
MODULE *Proposal*

INSTANCE *Naturals*

INSTANCE *FiniteSets*

INSTANCE *Sequences*

INSTANCE *TLC*

An empty constant
CONSTANT *Nil*

Event constants
CONSTANTS
 Change,
 Rollback

Phase constants
CONSTANTS
 Commit,
 Apply

$Phase \triangleq$
 {*Nil*,
 Commit,
 Apply}

Status constants
CONSTANTS
 Pending,
 InProgress,
 Complete,
 Failed

$Status \triangleq$
 {*Nil*,
 Pending,
 InProgress,
 Complete,
 Failed}

The set of all nodes
CONSTANT *Node*

Variables defined by other modules.

VARIABLES

configuration,
mastership,
conn,
target

A record of per-target proposals

VARIABLE *proposal*

A sequence of configuration changes used for model checking.

VARIABLE *history*

TypeOK \triangleq

$\forall i \in \text{DOMAIN } \textit{proposal} :$
 $\wedge \textit{proposal}[i].\textit{change.phase} \in \textit{Phase}$
 $\wedge \textit{proposal}[i].\textit{change.state} \in \textit{Status}$
 $\wedge \forall p \in \text{DOMAIN } \textit{proposal}[i].\textit{change.values} :$
 $\wedge \textit{proposal}[i].\textit{change.values}[p].\textit{index} \in \textit{Nat}$
 $\wedge \textit{proposal}[i].\textit{change.values}[p].\textit{value} \neq \textit{Nil} \Rightarrow$
 $\textit{proposal}[i].\textit{change.values}[p].\textit{value} \in \textit{STRING}$
 $\wedge \textit{proposal}[i].\textit{rollback.phase} \in \textit{Phase}$
 $\wedge \textit{proposal}[i].\textit{rollback.state} \in \textit{Status}$
 $\wedge \textit{proposal}[i].\textit{rollback.revision} \in \textit{Nat}$
 $\wedge \forall p \in \text{DOMAIN } \textit{proposal}[i].\textit{rollback.values} :$
 $\wedge \textit{proposal}[i].\textit{rollback.values}[p].\textit{index} \in \textit{Nat}$
 $\wedge \textit{proposal}[i].\textit{rollback.values}[p].\textit{value} \neq \textit{Nil} \Rightarrow$
 $\textit{proposal}[i].\textit{rollback.values}[p].\textit{value} \in \textit{STRING}$

LOCAL *CurrState* \triangleq [

proposals $\mapsto [i \in \text{DOMAIN } \textit{proposal} \mapsto \textit{proposal}[i] @@ [index \mapsto i]]$,
configuration $\mapsto \textit{configuration}$,
mastership $\mapsto \textit{mastership}$,
conn $\mapsto \textit{conn}$,
target $\mapsto \textit{target}$]

LOCAL *SuccState* \triangleq

LET

proposals $\triangleq \{i \in \text{DOMAIN } \textit{proposal}' : \\ i \in \text{DOMAIN } \textit{proposal} \Rightarrow \textit{proposal}'[i] \neq \textit{proposal}[i]\}$

IN

$[\textit{proposals} \mapsto [i \in \textit{proposals} \mapsto \textit{proposal}'[i] @@ [index \mapsto i]]] @@$
 $(\text{IF } \textit{configuration}' \neq \textit{configuration} \text{ THEN } [\textit{configuration} \mapsto \textit{configuration}'] \text{ ELSE } \langle \rangle) @@$
 $(\text{IF } \textit{mastership}' \neq \textit{mastership} \text{ THEN } [\textit{mastership} \mapsto \textit{mastership}'] \text{ ELSE } \langle \rangle) @@$
 $(\text{IF } \textit{conn}' \neq \textit{conn} \text{ THEN } [\textit{conn} \mapsto \textit{conn}'] \text{ ELSE } \langle \rangle) @@$
 $(\text{IF } \textit{target}' \neq \textit{target} \text{ THEN } [\textit{target} \mapsto \textit{target}'] \text{ ELSE } \langle \rangle)$

$Test \triangleq$ INSTANCE $Test$ WITH
 $File \leftarrow$ "Proposal.log"

LOCAL $Max(s) \triangleq$ CHOOSE $i \in s : \forall j \in s : i > j$

$CommitChange(n, i) \triangleq$
 $\wedge proposal[i].change.phase = Commit$
 $\wedge proposal[i].change.state = InProgress$
 If the committed index does not match the proposal index, commit the change.
 $\wedge \vee \wedge configuration.committed.index = i - 1$
 If the change is valid, update the committed index, revision, and values.
 $\wedge \vee \wedge configuration' = [configuration \text{ EXCEPT } !.committed.index = i,$
 $!.committed.revision = i,$
 $!.committed.values = proposal[i].change.values @@$
 $configuration.committed.values]$
 $\wedge history' = Append(history, [type \mapsto Change, phase \mapsto Commit, index \mapsto i])$
 If the change is invalid, update only the committed index.
 $\vee \wedge configuration' = [configuration \text{ EXCEPT } !.committed.index = i]$
 $\wedge UNCHANGED \langle history \rangle$
 $\wedge UNCHANGED \langle proposal \rangle$
 If both the committed index and committed revision were updated, the proposal was successful.
 $\vee \wedge configuration.committed.index = i$
 $\wedge configuration.committed.revision = i$
 $\wedge proposal' = [proposal \text{ EXCEPT } ![i].change.state = Complete]$
 $\wedge UNCHANGED \langle configuration, history \rangle$
 If the committed index was updated but the revision was not, the proposal failed validation.
 $\vee \wedge configuration.committed.index = i$
 $\wedge configuration.committed.revision \neq i$
 $\wedge proposal' = [proposal \text{ EXCEPT } ![i].change.state = Failed]$
 $\wedge UNCHANGED \langle configuration, history \rangle$
 $\wedge UNCHANGED \langle target \rangle$

$ApplyChange(n, i) \triangleq$
 $\wedge proposal[i].change.phase = Apply$
 $\wedge proposal[i].change.state = InProgress$
 If the applied index does not match the proposal index, apply the change.
 $\wedge \vee \wedge configuration.applied.index = i - 1$
 $\wedge configuration.state = Complete$
 $\wedge configuration.term = mastership.term$
 $\wedge conn[n].id = mastership.conn$
 $\wedge conn[n].connected$
 $\wedge target.running$
 If the change can be applied, update the index, revision, and values.
 $\wedge \vee \wedge target' = [target \text{ EXCEPT } !.values = proposal[i].change.values @@ target.values]$

$$\begin{aligned}
& \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.index} = i, \\
& \quad !.\text{applied.revision} = i, \\
& \quad !.\text{applied.values} = \text{proposal}[i].\text{change.values} @@ \\
& \quad \quad \quad \text{configuration.applied.values}] \\
& \wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Change}, \text{phase} \mapsto \text{Apply}, \text{index} \mapsto i]) \\
& \text{If the change is invalid, update only the applied index.} \\
& \vee \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.index} = i] \\
& \quad \wedge \text{UNCHANGED } \langle \text{target}, \text{history} \rangle \\
& \quad \wedge \text{UNCHANGED } \langle \text{proposal} \rangle \\
& \text{If the applied index and revision both match the proposal index, the change was successful.} \\
& \vee \wedge \text{configuration.applied.index} = i \\
& \quad \wedge \text{configuration.applied.revision} = i \\
& \quad \wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{change.state} = \text{Complete}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
& \text{If the applied index matches the proposal index but the revision does not, the proposal failed.} \\
& \vee \wedge \text{configuration.applied.index} = i \\
& \quad \wedge \text{configuration.applied.revision} \neq i \\
& \quad \wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{change.state} = \text{Failed}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
\text{CommitRollback}(n, i) & \triangleq \\
& \wedge \text{proposal}[i].\text{rollback.phase} = \text{Commit} \\
& \wedge \text{proposal}[i].\text{rollback.state} = \text{InProgress} \\
& \quad \text{If the committed revision matches the proposal revision, roll back to the previous revision.} \\
& \wedge \vee \wedge \text{configuration.committed.revision} = i \\
& \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{committed.revision} = \text{proposal}[i].\text{rollback.revision}, \\
& \quad \quad \quad !.\text{committed.values} = \text{proposal}[i].\text{rollback.values} @@ \\
& \quad \quad \quad \text{configuration.committed.values}] \\
& \quad \wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Rollback}, \text{phase} \mapsto \text{Commit}, \text{index} \mapsto i]) \\
& \quad \wedge \text{UNCHANGED } \langle \text{proposal} \rangle \\
& \quad \text{If the committed index matches the rollback index, complete the rollback.} \\
& \vee \wedge \text{configuration.committed.revision} = \text{proposal}[i].\text{rollback.revision} \\
& \quad \wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{rollback.state} = \text{Complete}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \wedge \text{UNCHANGED } \langle \text{target} \rangle \\
\text{ApplyRollback}(n, i) & \triangleq \\
& \wedge \text{proposal}[i].\text{rollback.phase} = \text{Apply} \\
& \wedge \text{proposal}[i].\text{rollback.state} = \text{InProgress} \\
& \quad \text{If the applied revision matches the proposal revision, roll back to the previous revision.} \\
& \wedge \vee \wedge \text{configuration.applied.revision} = i \\
& \quad \wedge \text{configuration.state} = \text{Complete} \\
& \quad \wedge \text{configuration.term} = \text{mastership.term} \\
& \quad \wedge \text{conn}[n].\text{id} = \text{mastership.conn} \\
& \quad \wedge \text{conn}[n].\text{connected}
\end{aligned}$$

$\wedge target.running$
 $\wedge target' = [target \text{ EXCEPT } !.values = proposal[i].rollback.values @@ target.values]$
 $\wedge configuration' = [configuration \text{ EXCEPT } !.applied.revision = proposal[i].rollback.revision,$
 $\quad \quad \quad !.applied.values = proposal[i].rollback.values @@$
 $\quad \quad \quad configuration.applied.values]$
 $\wedge history' = Append(history, [type \mapsto Rollback, phase \mapsto Apply, index \mapsto i])$
 $\wedge \text{UNCHANGED } \langle proposal \rangle$
 If the committed index matches the rollback index, complete the rollback.
 $\vee \wedge configuration.committed.revision = proposal[i].rollback.revision$
 $\wedge proposal' = [proposal \text{ EXCEPT } ![i].rollback.state = Complete]$
 $\wedge \text{UNCHANGED } \langle configuration, target, history \rangle$

Reconcile a proposal
 $ReconcileProposal(n, i) \triangleq$
 $\wedge i \in \text{DOMAIN } proposal$
 $\wedge mastership.master = n$
 $\wedge \vee CommitChange(n, i)$
 $\quad \vee ApplyChange(n, i)$
 $\quad \vee CommitRollback(n, i)$
 $\quad \vee ApplyRollback(n, i)$
 $\wedge \text{UNCHANGED } \langle mastership, conn \rangle$
