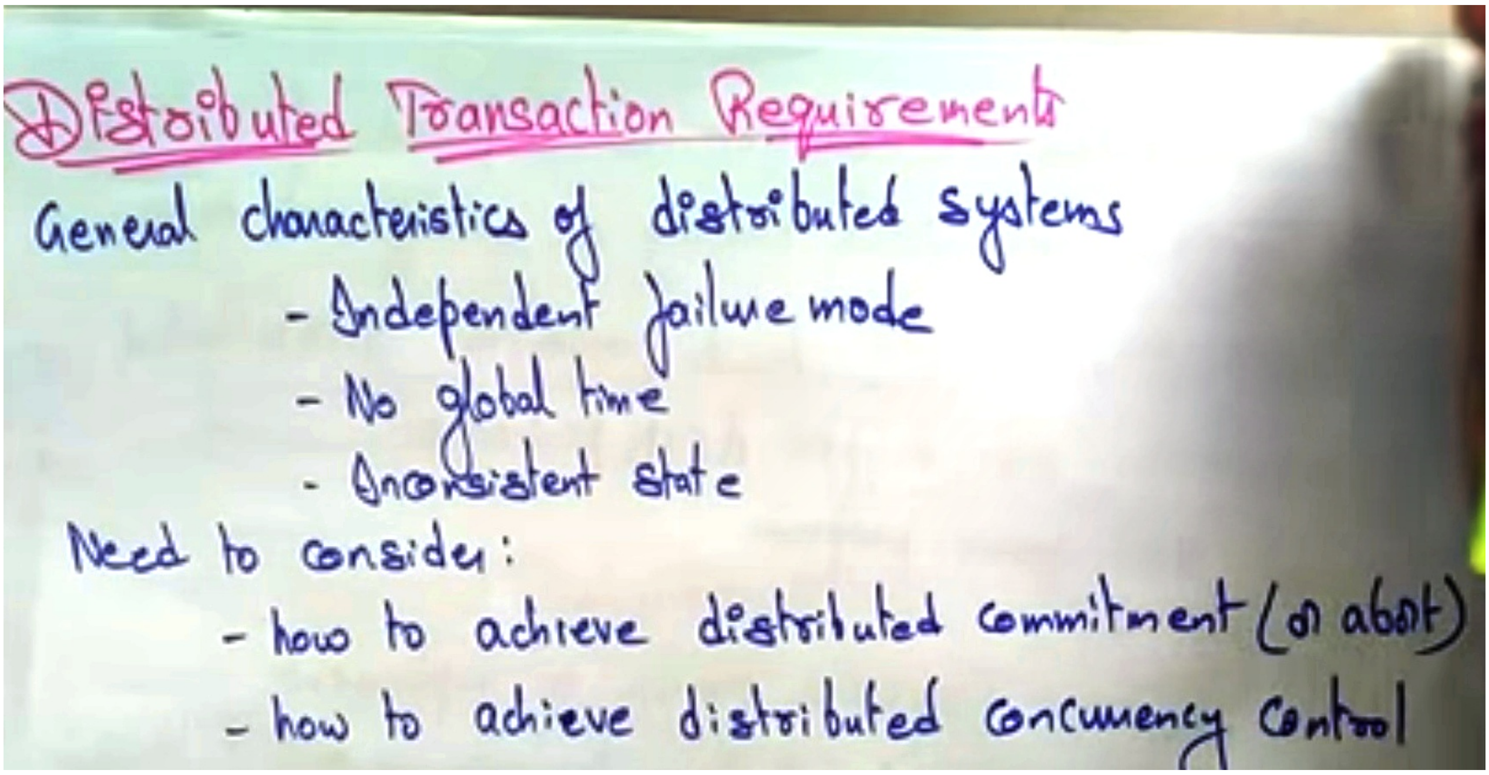


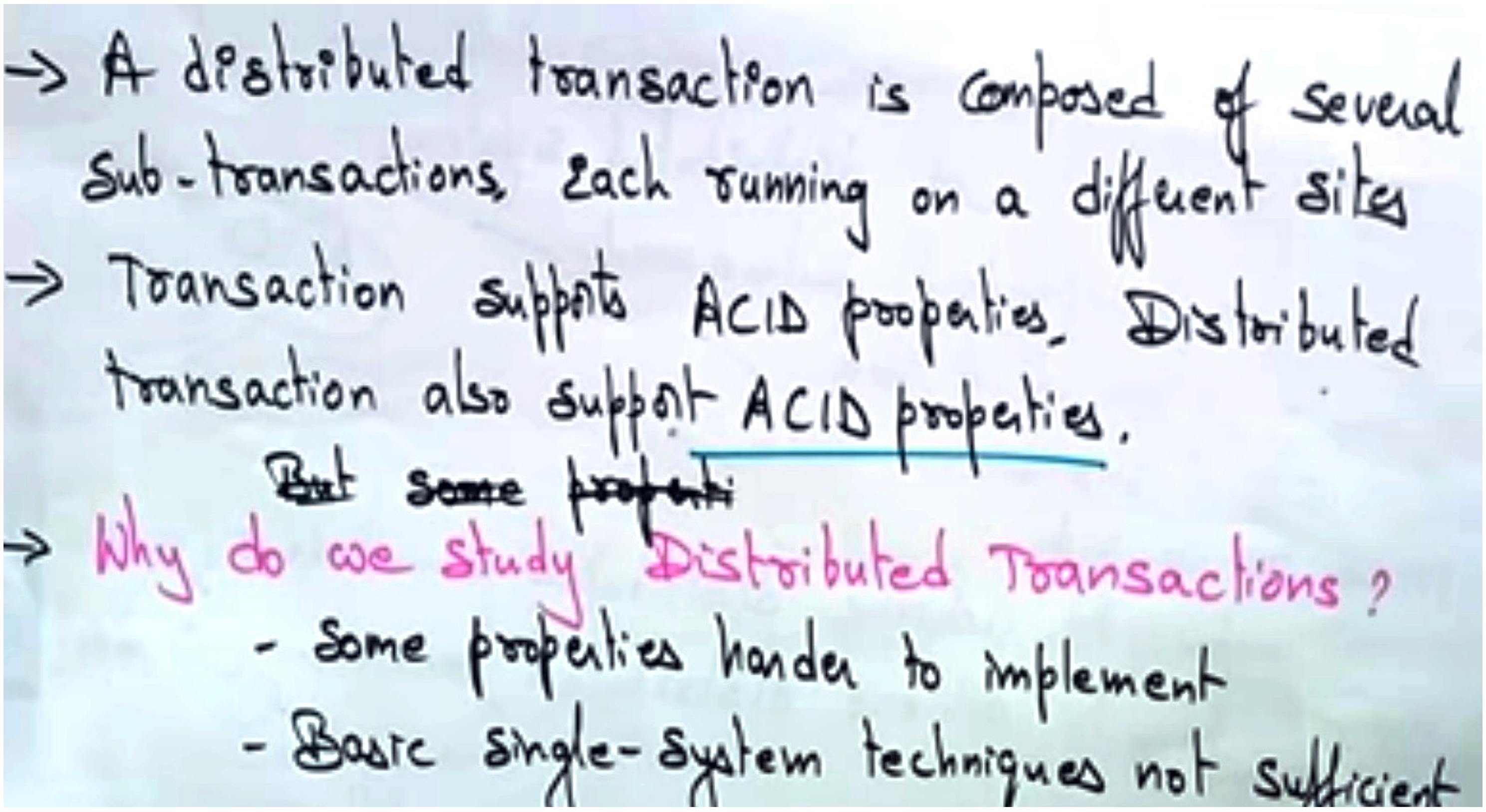
Distributed Transaction Requirements

General characteristics of distributed systems

- Independent failure mode
- No global time
- Inconsistent state

Need to consider:

- how to achieve distributed commitment (or abort)
- how to achieve distributed concurrency control



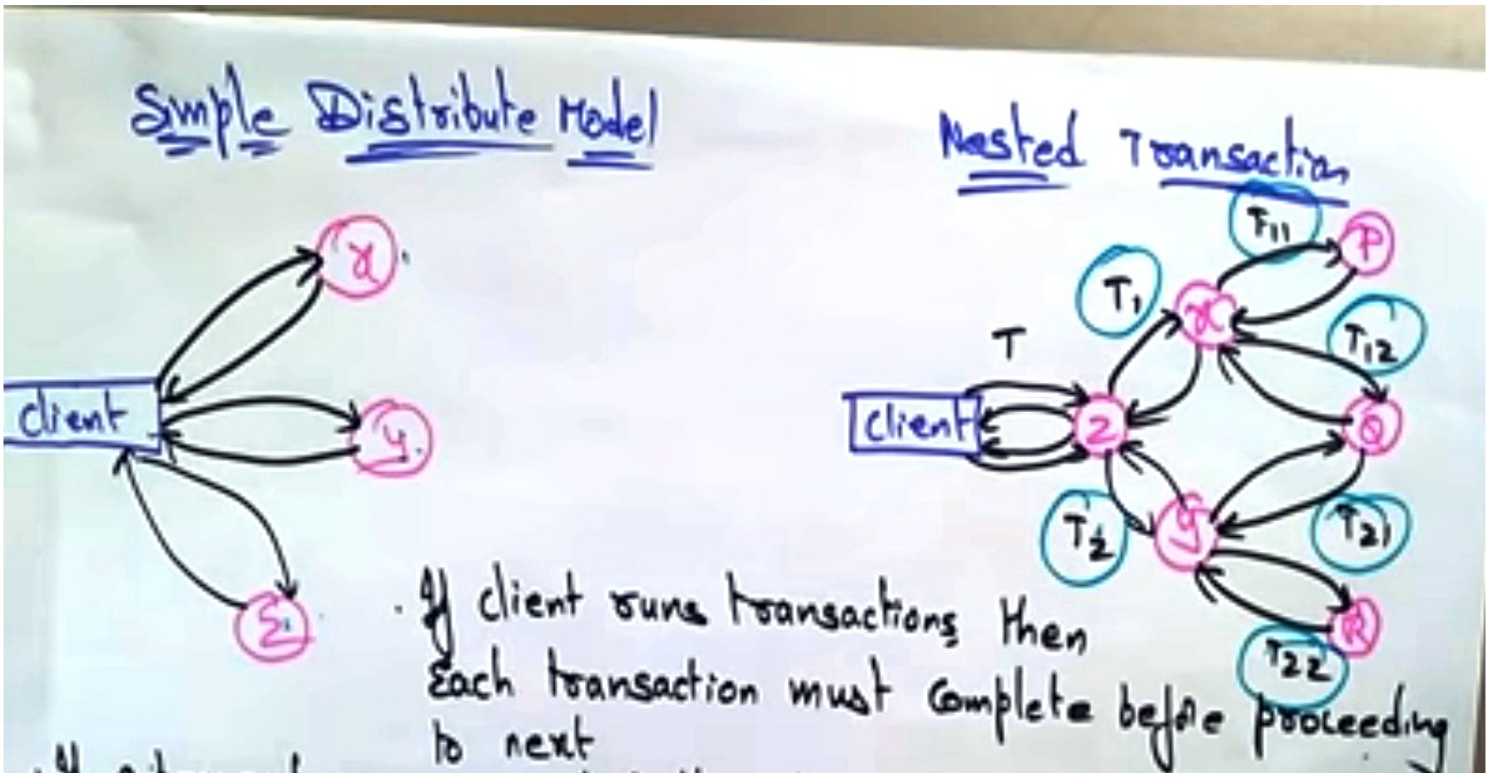
→ A distributed transaction is composed of several sub-transactions, Each running on a different sites

→ Transaction supports ACID properties. Distributed transaction also support ACID properties.

~~But some properti~~

→ Why do we study Distributed Transactions?

- Some properties harder to implement
- Basic single-system techniques not sufficient

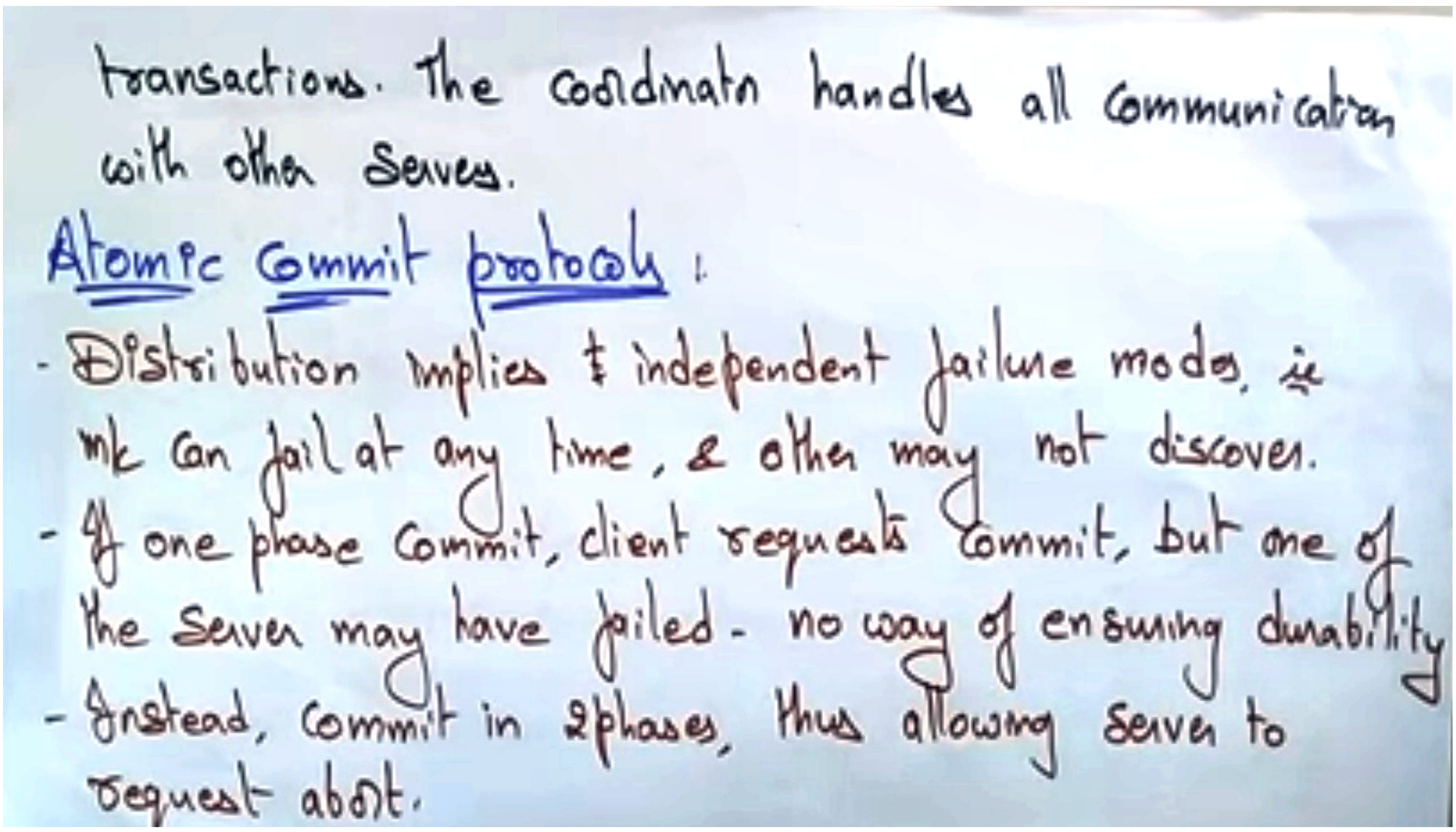


client
x
y
z
client
T
z
x
y
T1
T2
T11
T12
T21
T22
P
Q
R

Simple Distribute Model

Nested Transaction

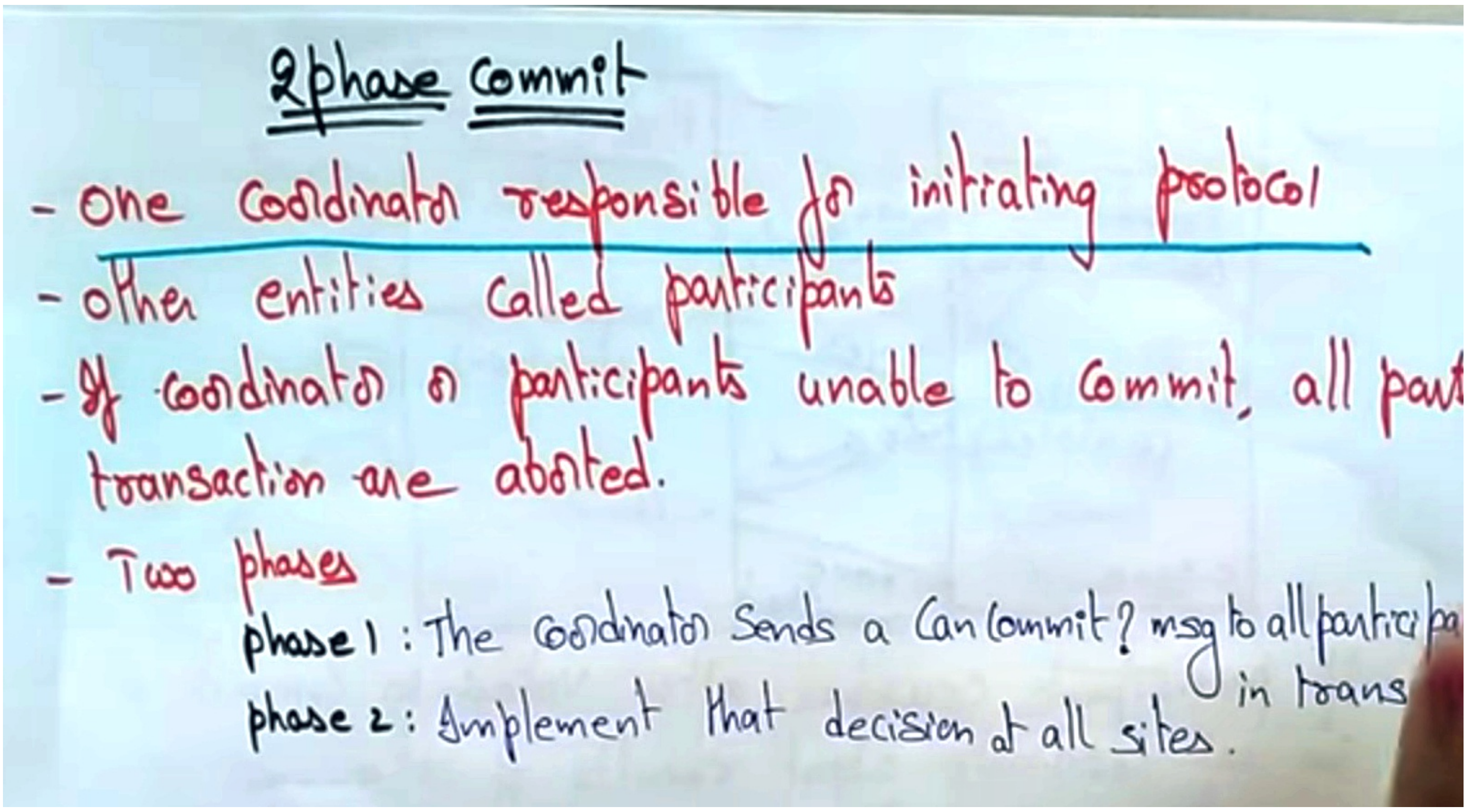
- If client runs transactions then Each transaction must complete before proceeding to next



transactions. The coordinator handles all communication with other servers.

Atomic commit protocols:

- Distribution implies & independent failure modes, ie mk can fail at any time, & other may not discover.
- If one phase commit, client requests commit, but one of the server may have failed - no way of ensuring durability
- Instead, commit in 2 phases, thus allowing server to request abort.

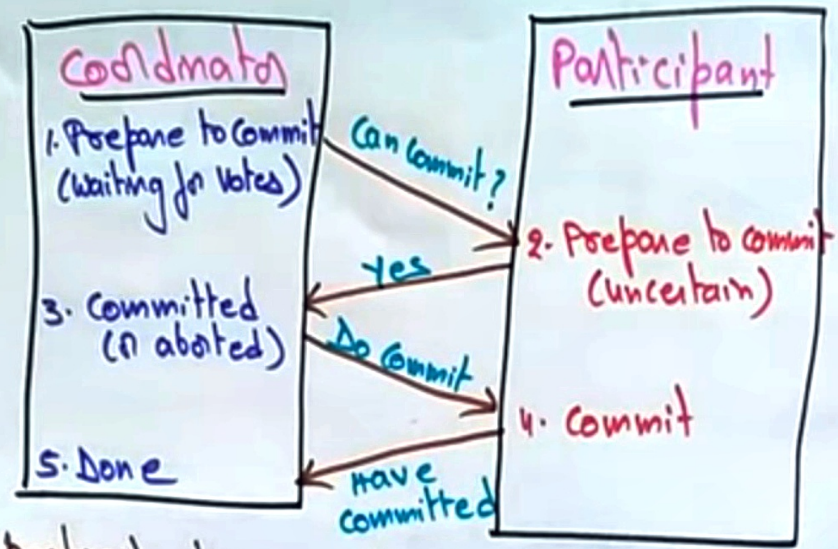


2phase commit

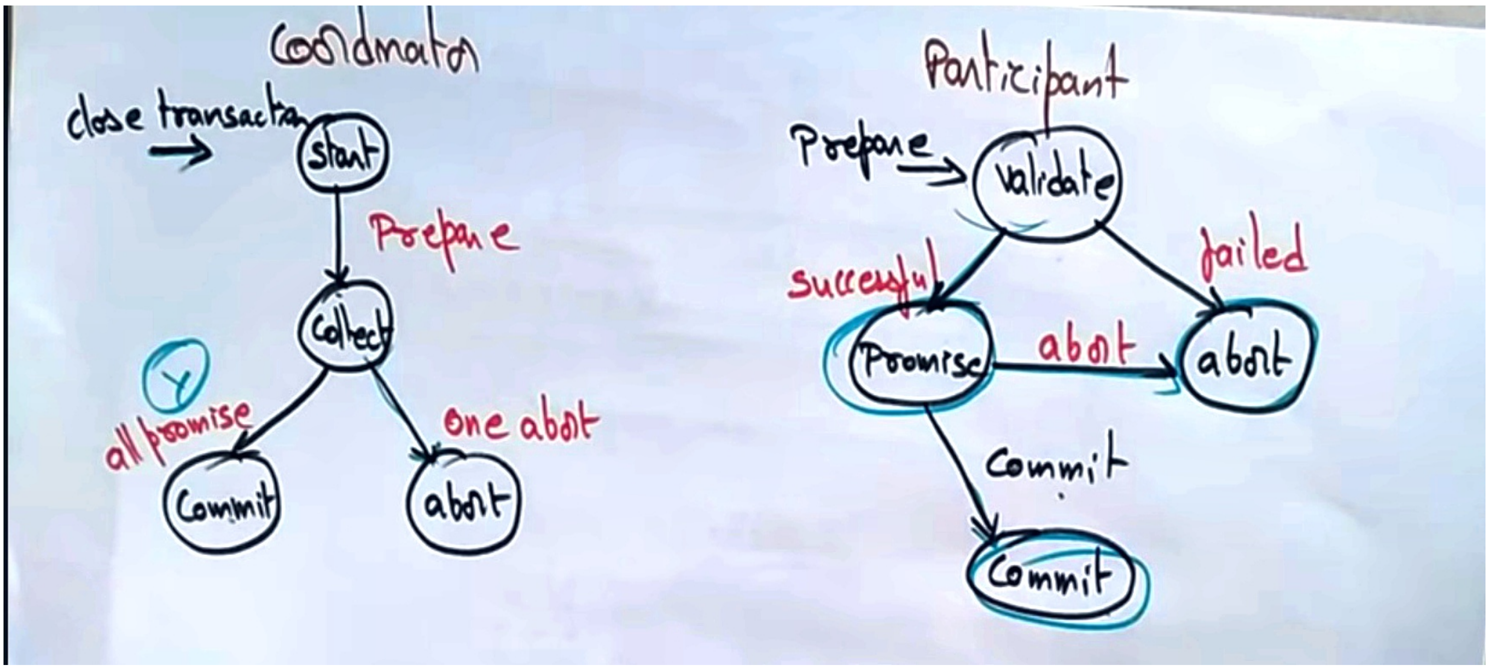
- one coordinator responsible for initiating protocol
- other entities called participants
- If coordinator or participants unable to commit, all part
  transaction are aborted.
- Two phases

  phase 1 : The coordinator sends a Can Commit ? msg to all participa
  in trans

  phase 2 : Implement that decision at all sites.


Coordinator
1. Prepare to commit (waiting for votes)
3. Committed (or aborted)
5. Done
Participant
2. Prepare to commit (uncertain)
4. Commit
Can commit?
Yes
Do commit
Have committed

Note:- If participant crashes after voted to commit, it can ask coordinator about results of vote.
- Timeouts are used when msg are expected
- Introduces new state in transaction prepare to co


Coordinator
close transaction
start
Prepare
collect
all promise
one abort
Commit
abort
Participant
Prepare
validate
successful
failed
Promise
abort
abort
commit
Commit