TC1 sends  $\langle \text{abort } T \rangle$  message to TM2 and TM3 informing them of the decision afterwards.

Ans.

answer - 22.1

1605023

Site 3 has initiated transaction T3 to transfer tk. 1000 from account P at site 4 to account Q at site 2. Explain the atomicity of T3 for different types of site and coordinator failures.

Solution

Here, TC3 is the coordinator and site 4 along with site 2 is the actively participating site. TM4 and TM2 manage the transaction T3 between accounts P and Q.

atomicity of T3 for site failure

In this case, site 4 or site 2 may fail during T3. When the failed site recovers, it examines its log to determine the fate of T3. If the log contains  $\langle \text{commit } T3 \rangle$  or  $\langle \text{abort } T3 \rangle$  record, then  $\text{redo}(T3)$  or  $\text{undo}(T3)$  operation must be carried out. If the log contains  $\langle \text{ready } T3 \rangle$  record, then the site must consult with TC3 to decide on the fate of T3. If none of the aforementioned scenario holds, then TC3 must abort T3 and site must carry out  $\text{undo}(T3)$  operation.

atomicity of T3 for coordinator failure

In this case, TC3 may fail during the execution of commit protocol for T3. Then, TM4 and TM2 must decide on T3's fate by looking into their respective

logs. If the log contains <commit T3> record, then manager must commit T3. If the log contains <abort T3> record, then manager must abort T3. If the log contains <ready T3> record, then manager must wait for TC3 to recover to find decision. If none of the above three records is in the log, then manager will abort T3.

Thus, atomicity is maintained for transaction T3 in different cases of failures.

Ans.