

EXTENDS *Naturals*, *FiniteSets*, *Sequences*, *TLC*

Is leader *ALIVE* or *CRASHED*

VARIABLE *leaderState*

A collection of heartbeat (*AppendEntries*) messages the leader has sent.

A single message is abstracted to represent the leader's index

VARIABLE *messages*

A representation of the *commitIndex* and term, leader increases index monotonically.

VARIABLE *leaderIndex*

VARIABLE *followerIndex*

$nodeIndexes \triangleq \langle leaderIndex, followerIndex \rangle$

Indicates whether the follower timed out after not hearing from the leader for the specified amount of time.

VARIABLE *isTimeout*

$vars \triangleq \langle leaderState, messages, nodeIndexes, isTimeout \rangle$

The leader crashes and doesn't recover

CrashLeader \triangleq

$\wedge leaderState = \text{"ALIVE"}$

$\wedge leaderState' = \text{"CRASHED"}$

$\wedge \text{UNCHANGED } \langle messages, nodeIndexes, isTimeout \rangle$

The leader sends the follower an *AppendEntries* message

SendMessage \triangleq

$\wedge leaderState = \text{"ALIVE"}$

$\wedge messages' = \text{Append}(messages, leaderIndex)$

$\wedge \text{UNCHANGED } \langle leaderState, nodeIndexes, isTimeout \rangle$

Helper function to remove a message from a sequence of messages

RemoveMessage(*i*, *seq*) \triangleq

$[j \in 1 \dots Len(seq) - 1 \mapsto \text{IF } j < i \text{ THEN } seq[j] \text{ ELSE } seq[j + 1]]$

The network drops a message

DropMessage \triangleq

$\wedge Len(messages) \geq 1$

$\wedge \exists i \in 1 \dots Len(messages) :$

$messages' = \text{RemoveMessage}(i, messages)$

$\wedge \text{UNCHANGED } \langle leaderState, nodeIndexes, isTimeout \rangle$

The leader increments its index

IncrementIndex \triangleq

$\wedge leaderState = \text{"ALIVE"}$

$$\wedge leaderIndex' = leaderIndex + 1$$

$$\wedge \text{UNCHANGED } \langle leaderState, messages, followerIndex, isTimeout \rangle$$

The follower receives a message from the leader.

$$ReceiveMessage \triangleq$$

$$\wedge Len(messages) \geq 1$$

$$\wedge \exists i \in 1 \dots Len(messages) :$$

$$((LET \ message \triangleq \ messages[i]$$

$$IN \ followerIndex' = IF \ message > followerIndex$$

$$THEN \ message$$

$$ELSE \ followerIndex)$$

$$\wedge \ messages' = RemoveMessage(i, messages))$$

$$\wedge \text{UNCHANGED } \langle leaderState, leaderIndex, isTimeout \rangle$$

The follower times out

$$Timeout \triangleq isTimeout' = \text{TRUE}$$

$$\wedge \text{UNCHANGED } \langle leaderState, messages, nodeIndexes \rangle$$

Initial state of model

$$Init \triangleq \wedge leaderState = \text{"ALIVE"}$$

$$\wedge messages = \langle \rangle$$

$$\wedge leaderIndex = 0$$

$$\wedge followerIndex = 0$$

$$\wedge isTimeout = \text{FALSE}$$

Next state function

$$Next \triangleq \wedge \neg isTimeout$$

$$\wedge \vee SendMessage$$

$$\vee IncrementIndex$$

$$\vee DropMessage$$

$$\vee ReceiveMessage$$

$$\vee CrashLeader$$

$$Spec \triangleq Init \wedge \Box [Next]_{vars}$$