NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications
Prof. Sandip Chakraborty

Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 29: Paxos – Safety and Liveness

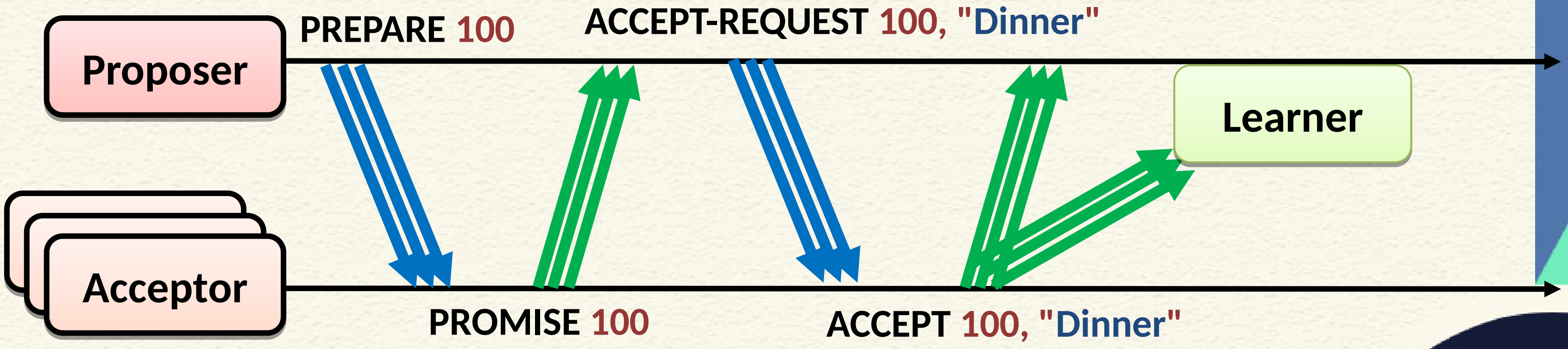
CONCEPTS COVERED

- **Safety and Liveness of Paxos**

KEYWORDS

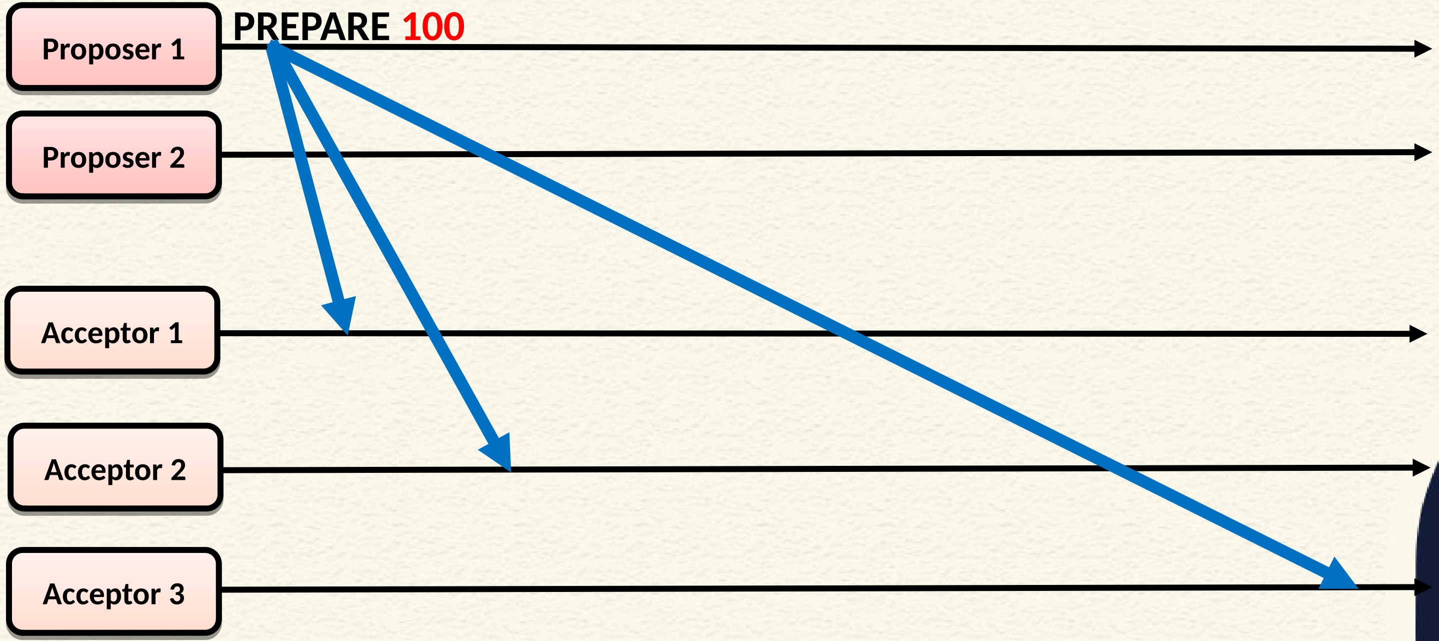
- **Paxos: Correctness**
- **Leader Election**
- **Multi-Paxos**

Paxos – Message Exchanges

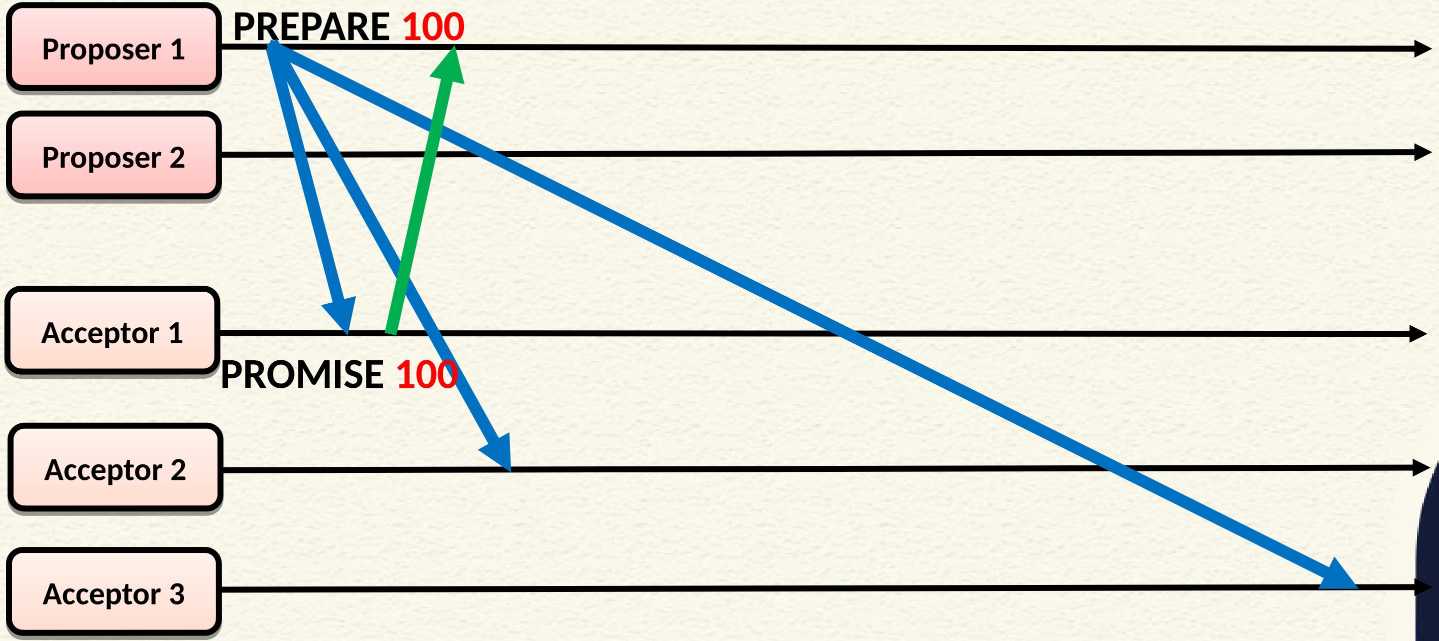


- Two rounds of message exchanges
  - PREPARE – PROMISE: Agree on a state (ID)
  - ACCEPT-REQUEST – ACCEPT: Agree on a value
- The consensus is on the "***value***"

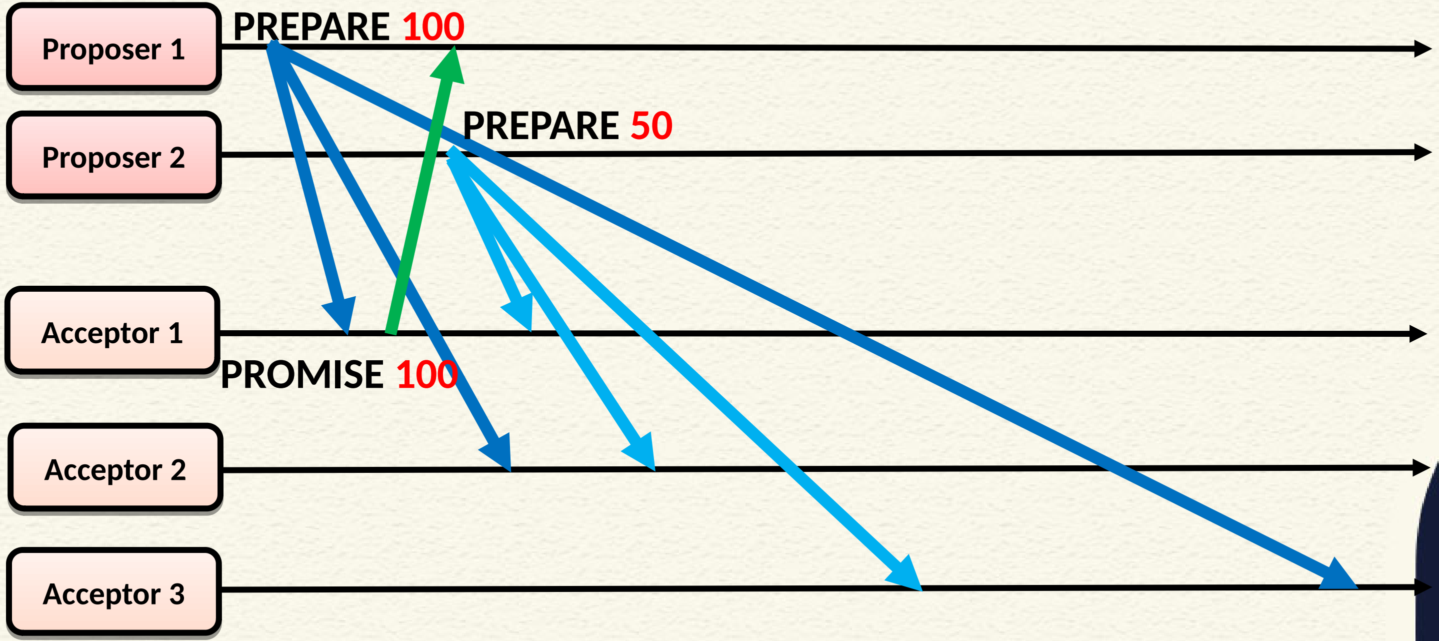
Majority Voting



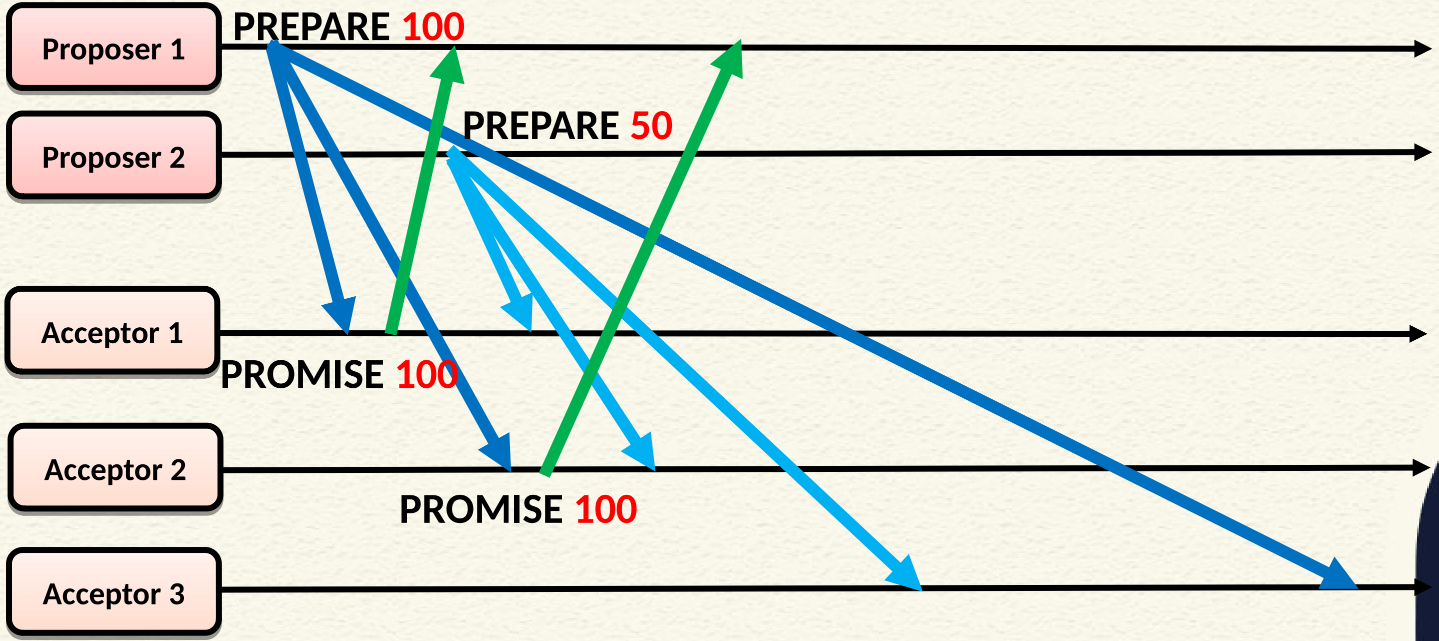
Majority Voting



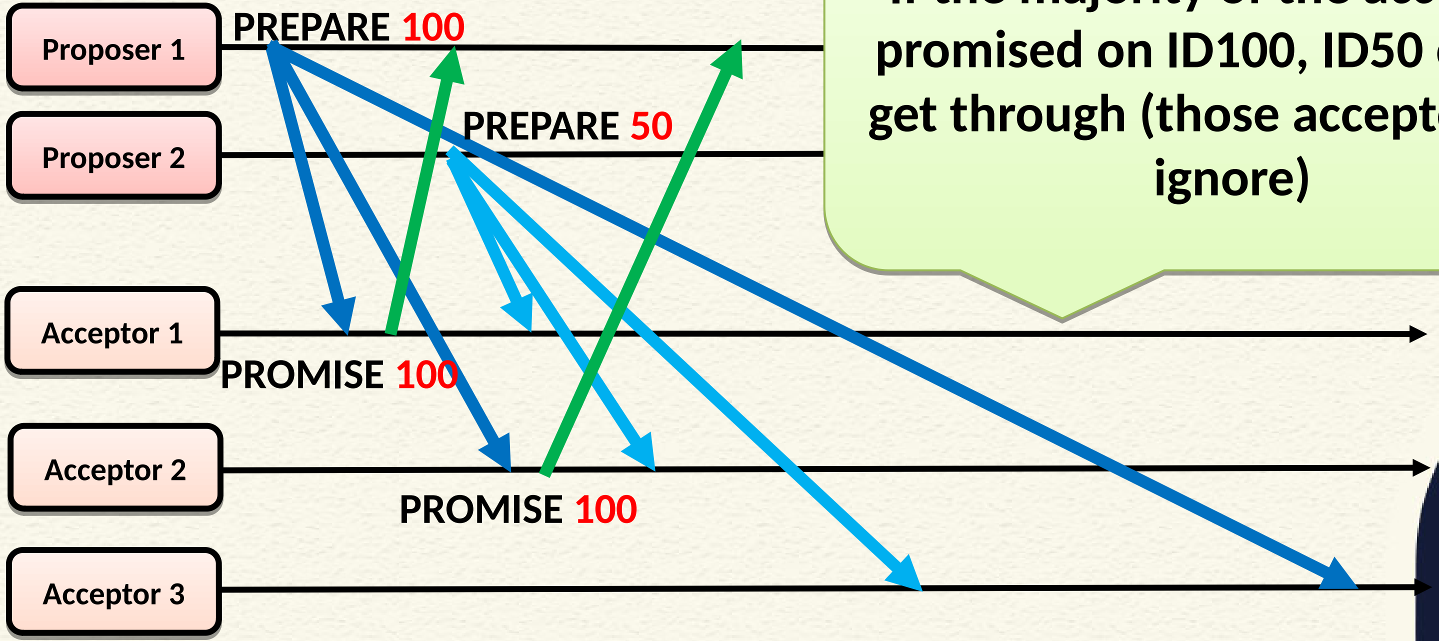
Majority Voting



Majority Voting

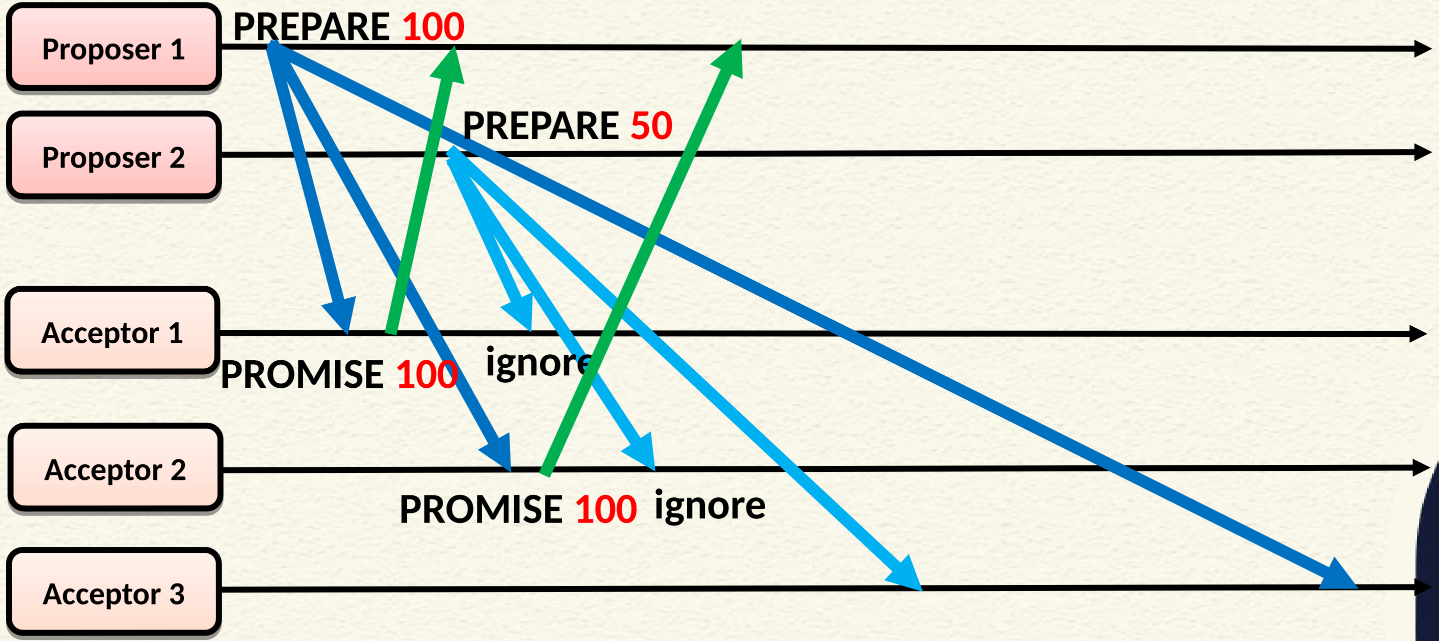


Majority Voting

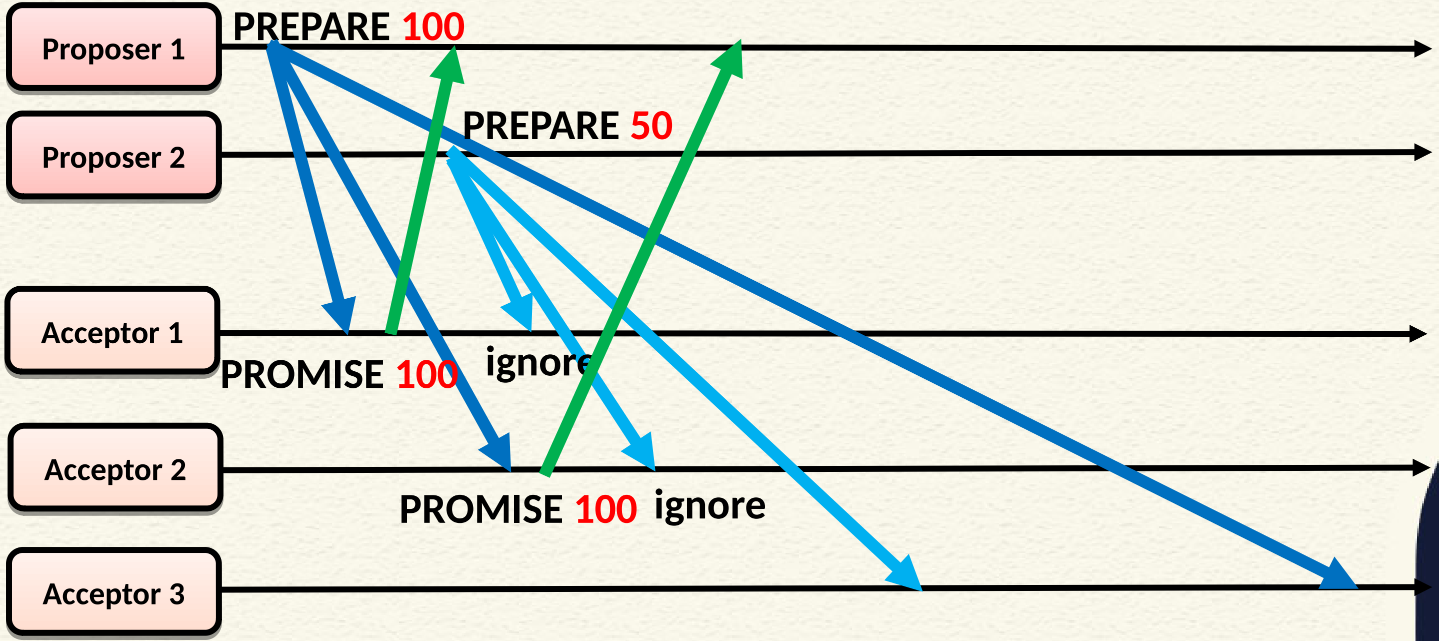


If the majority of the acceptors promised on ID100, ID50 cannot get through (those acceptors will ignore)

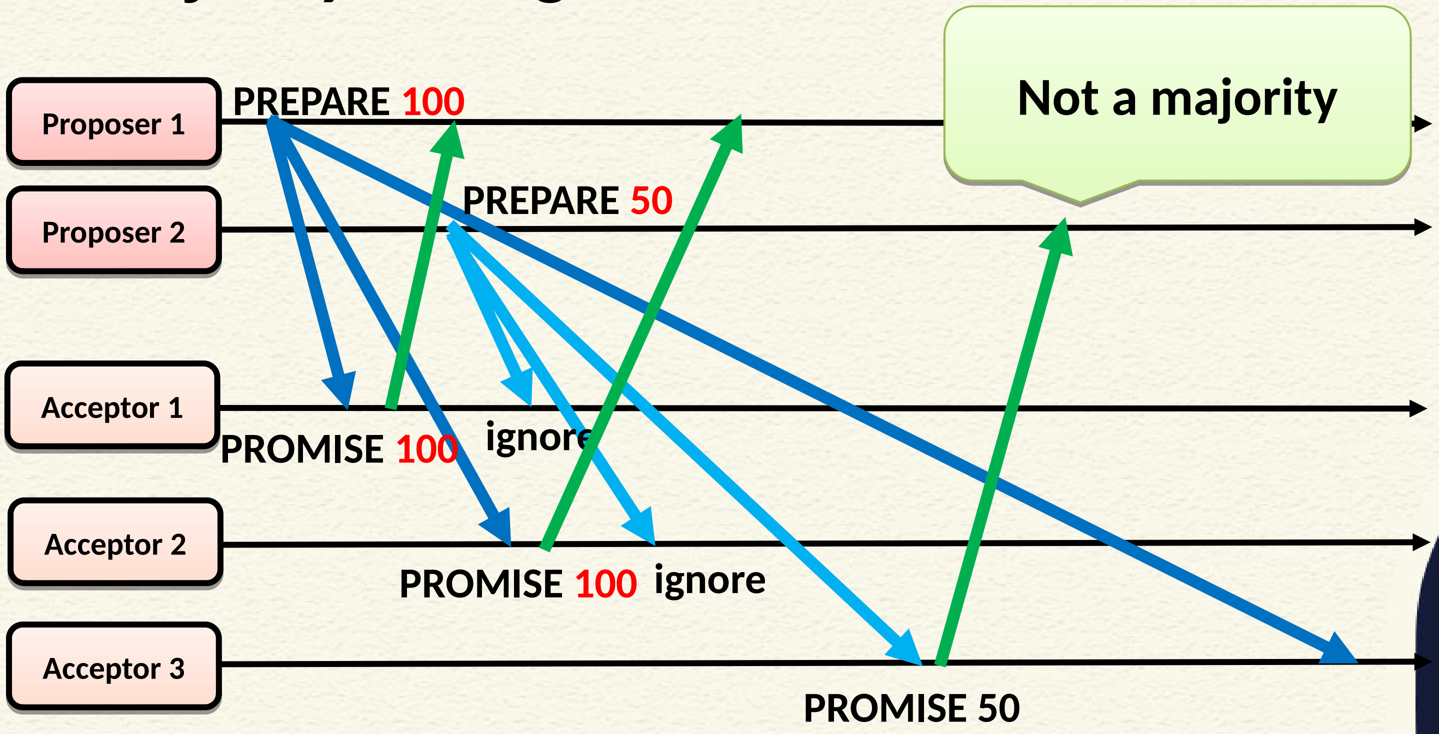
Majority Voting



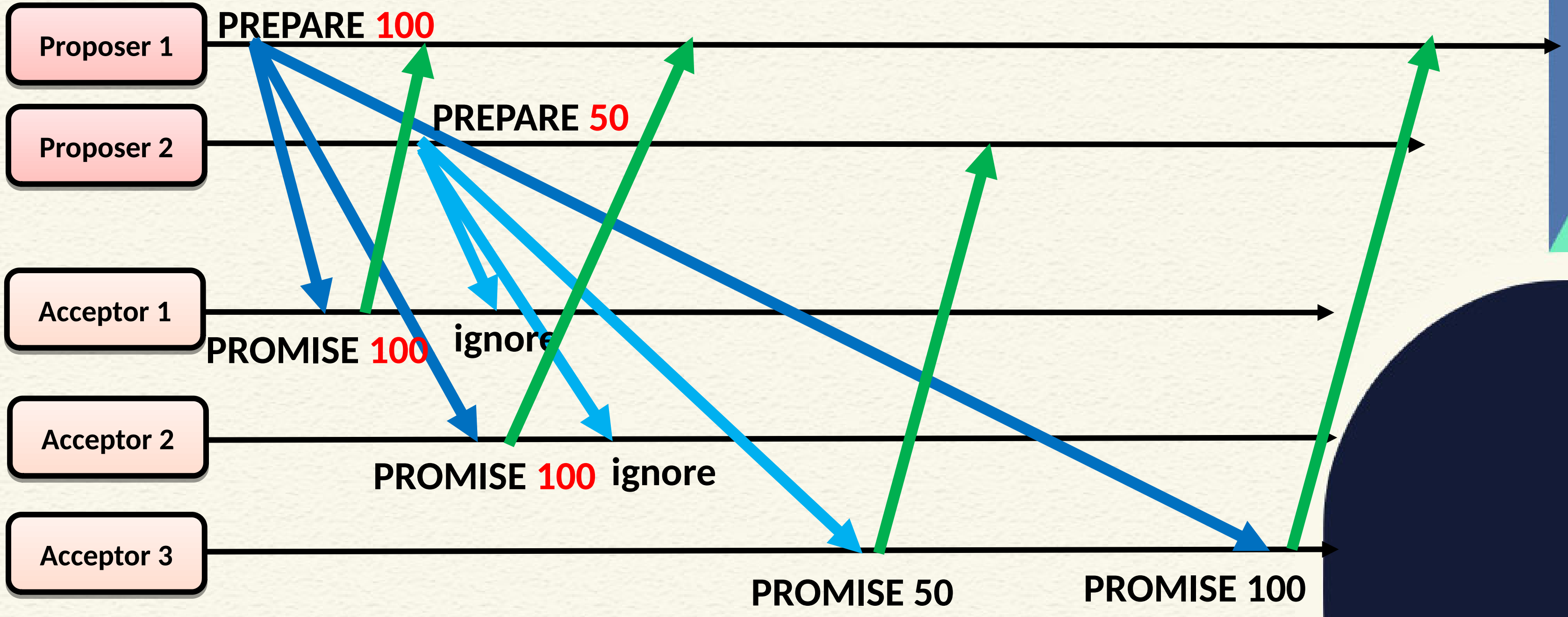
Majority Voting



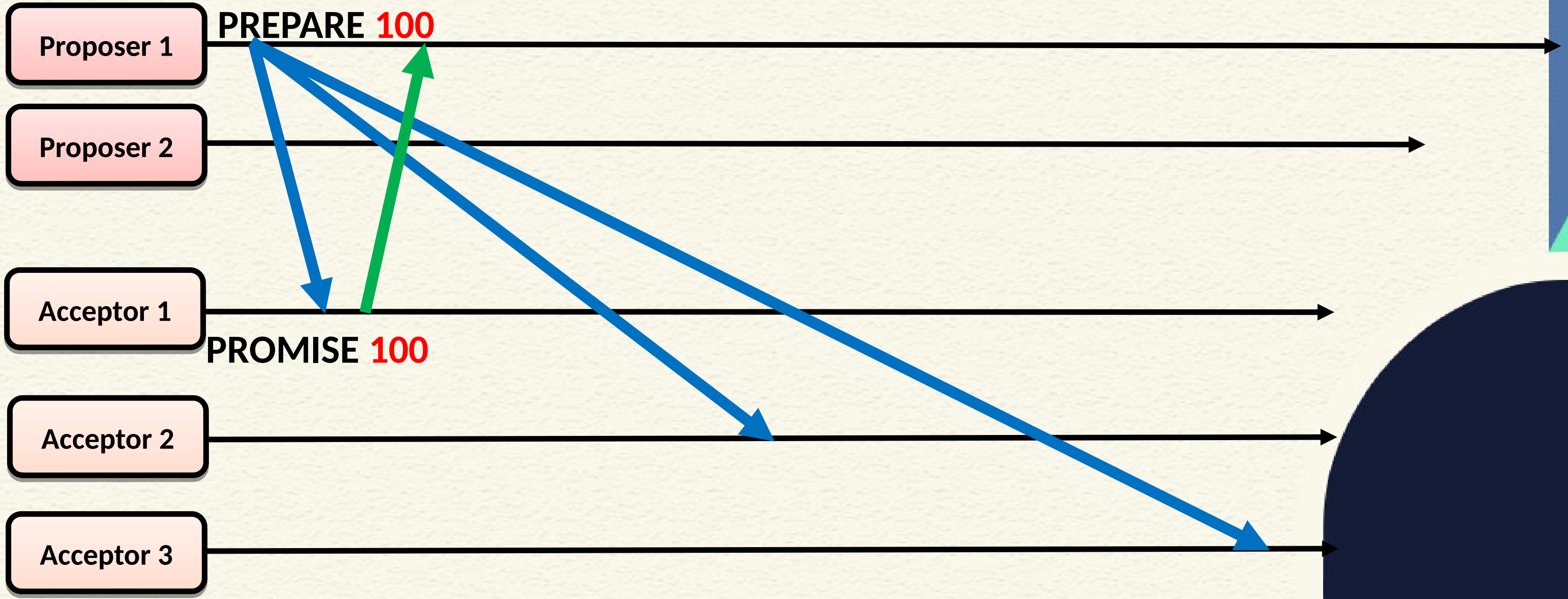
Majority Voting



Majority Voting



Majority Voting – Case 2


Proposer 1

Proposer 2

Acceptor 1

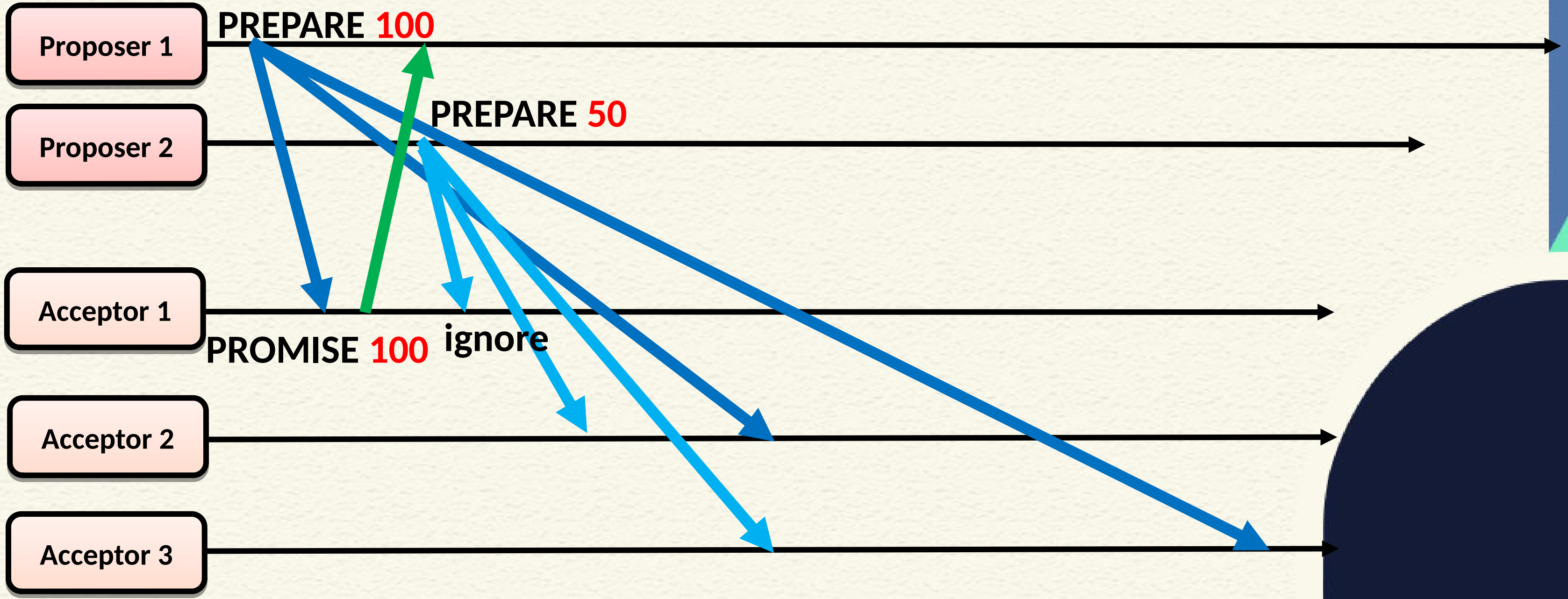
Acceptor 2

Acceptor 3

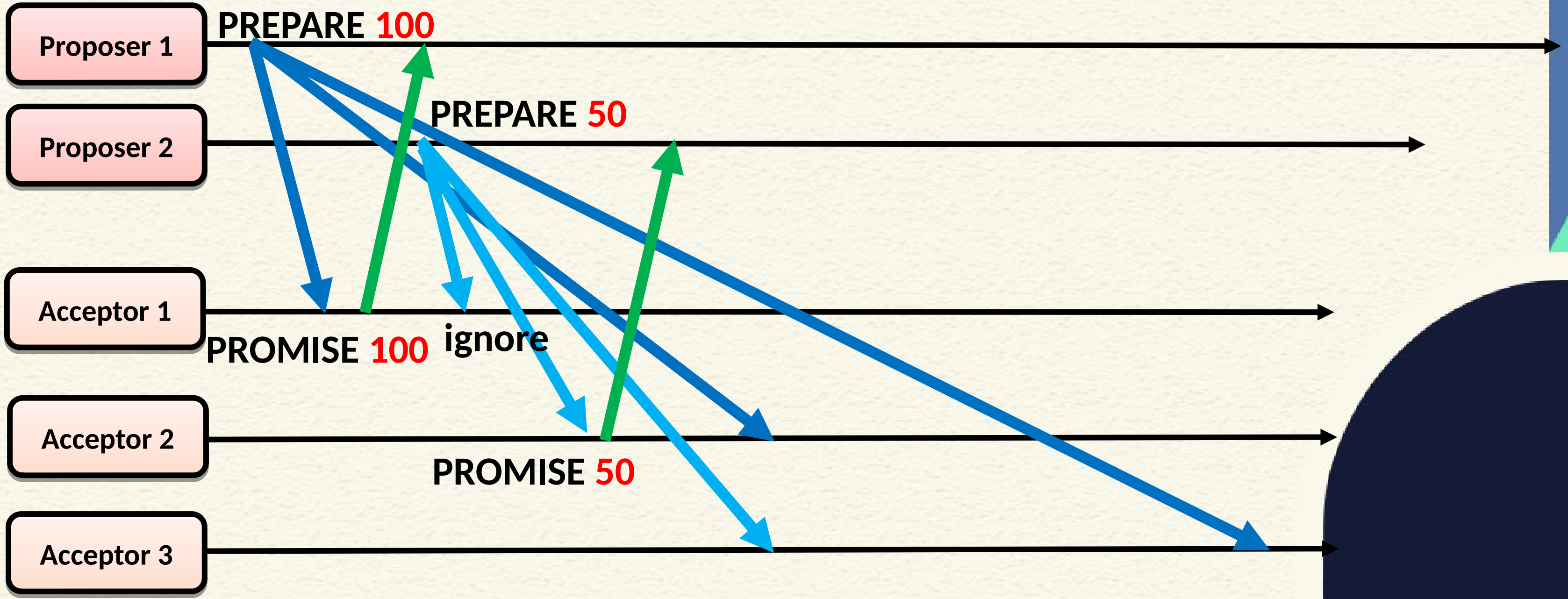
PREPARE 100

PROMISE 100

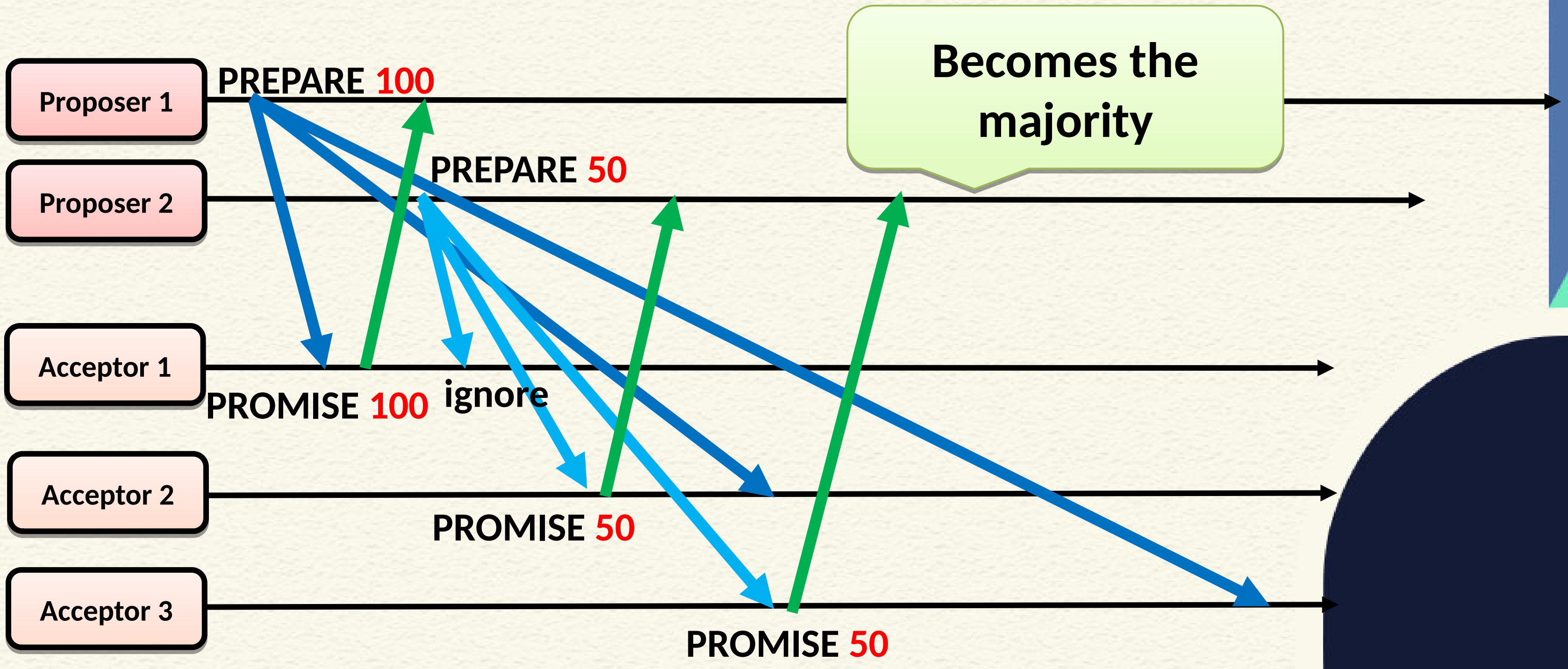
Majority Voting – Case 2



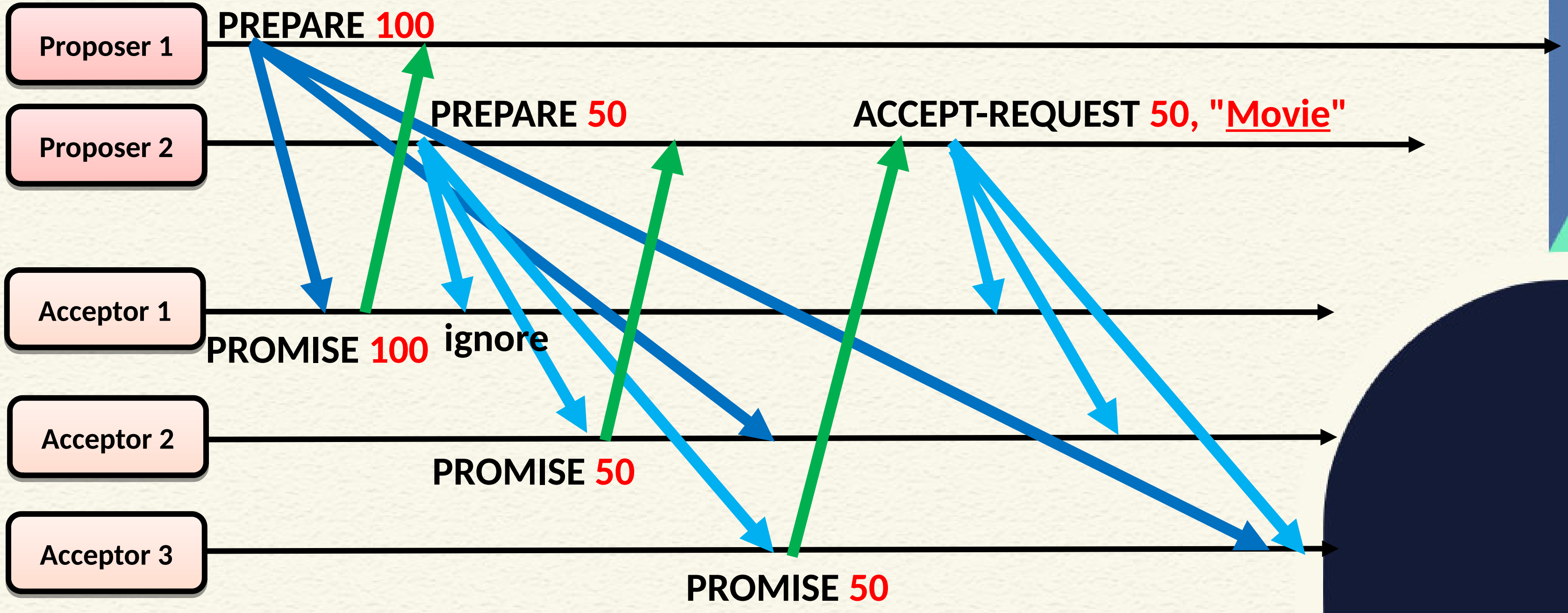
Majority Voting – Case 2



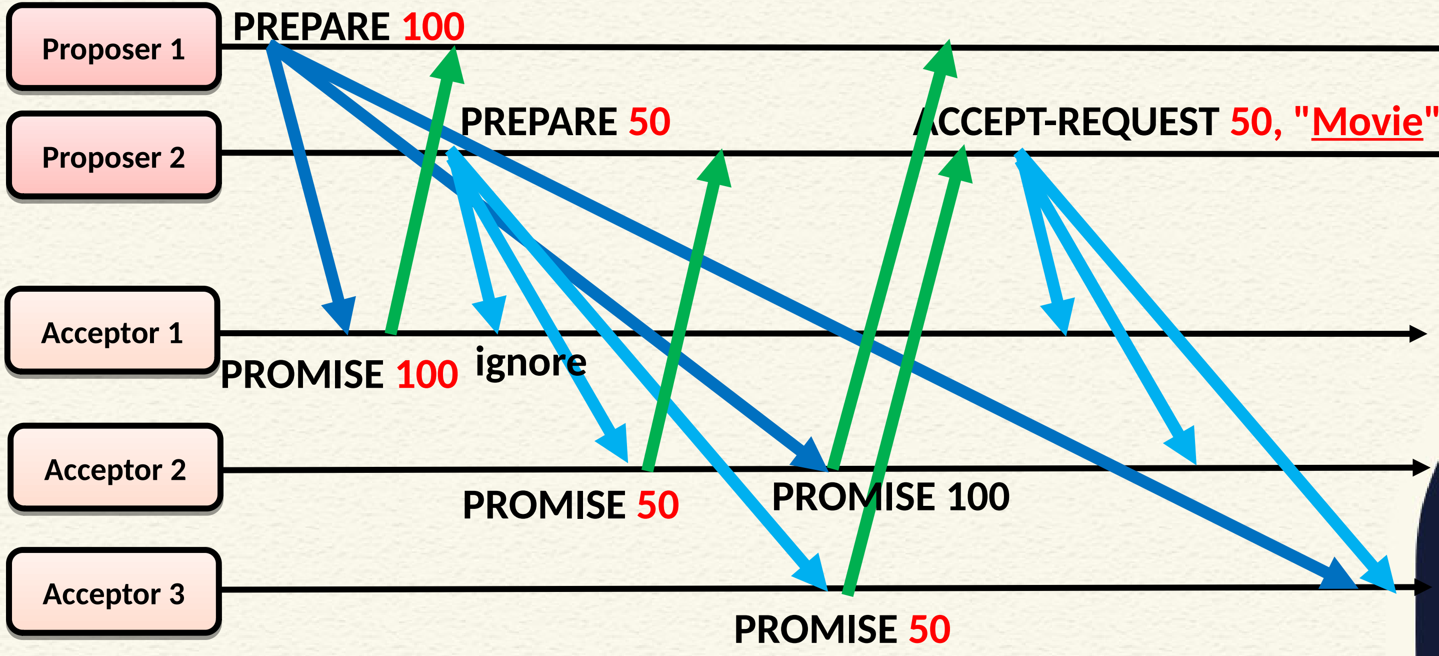
Majority Voting – Case 2



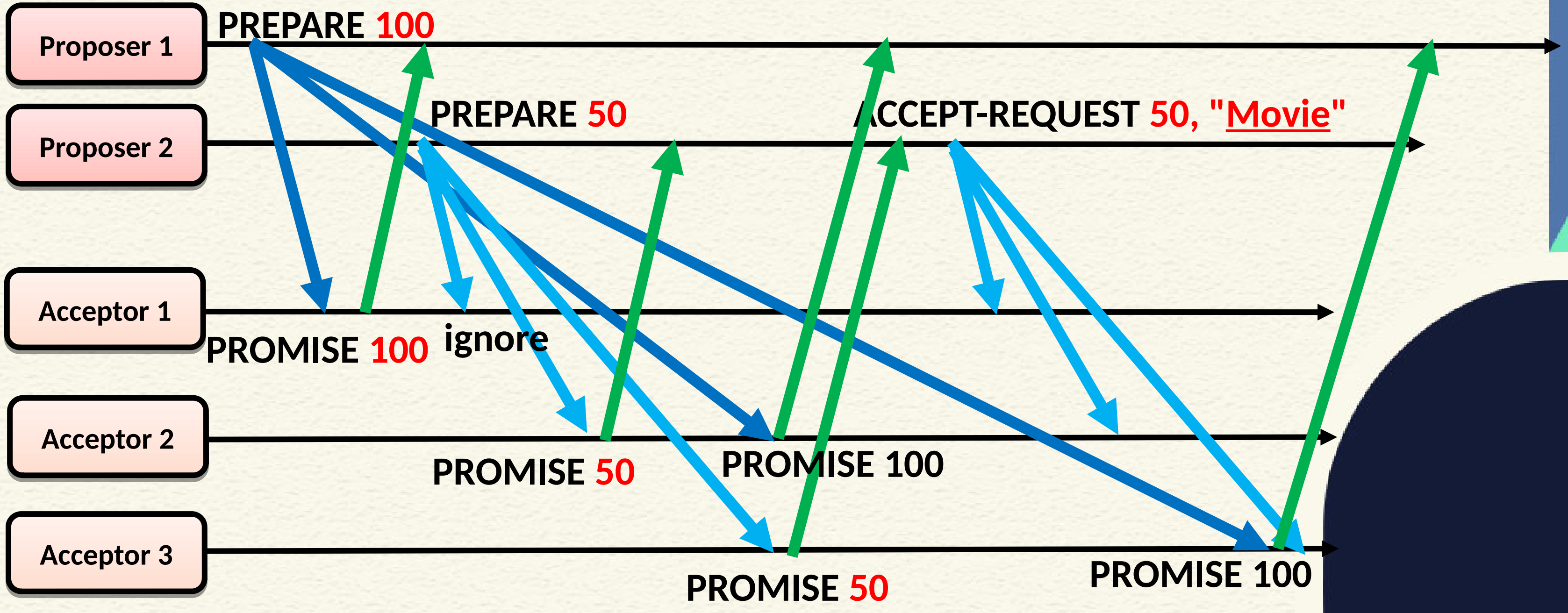
Majority Voting – Case 2



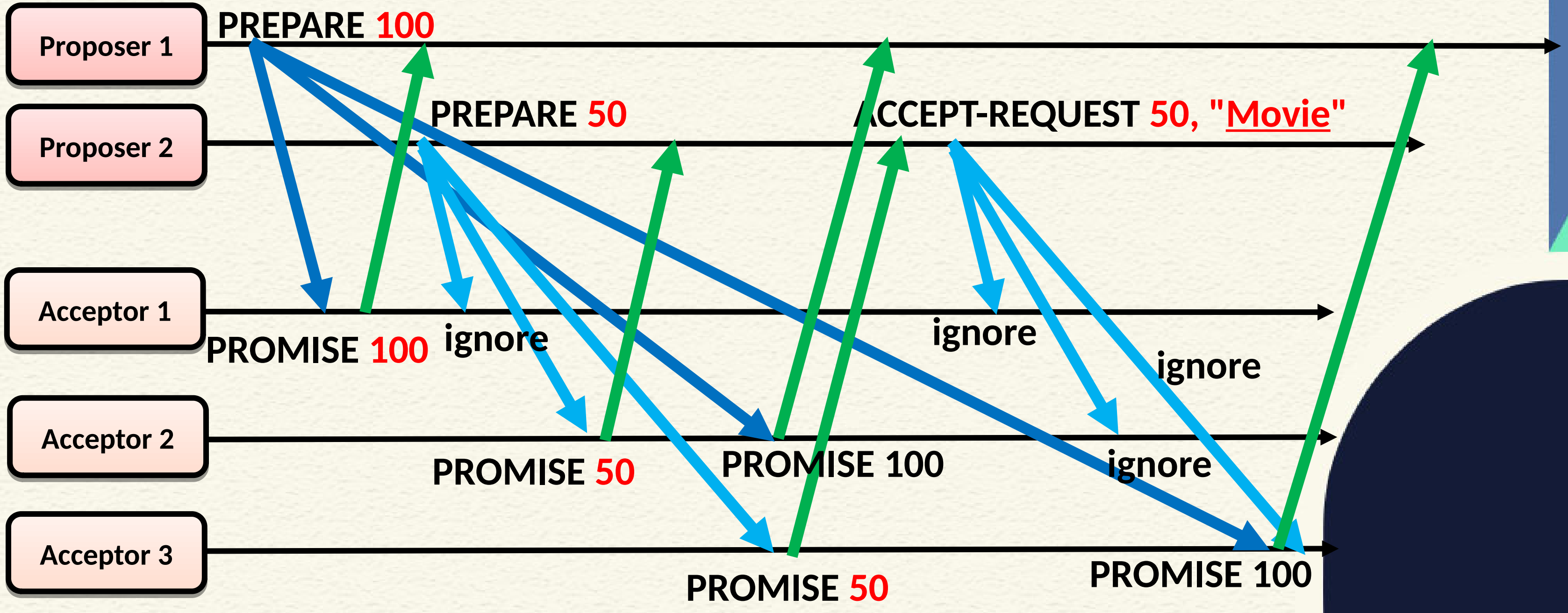
Majority Voting – Case 2



Majority Voting – Case 2

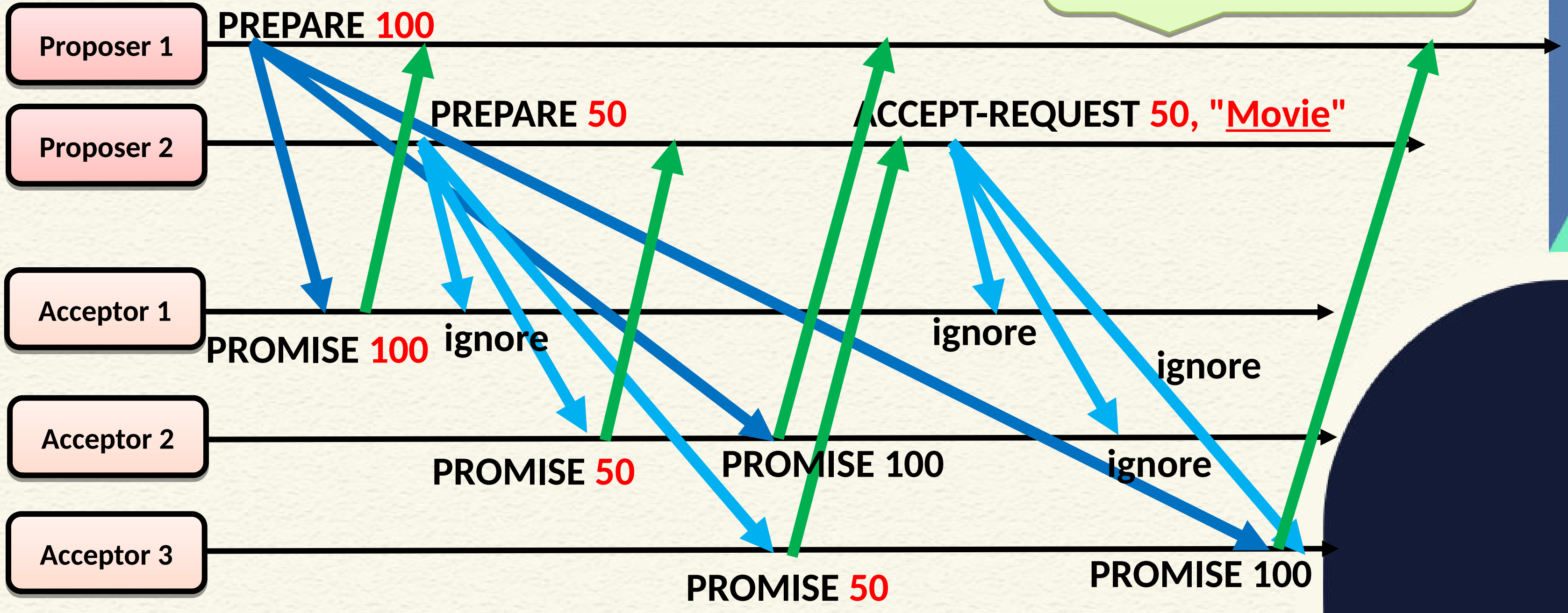


Majority Voting – Case 2

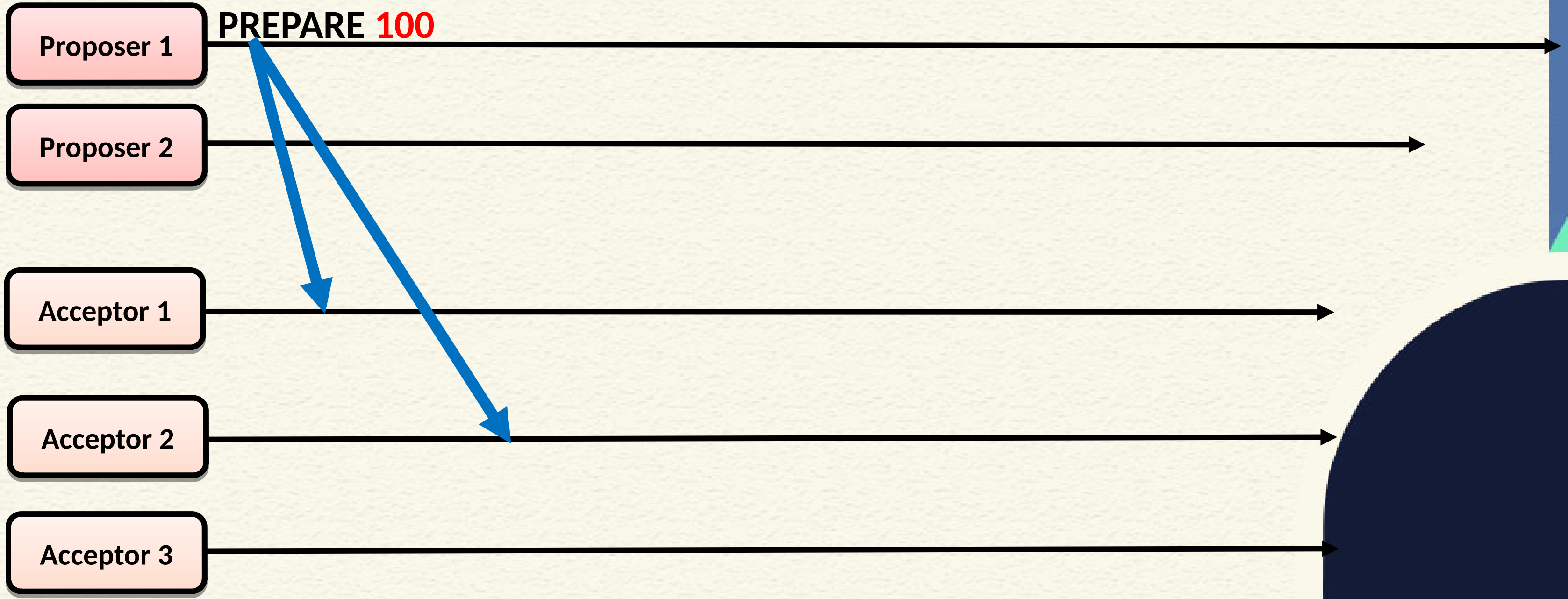


Majority Voting – Case 2

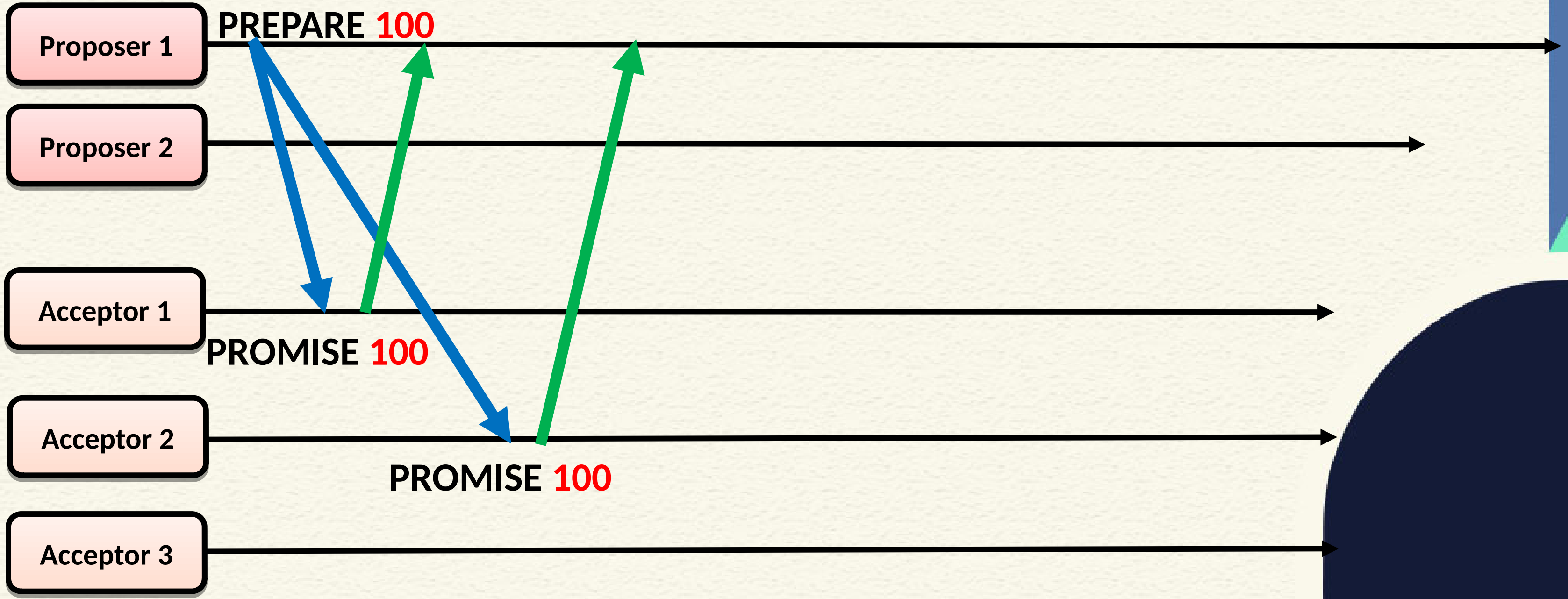
100 Becomes the majority



Liveness

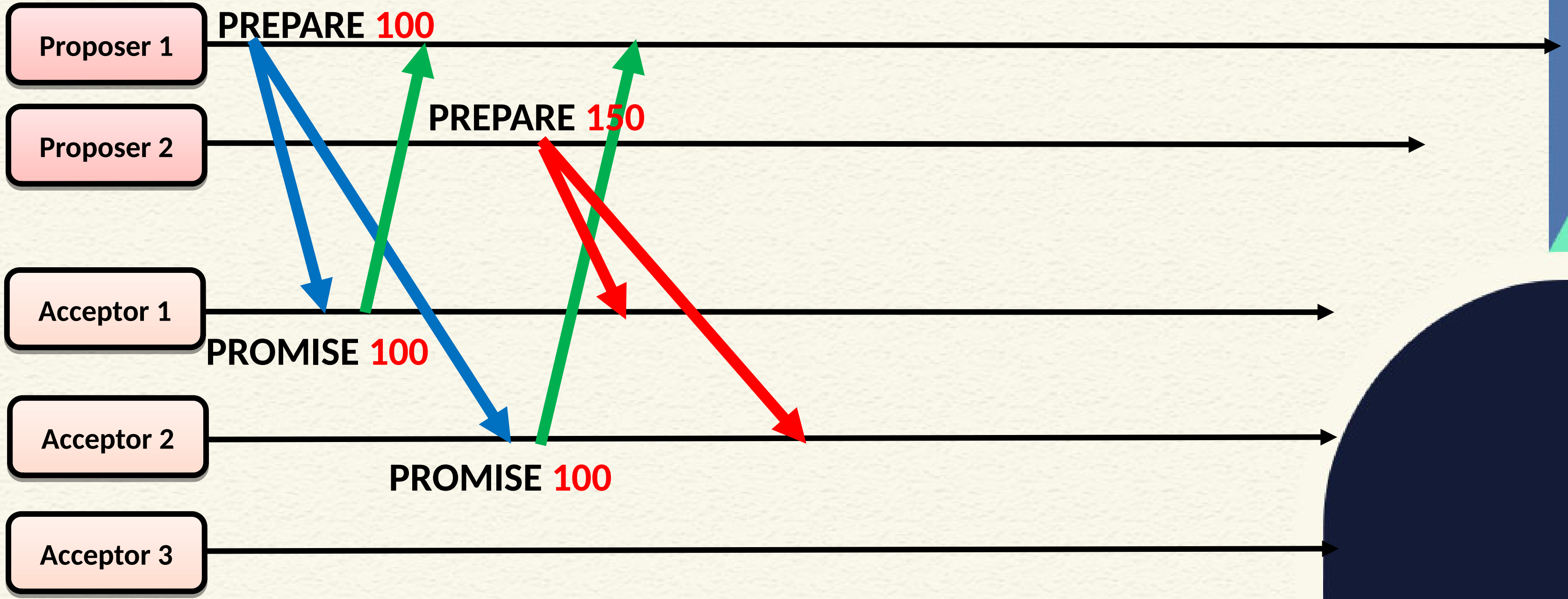

Proposer 1
PREPARE 100
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3

Liveness

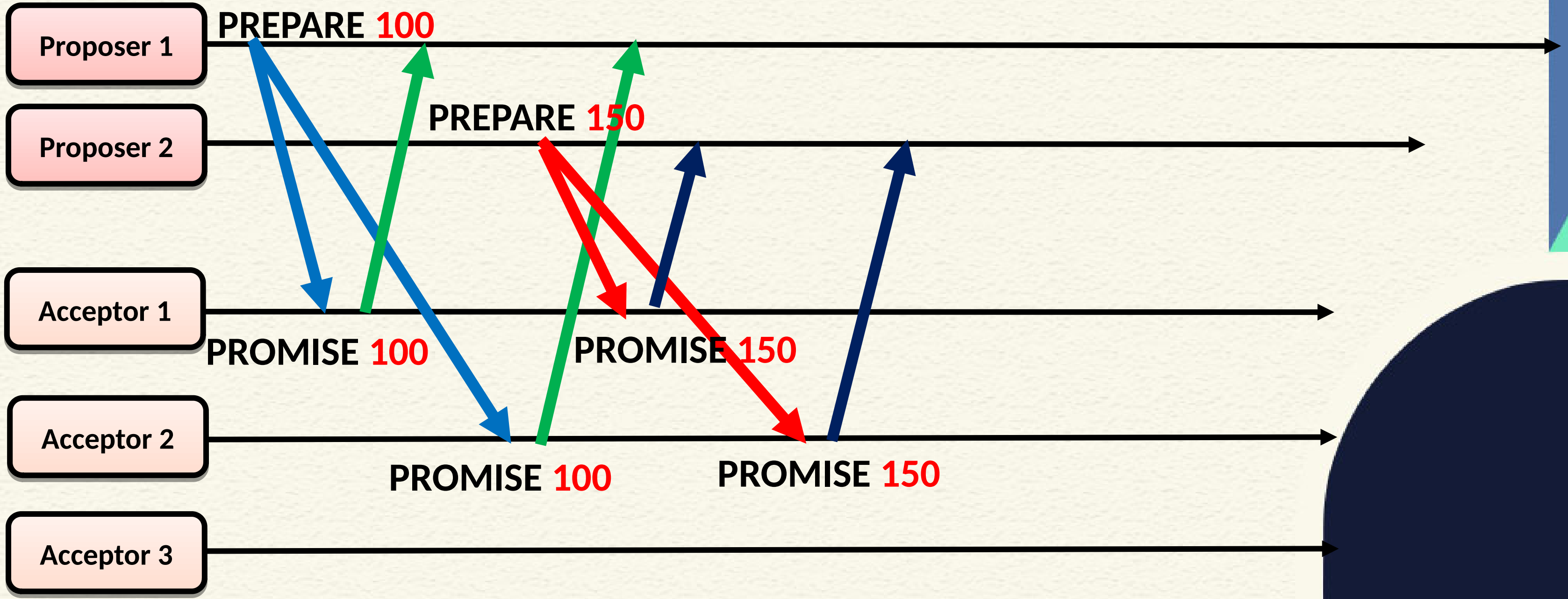


Proposer 1
PREPARE 100
Proposer 2
Acceptor 1
PROMISE 100
Acceptor 2
PROMISE 100
Acceptor 3

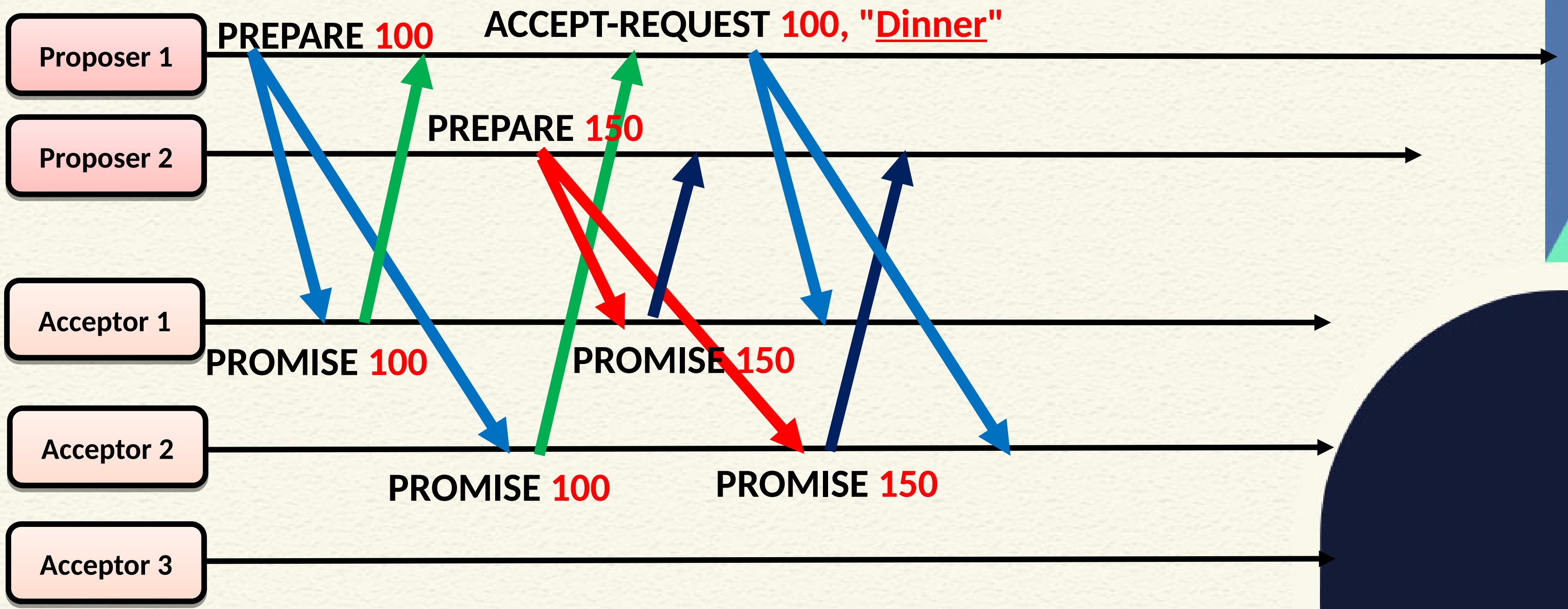
Liveness


Proposer 1
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3
PREPARE 100
PREPARE 150
PROMISE 100
PROMISE 100

Liveness

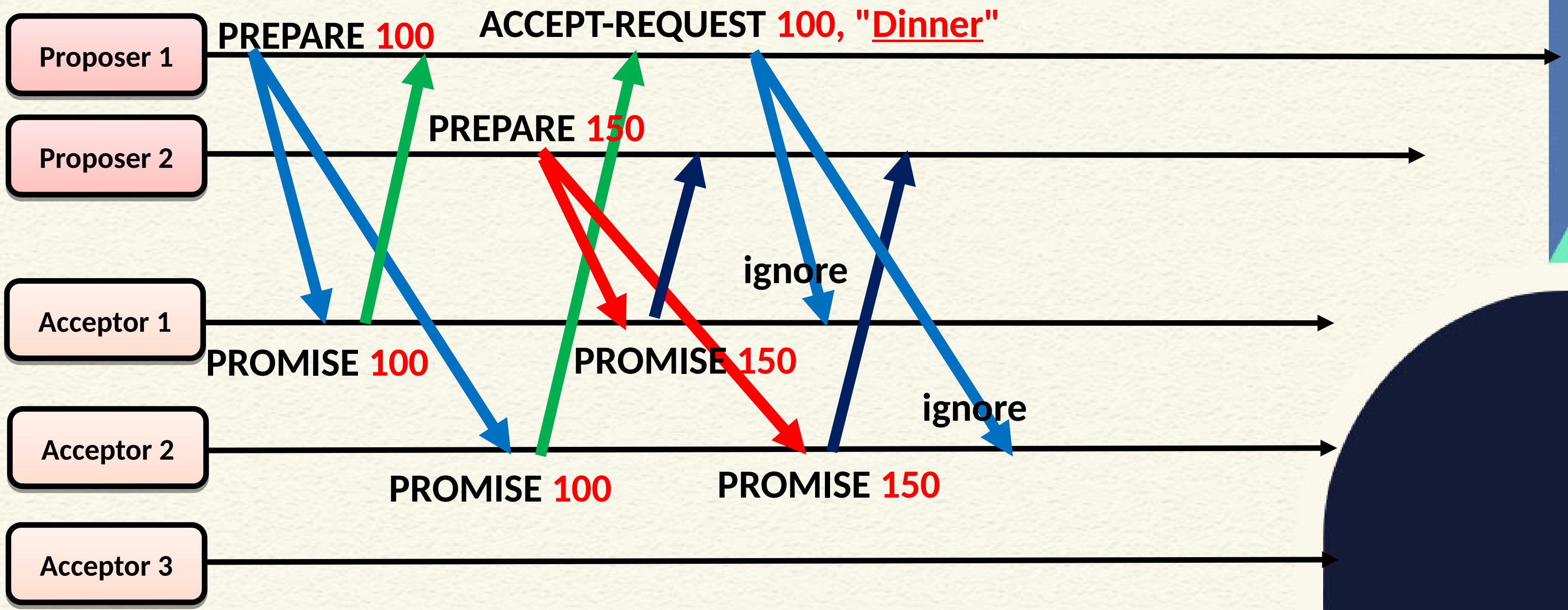

PREPARE 100
Proposer 1
PREPARE 150
Proposer 2
Acceptor 1
PROMISE 100
PROMISE 150
Acceptor 2
PROMISE 100
PROMISE 150
Acceptor 3

Liveness

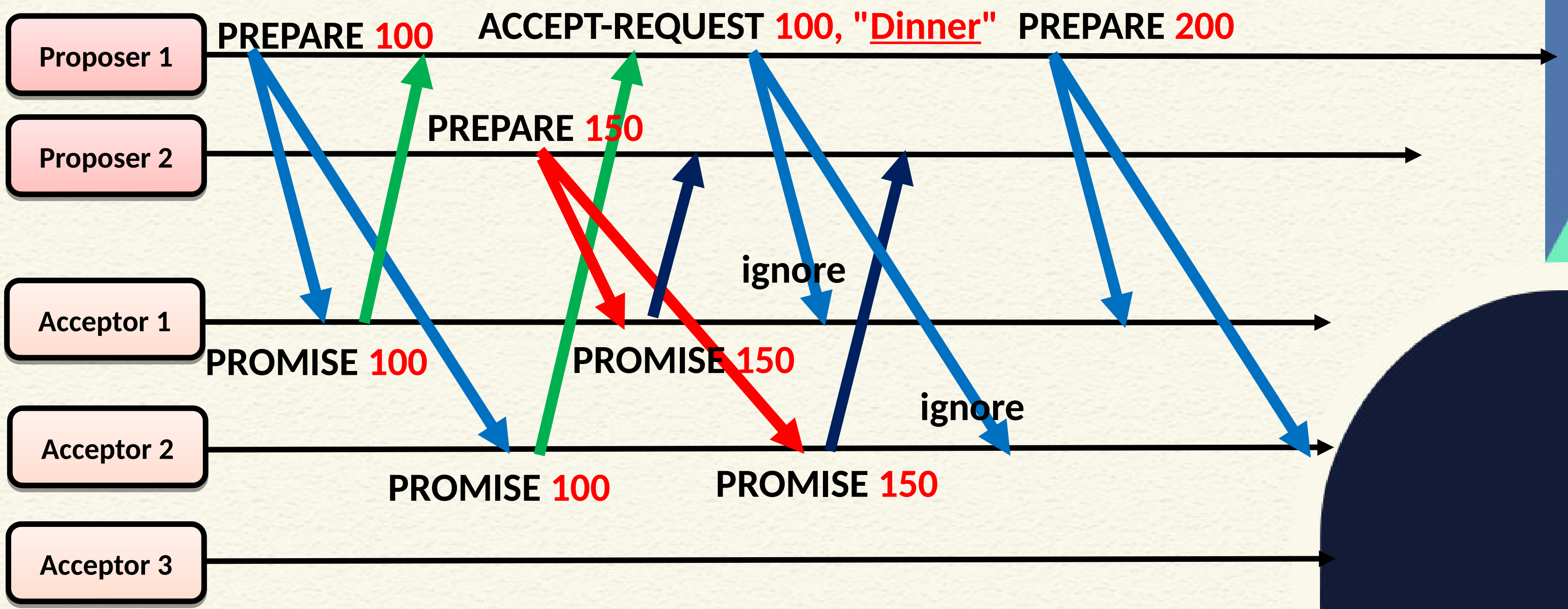


Proposer 1
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3
PREPARE 100
ACCEPT-REQUEST 100, "Dinner"
PREPARE 150
PROMISE 100
PROMISE 150
PROMISE 100
PROMISE 150

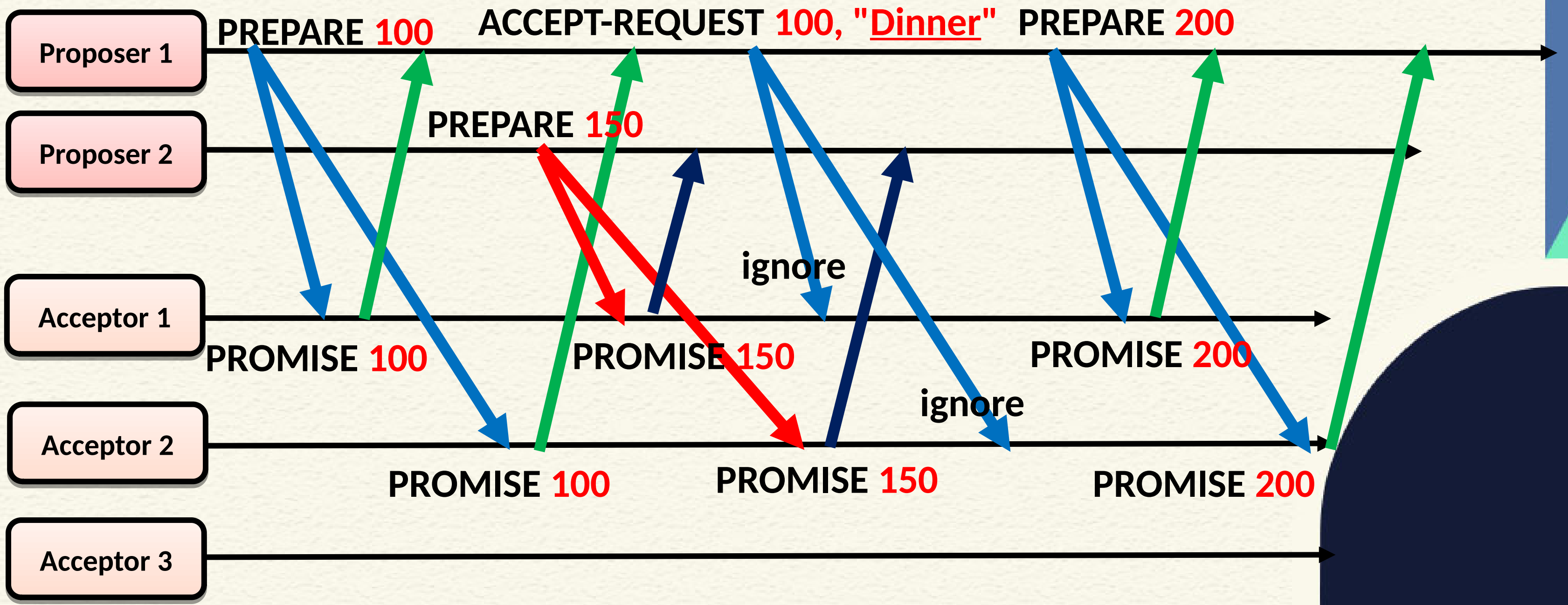
Liveness


Proposer 1
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3
PREPARE 100
ACCEPT-REQUEST 100, "Dinner"
PREPARE 150
ignore
PROMISE 100
PROMISE 150
ignore
PROMISE 100
PROMISE 150

Liveness

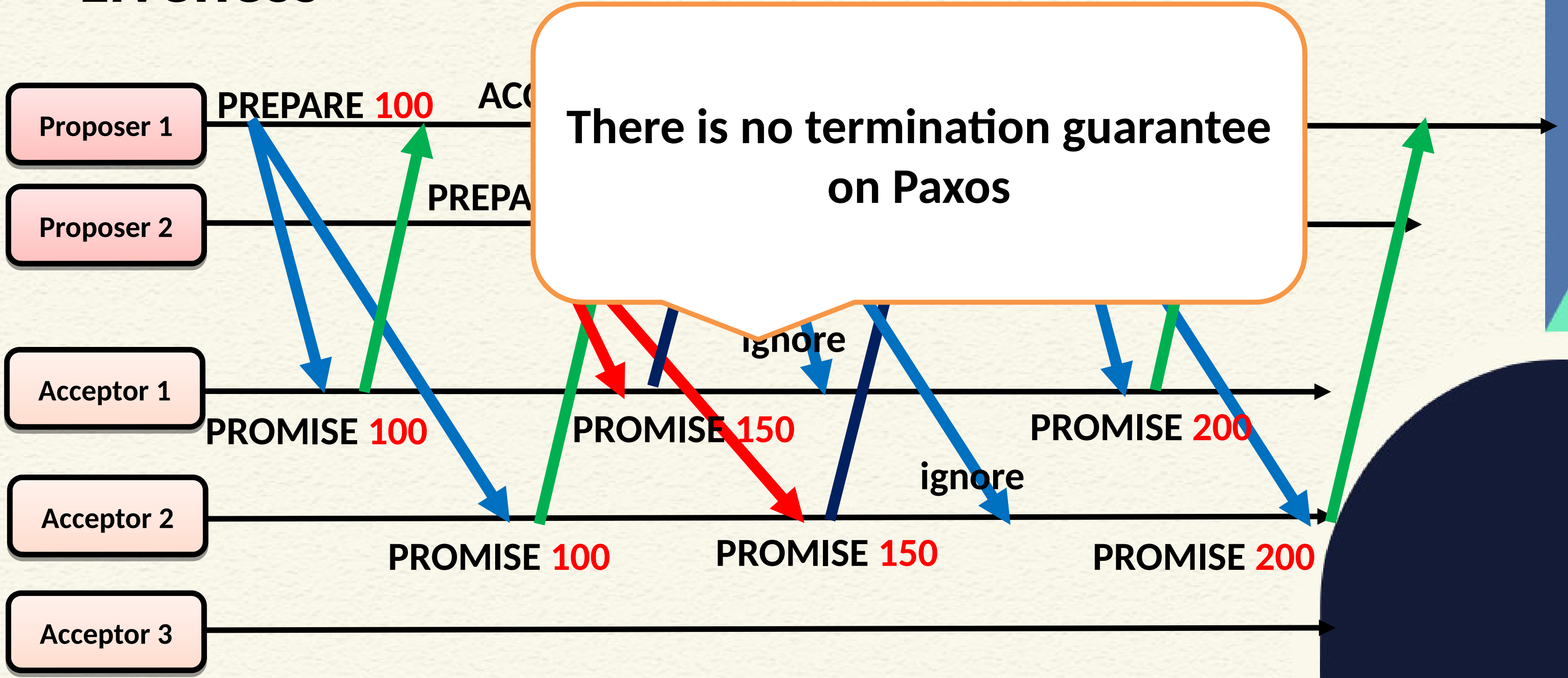

Proposer 1
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3
PREPARE 100
ACCEPT-REQUEST 100, "Dinner"
PREPARE 200
PREPARE 150
ignore
PROMISE 100
PROMISE 150
ignore
PROMISE 100
PROMISE 150

Liveness



Proposer 1
Proposer 2
Acceptor 1
Acceptor 2
Acceptor 3
PREPARE 100
ACCEPT-REQUEST 100, "Dinner"
PREPARE 200
PREPARE 150
ignore
PROMISE 100
PROMISE 150
PROMISE 200
ignore
PROMISE 100
PROMISE 150
PROMISE 200

Liveness



There is no termination guarantee on Paxos

Majority of Accepts

- Majority of acceptors accept a request with an ID and a value
  - Consensus has been reached
  - The consensus is on the **value**

- Accept request with a lower ID
  - Will not be accepted by the majority (Would require majority of promises with the lower ID, but we got for a higher one, hence the accept request)

Majority of Accepts

- Majority of acceptors accept a request with an ID and a value
  - Consensus has been reached
  - The consensus is on the **value**

- Accept request with a lower ID
  - Will not be accepted by the majority (Would require majority of promises with the lower ID, but we got for a higher one, hence the accept request)

Majority of Accepts

- Accept request with a higher ID but a **<u>different value</u>**
  - Will not be accepted by the majority
  - At least one acceptor will piggyback the previously accepted value (Remember, two majority implies that there is a common node)

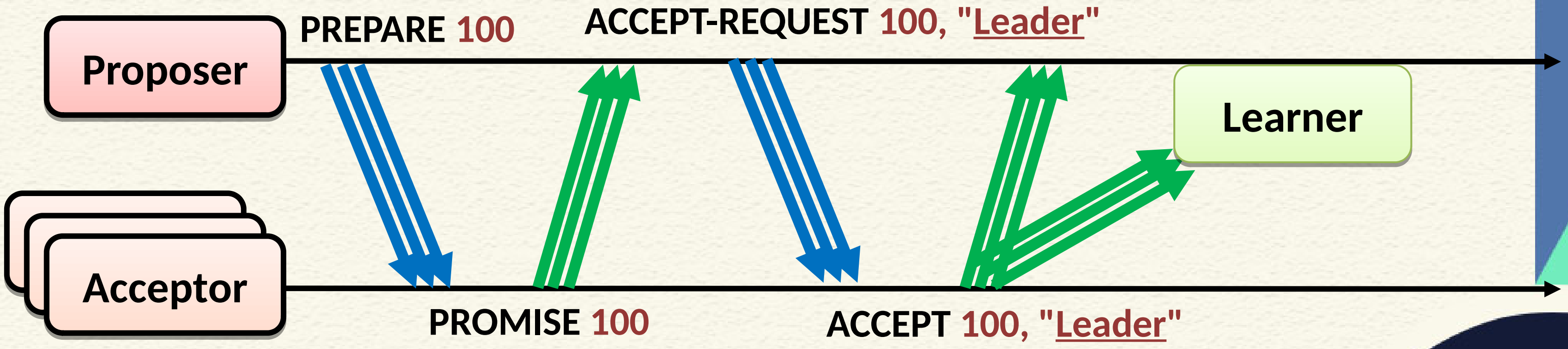
Majority of Accepts

- Accept request with a higher ID but a **<u>different value</u>**
  - Will not be accepted by the majority

So, the consensus is on the value

We need the ID to maintain the <u>current state of promise and accept</u>, so that multiple values does not propagate

Paxos for Leader Election



Multi-Paxos

- Applications often needs a continuous stream of agreed values
  - Commit the transactions in a replicated database – each transaction needs a consensus to be agreed upon by the replicas

- Run multiple instances of Paxos with different round numbers
  - Each value is associated with a round number

Multi-Paxos

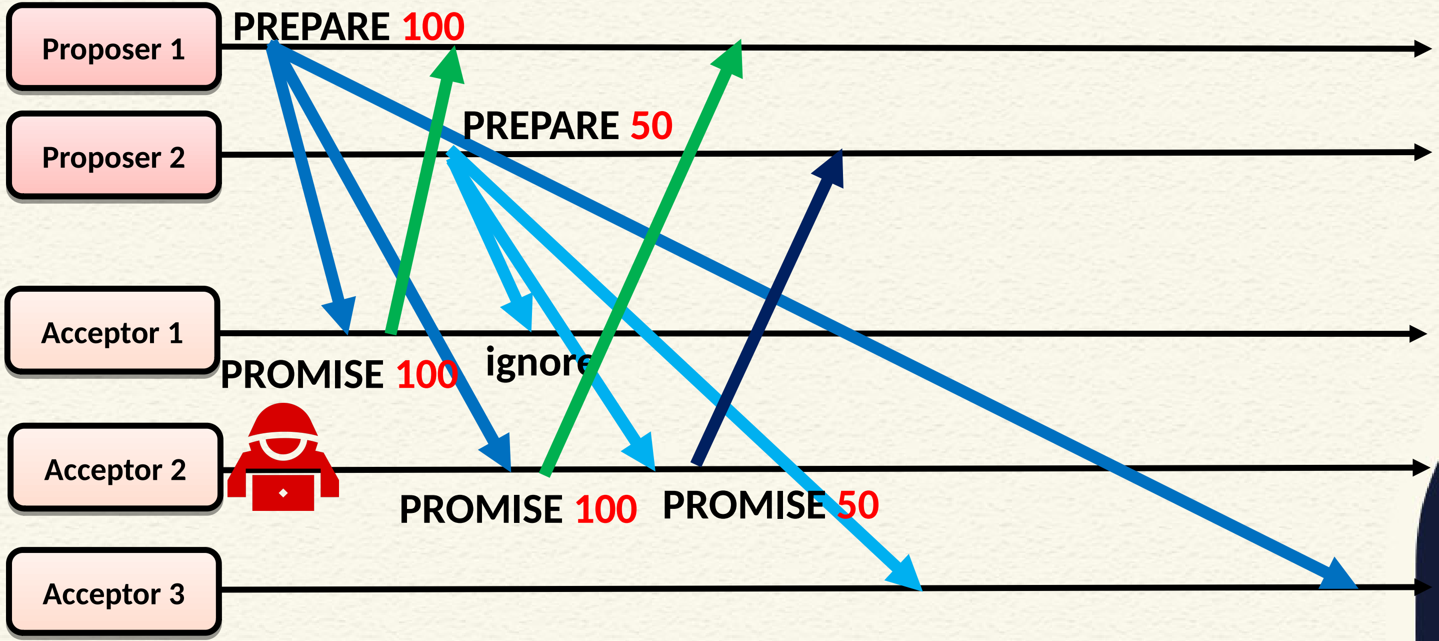
- If a value is already accepted for Round *n*, ignore the accept requests for a different value under Round *n*
    - Forward an ACCEPT IDp, (ROUNDn, VALUE) only when no value has been agreed upon for the Round *n*

Conclusion

- CFT consensus in asynchronous system – Paxos
  - Safety is ensured, but liveness is compromised

- Does Paxos work when a node sends a wrong message?

Conclusion – Attack on Paxos



Thank you