EasyCommit: A Non-blocking Two-phase Commit Protocol

Presentation for ECS 265: Distributed Database Systems

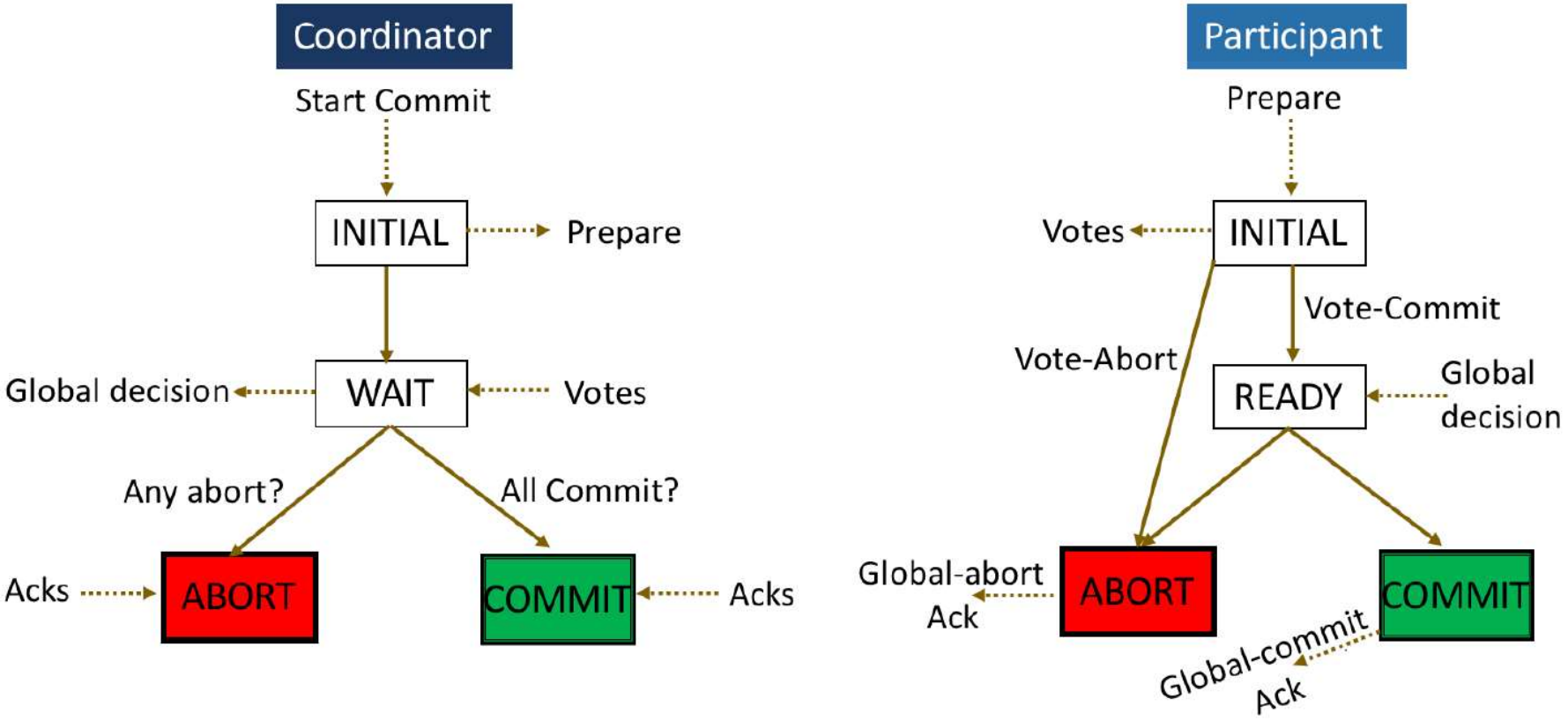
Presented by Likang Yin

October 4, 2018

Roadmap

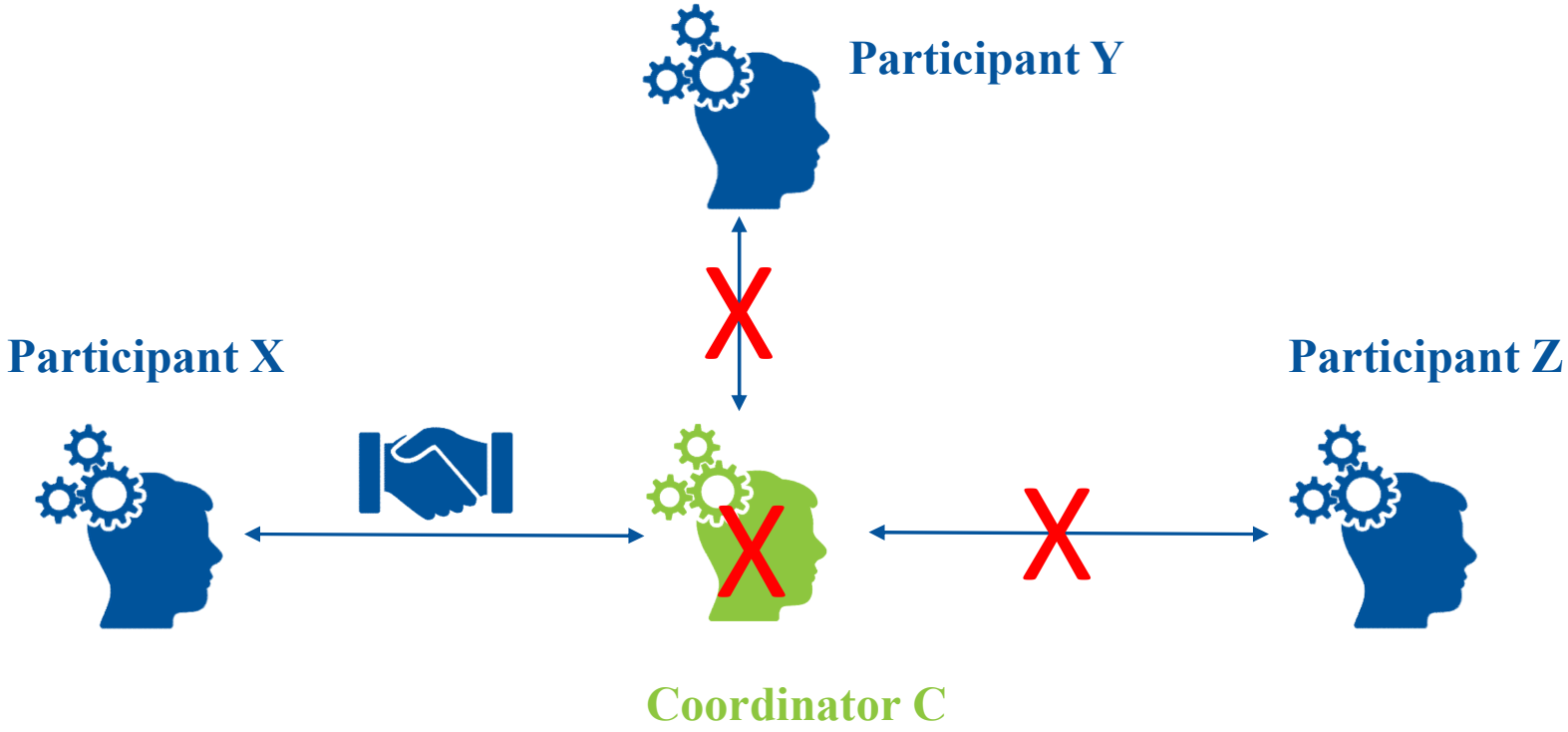
- 1. Background
- 2. Two-phase Commit (2PC) Protocol
- 3. Three-phase Commit (3PC) Protocol
- 4. EasyCommit Protocol
- 5. Comparison and Analysis

Two-phase Commit (2PC) Protocol

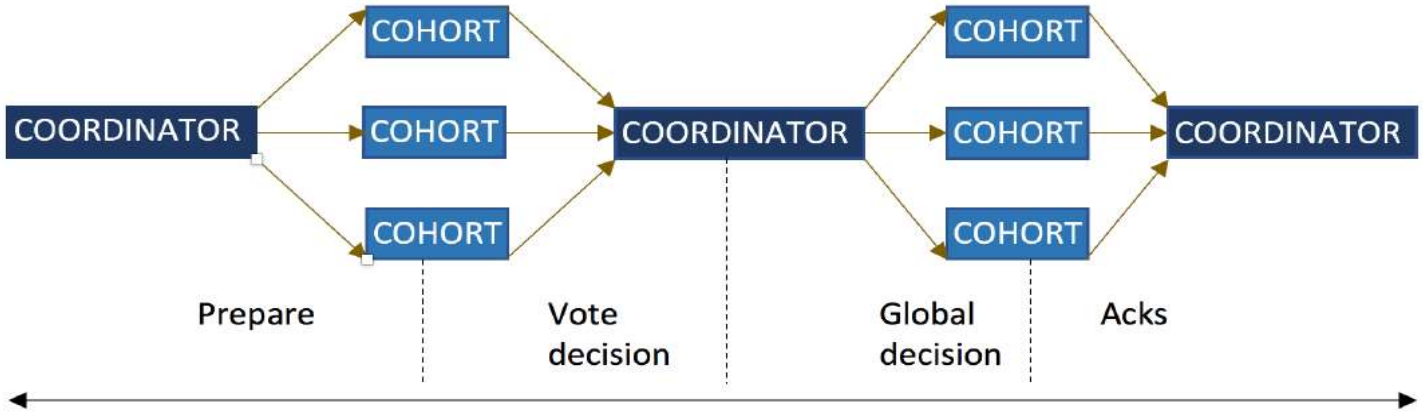


Source: EasyCommit: A Non-blocking Two-phase Commit Protocol. EDBT'18.

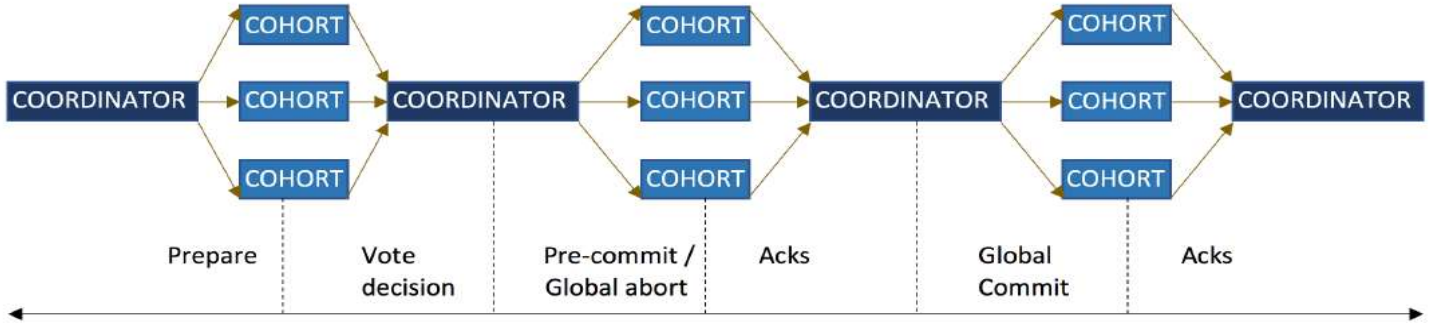
Blocking Problem in 2PC Protocol



Three-phase Commit (3PC) Protocol



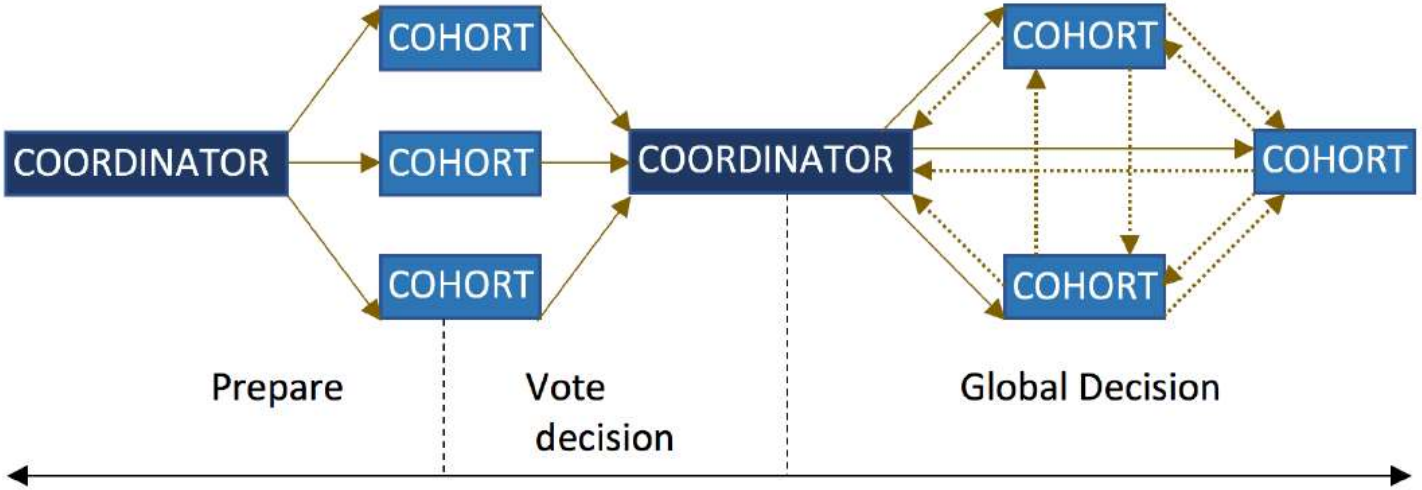
(a) Time span of 2PC Protocol



(b) Time span of 3PC Protocol

Source: EasyCommit: A Non-blocking Two-phase Commit Protocol. EDBT'18.

EasyCommit Protocol



(c) Time span of EC Protocol

Source: EasyCommit: A Non-blocking Two-phase Commit Protocol. EDBT'18.

Termination Protocol

Three different situations:

1. **Coordinator Timeout in WAIT State**
2. **Cohort Timeout in INITIAL State**
3. **Cohort Timeout in READY State**

Independent Recovery

EasyCommit Protocol can do the independent recovery when:

1. **If a cohort fails before transmitting its vote, then on recovery, it can simply abort the transaction.**

2. **If the coordinator fails before transmitting the global decision, then it aborts the transaction on recovery.**

3. **If either coordinator or participant fail after transmitting the global decision and writing the log, then on recovery they can use this entry to reach the consistent state.**

Conclusion

The easy commit protocol takes the advantages of both 2PC Protocol and 3PC Protocol. It is able to:

1. **Handle commit transactions efficiently (Two phases).**

2. **Resolve the transactions safely if the coordinator fails (Non-blocking).**

3. **Execute independent recovery in three different cases. (Independent recovery).**