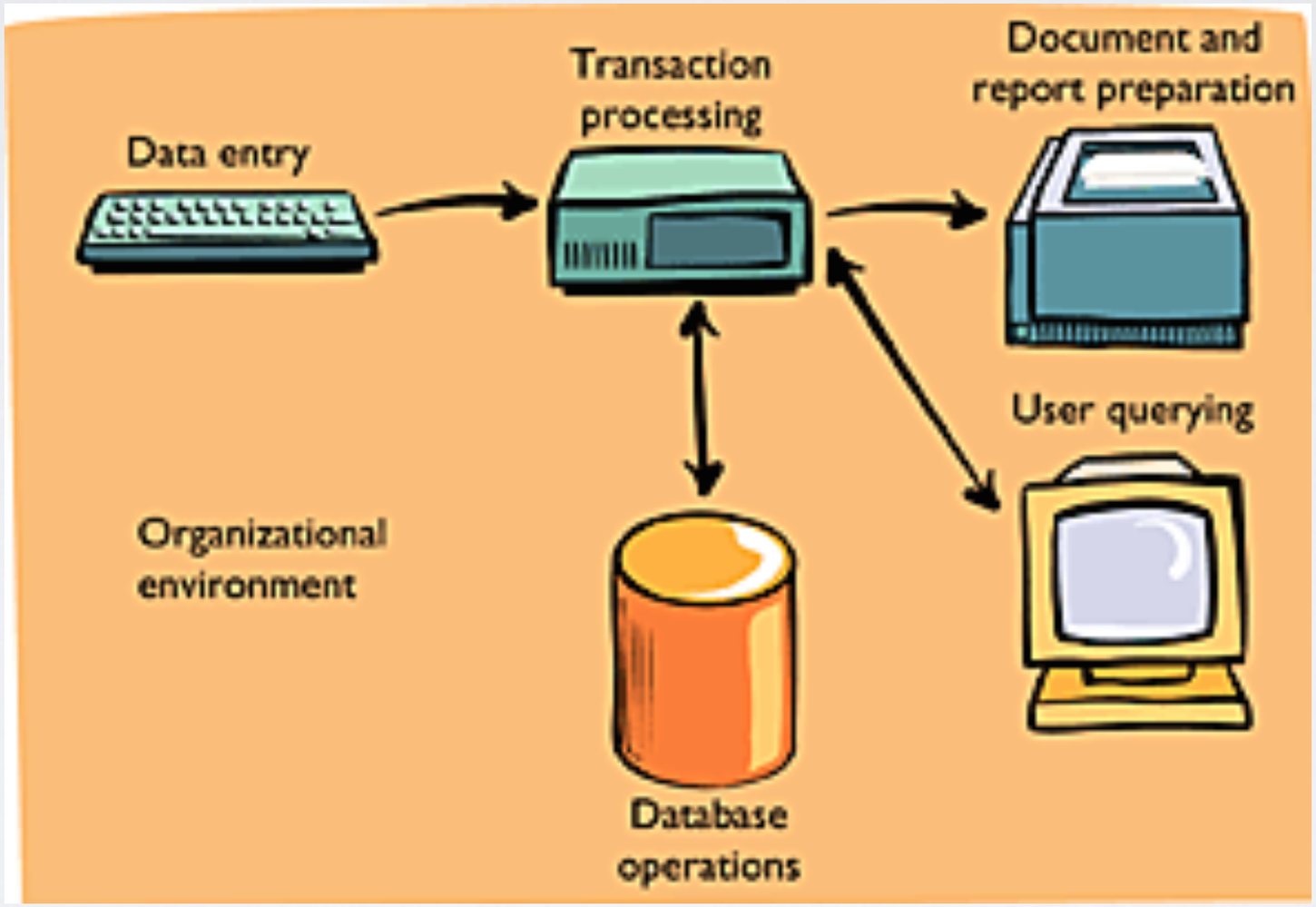
DISTRIBUTED TRANSACTIONS

Network Distributed Systems
Dr. Christina Thorpe

TRANSACTION PROCESSING SYSTEM



DEFINITION

Transaction - a collection of operations on the physical and abstract application state, with the following properties:

- Atomicity.
- Consistency.
- Isolation.
- Durability.

The ACID properties of a transaction.

ATOMICITY

Changes to the state are atomic:

- A jump from the initial state to the result state without any observable intermediate state.
- All or nothing (Commit / Abort) semantics.
- Changes include:
  - Database changes.
  - Messages to outside world.
  - Actions on transducers.

  (testable / untestable)

CONSISTENCY

- The transaction is a correct transformation of the state.
- This means that the transaction is a correct program.

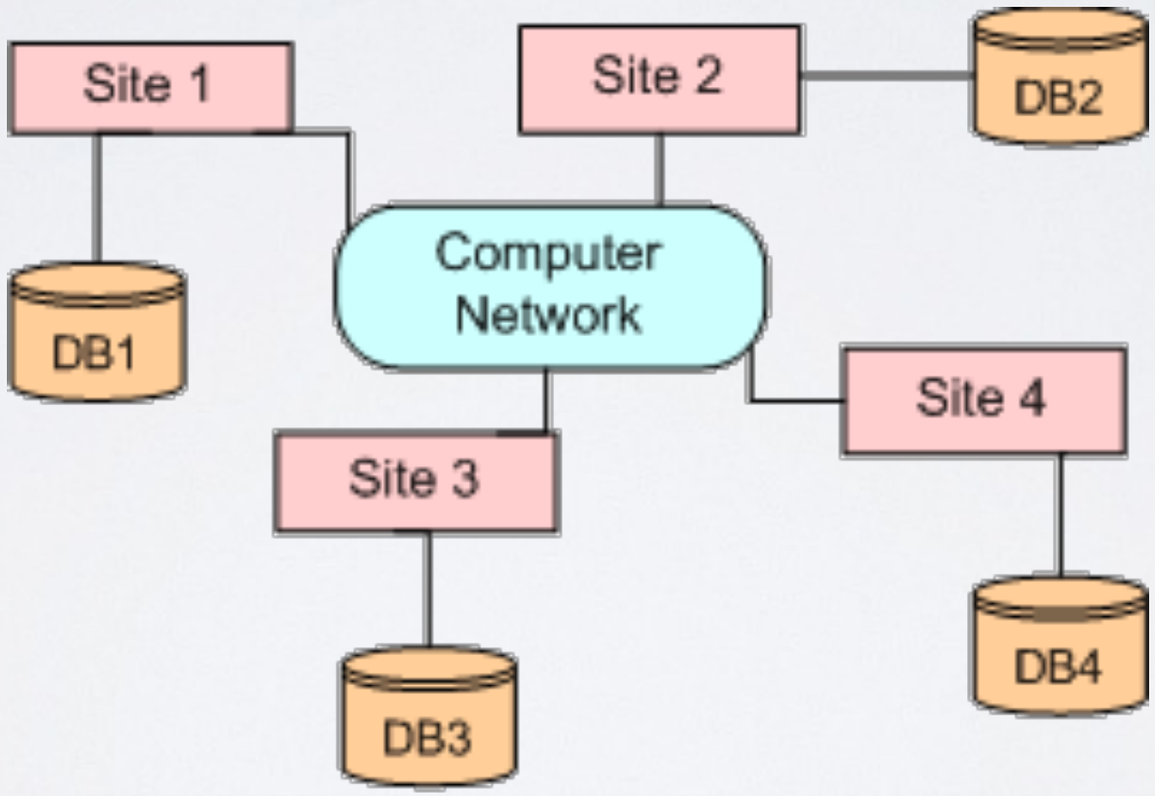
ISOLATION

- Even though transactions execute concurrently, it appears to the **outside observer** as if they execute in some serial order.
- Isolation is required to guarantee consistent input, which is needed for a consistent program to provide consistent output.

DURABILITY

- Once a transaction completes successfully (commits), its changes to the state survive failures (what is the failure model ?).
- The only way to get rid of what a committed transaction has done is to execute a compensating transaction (which is, sometimes, impossible).

A DISTRIBUTED DATABASE



A DISTRIBUTED TRANSACTION

- A distributed transaction is composed of several sub-transactions, each running on a different site.

- Each database manager (DM) can decide to abort (the veto property).

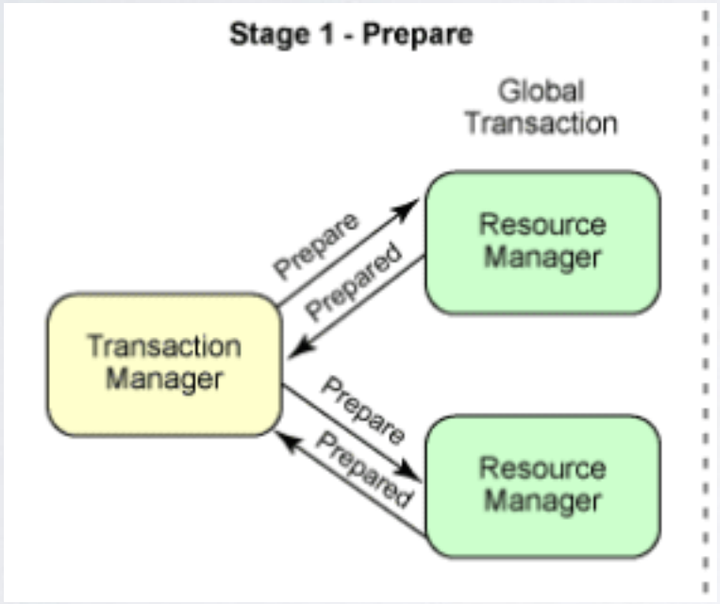
- An Atomic Commitment Protocol (ACP) is run by each of the DMs to ensure that all the sub-transactions are consistently committed or aborted.

ATOMIC COMMITMENT PROTOCOL

A correct ACP guarantees that:

- All the DM that reach a decision, reach the same decision.
- Decisions are not reversible.
- A Commit decision can only be reached if all the DMs voted to commit.
- If there are no failures and all the DMs voted to commit, the decision will be Commit.
- At any point, if all failures are repaired, and no new failures are introduced, then all the DMs eventually reach a decision.

TWO PHASE COMMIT

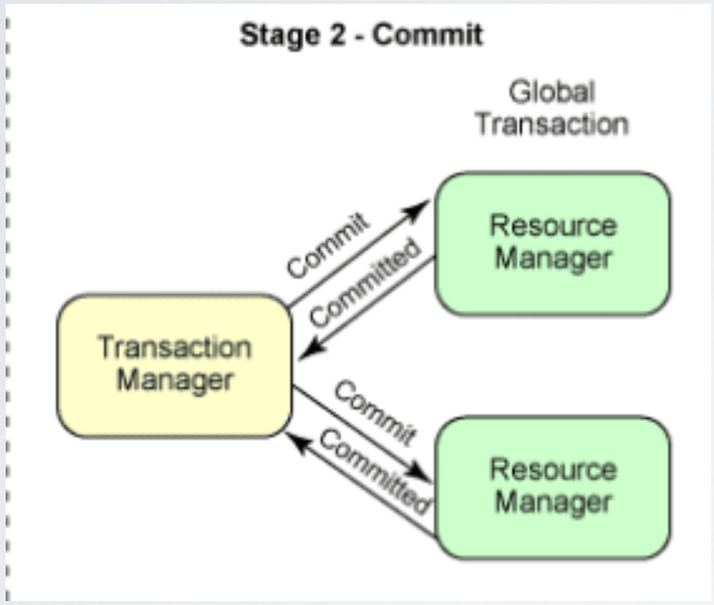


1. The transaction manager asks all the resource managers to prepare to commit recoverable resources (prepare).

2. Each resource manager can vote either positively (prepared) or negatively (rolled-back). If a resource manager is to reply positively, it records stably the information it needs to do so, replies prepared, and is then obliged to follow the eventual outcome of the transaction, as determined at the next stage.

3. The resource manager is now described as in-doubt, since it has delegated the eventual outcome of the transaction to the transaction manager and is now in-doubt about the actual outcome of the transaction.

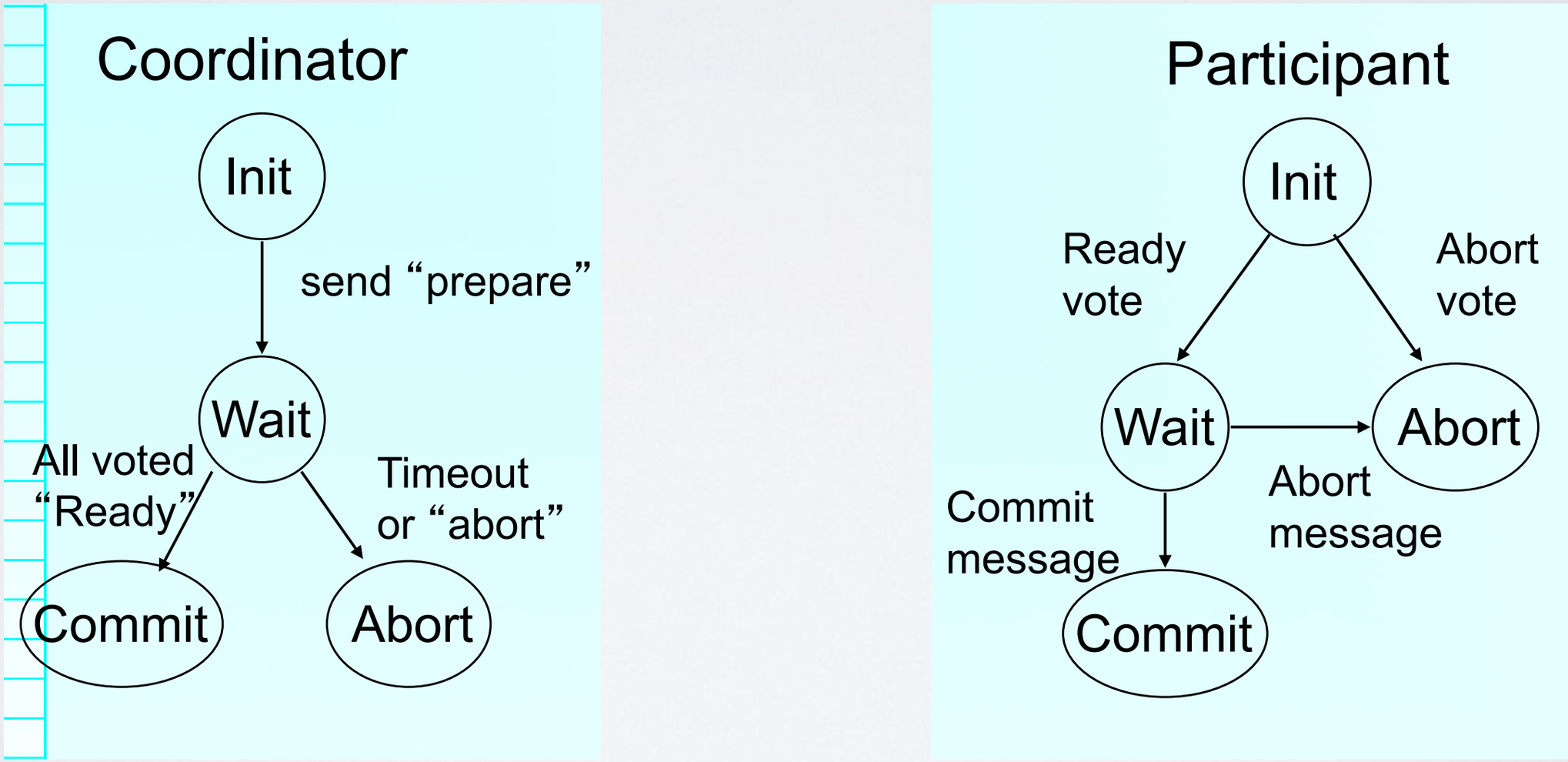
TWO PHASE COMMIT



Assuming all the resource managers replied positively:

1. The transaction manager replies to each resource manager with a commit flow. However, if a resource manager fails to reply, the transaction manager may re-transmit the prepare flow before assuming the transaction should be aborted.
2. Upon receipt of the commit flow, the resource manager finalizes the updates to recoverable resources, and releases any locks held on the resources or open files.
3. The resource manager then responds with a final committed flow, which indicates to the transaction manager that it is no longer in-doubt.
4. If the final committed flow is not received by the transaction manager, the transaction manager must assume the commit did not arrive at the resource manager, and so would need to re-transmit the commit, until a positive reply is received.

2PC STATE DIAGRAM



NON BLOCKING APCS

- An ACP is called blocking if the occurrence of some failures forces the DMs to wait until failures are repaired before terminating the transaction.
- When a transaction is blocked at the DM, its locks cannot be released. This may lead to system blocking.
- What can we say about network partitions and blocking?

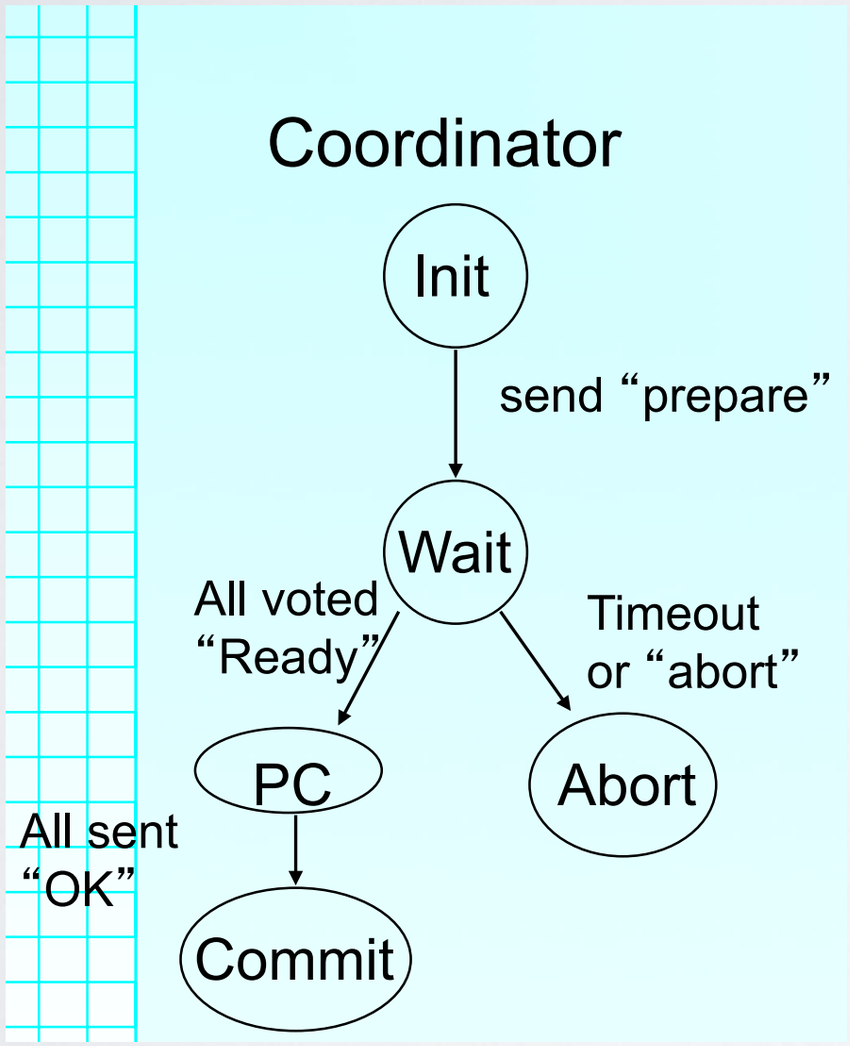
NON BLOCKING APCS

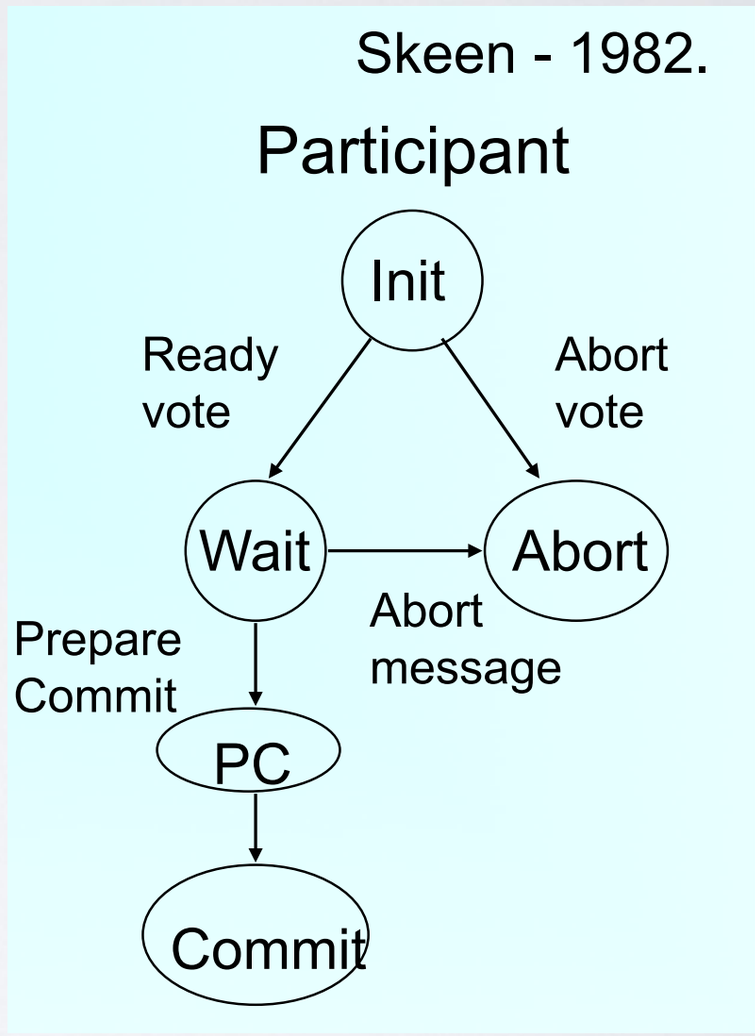
- An ACP is called blocking if the occurrence of some failures forces the DMs to wait until failures are repaired before terminating the transaction.
- When a transaction is blocked at the DM, its locks cannot be released. This may lead to system blocking.
- Every protocol that tolerates network partitions is bound to be blocking.

QUORUM BASED PROTOCOL

- Every DM has to agree locally.
- A majority of the DMs must agree to abort or commit after all the DMs agreed locally.
- Simple majority can be generalized to weighted majority.
- Majority can be generalized to quorum.
- Instead of one quorum, there can be an abort quorum and a commit quorum.

3PC STATE DIAGRAM (FAULTS)





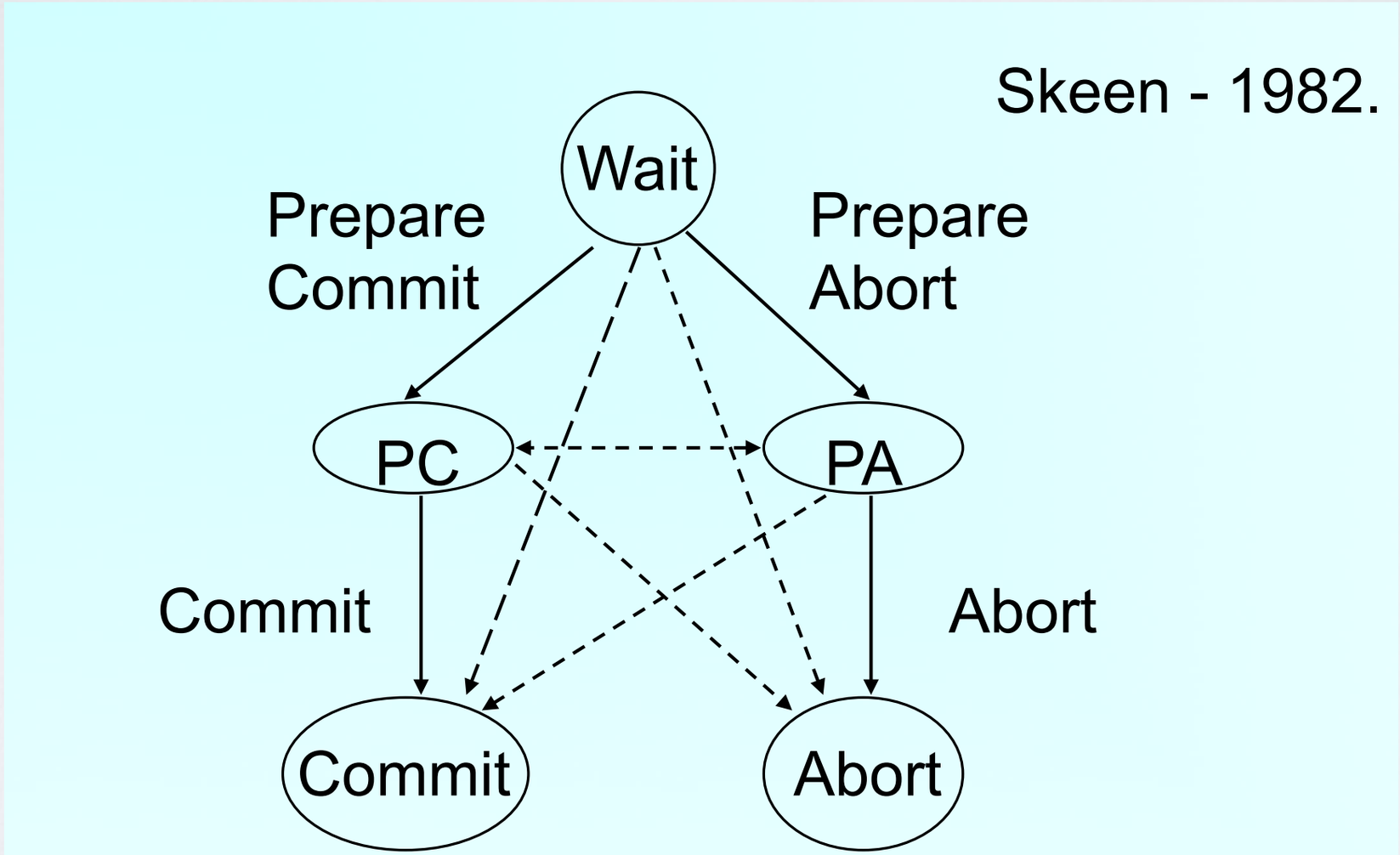
3PC DECISION RULE FOR RECOVERY

- Collected States:
  - If at least one DM aborted - decide to abort.
  - If at least one DM committed - decide to commit.
  - Otherwise if at least one DM in Pre-Commit and a quorum of DMs in (Pre-Commit and Wait) - move to Pre-Commit and send "prepare commit".
  - Otherwise if there is a quorum of DMs in (Waitand Pre-Abort) move to Pre-Abort and send "prepare abort".
  - Otherwise-Block.

3PC RECOVERY PROCEDURE

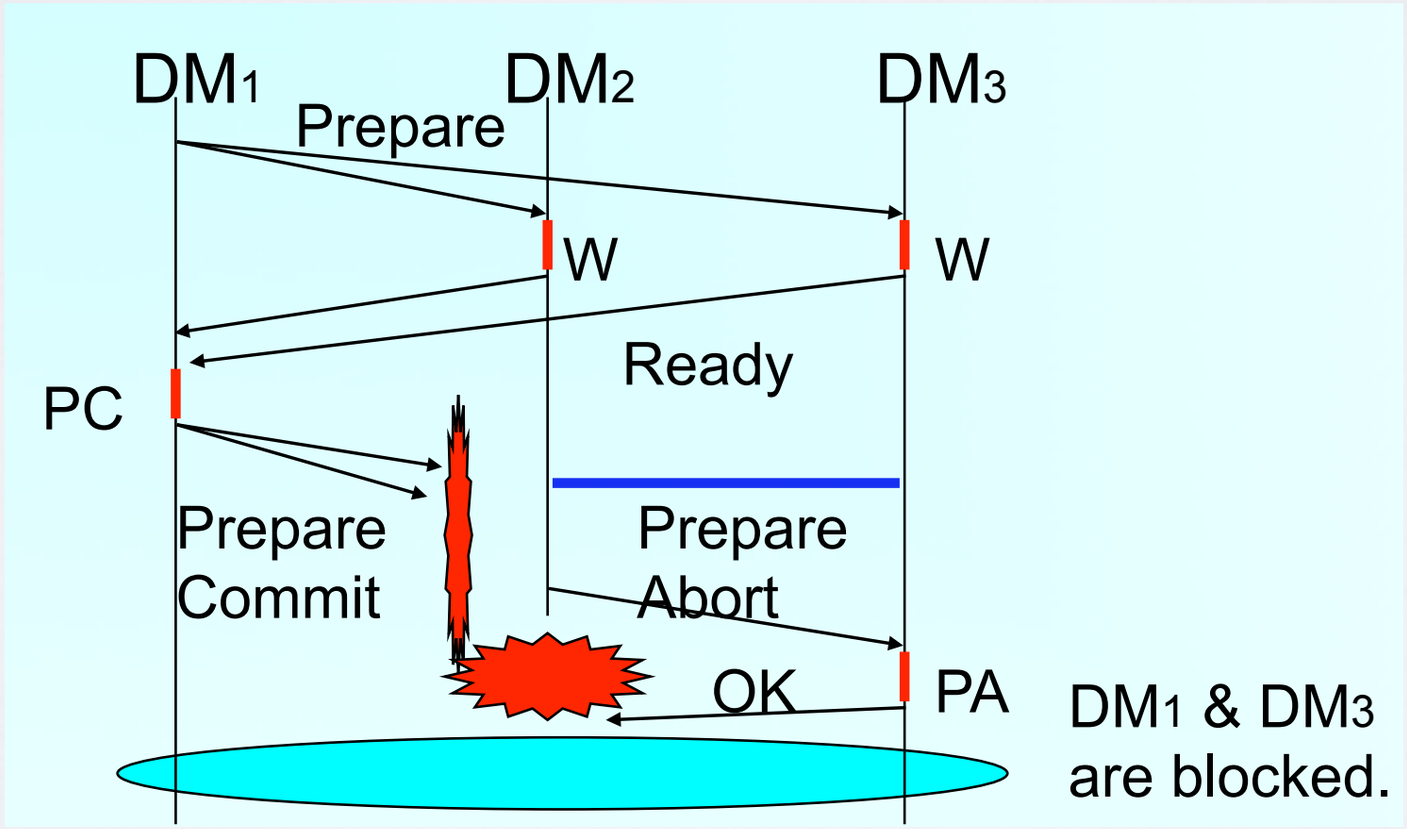
- Send state and id.
- The new coordinator collects the states from all the connected DMs, it computes its next step according to the decision rule.
- Upon receiving a Prepare-Commit/Prepare- abort, each DM sends an OK message.
- Upon receiving an OK message from a quorum, the coordinator commits/aborts and sends the decision.

3PC RECOVERY STATE DIAGRAM



3PC CAN BLOCK A QUORUM

Simple majority, 3 DMs, smallest connected DM is the coordinator.



ENHANCED 3PC HIGHLIGHTS

- Uses identical state diagrams as 3PC.
- Uses similar communication to 3PC (with different message contents).
- Maintains two additional counters:
  - Last_elected- the index of the last election this DM participated in.
  - Last_attempt - the election number in the last attempt this DM made to commit or abort.
    - Uses a different decision rule and recover procedure.

E3PC DECISION RULES

- IMAC : a predicate that is true iff all the connected members with max Last_attempt are in the PC state.
  - If at least one DM aborted - decide abort.
  - If at least one DM committed - decide commit.
  - If IMAC and there is a quorum - move to Prepare-Commit.
  - If not IMAC and there is a quorum - move to Prepare-Abort.
  - Otherwise (i.e. no quorum) - Block

E3PC RECOVERY PROCEDURE

- Elect a coordinator - send state and 2 counters.
- upon getting the Max_elected from the coordinator, set Last_elected = Max_elected +1.
- If the coordinator decision is not to block
    - It sets Last_attempt = Last_elected.
    - move to the calculated state and multicast decision.
- Upon receiving Prepare-Commit/Prepare-Abort, the DM:
    - Sets Last_attempt = Last_elected.
    - Changes state to PC or PA and sends OK.
- If a fault happens - restart the recovery procedure, otherwise termination is guaranteed.

E3PC NEVER BLOCKS A QUORUM

Simple majority, 3 DMs, smallest connected DM is the coordinator.