

---

MODULE *ConfigImpl*

---

INSTANCE *Naturals*  
 INSTANCE *FiniteSets*  
 INSTANCE *Sequences*  
 LOCAL INSTANCE *TLC*

---

This section specifies constant parameters for the model.  
 CONSTANT *LogEnabled*  
 ASSUME *LogEnabled* ∈ BOOLEAN  
 CONSTANT *None*  
 ASSUME *None* ∈ STRING  
 CONSTANT *Node*  
 ASSUME  $\forall n \in \text{Node} : n \in \text{STRING}$   
 CONSTANTS  
     *Change*,  
     *Rollback*  
 $\text{Event} \triangleq \{ \text{Change}, \text{Rollback} \}$   
 ASSUME  $\forall e \in \text{Event} : e \in \text{STRING}$   
 CONSTANTS  
     *Commit*,  
     *Apply*  
 $\text{Phase} \triangleq \{ \text{Commit}, \text{Apply} \}$   
 ASSUME  $\forall p \in \text{Phase} : p \in \text{STRING}$   
 CONSTANTS  
     *Pending*,  
     *InProgress*,  
     *Complete*,  
     *Aborted*,  
     *Failed*  
 $\text{State} \triangleq \{ \text{Pending}, \text{InProgress}, \text{Complete}, \text{Aborted}, \text{Failed} \}$   
 $\text{Done} \triangleq \{ \text{Complete}, \text{Aborted}, \text{Failed} \}$

ASSUME  $\forall s \in State : s \in \text{STRING}$

CONSTANT *Path*

ASSUME  $\forall p \in Path : p \in \text{STRING}$

CONSTANT *Value*

ASSUME  $\forall v \in Value : v \in \text{STRING}$

$AllValues \triangleq Value \cup \{None\}$

CONSTANT *NumProposals*

ASSUME  $NumProposals \in Nat$

---

This section defines model state variables.

$proposal \triangleq [ i \in 1 \dots Nat \mapsto [$   
     $phase \mapsto Phase,$   
     $change \mapsto [$   
         $values \mapsto Change,$   
         $commit \mapsto State,$   
         $apply \mapsto State],$   
     $rollback \mapsto [$   
         $index \mapsto Nat,$   
         $values \mapsto Change,$   
         $commit \mapsto State,$   
         $apply \mapsto State]]]$

$configuration \triangleq [$   
     $committed \mapsto [$   
         $index \mapsto Nat,$   
         $values \mapsto Change],$   
     $applied \mapsto [$   
         $index \mapsto Nat,$   
         $values \mapsto Change,$   
         $term \mapsto Nat]]]$

$mastership \triangleq [$   
     $master \mapsto \text{STRING},$   
     $term \mapsto Nat,$   
     $conn \mapsto Nat]$

$conn \triangleq [ n \in Node \mapsto [$   
     $id \mapsto Nat,$   
     $connected \mapsto \text{BOOLEAN} ]]$

$target \triangleq [$   
     $id \mapsto Nat,$   
     $values \mapsto Change,$   
     $running \mapsto \text{BOOLEAN} ]]$

VARIABLE *proposal*

VARIABLE *configuration*

VARIABLE *mastership*

VARIABLE *conn*

VARIABLE *target*

VARIABLE *history*

$\text{vars} \triangleq \langle \text{proposal}, \text{configuration}, \text{mastership}, \text{conn}, \text{target}, \text{history} \rangle$

---

LOCAL *MastershipLog*  $\triangleq$  INSTANCE *Log* WITH

*File*  $\leftarrow$  "Mastership.log",

*CurrState*  $\leftarrow$  [

*target*  $\mapsto$  *target*,

*mastership*  $\mapsto$  *mastership*,

*conns*  $\mapsto$  *conn*],

*SuccState*  $\leftarrow$  [

*target*  $\mapsto$  *target'*,

*mastership*  $\mapsto$  *mastership'*,

*conns*  $\mapsto$  *conn'*],

*Enabled*  $\leftarrow$  *LogEnabled*

LOCAL *ConfigurationLog*  $\triangleq$  INSTANCE *Log* WITH

*File*  $\leftarrow$  "Configuration.log",

*CurrState*  $\leftarrow$  [

*configuration*  $\mapsto$  *configuration*,

*target*  $\mapsto$  *target*,

*mastership*  $\mapsto$  *mastership*,

*conns*  $\mapsto$  *conn*],

*SuccState*  $\leftarrow$  [

*configuration*  $\mapsto$  *configuration'*,

*target*  $\mapsto$  *target'*,

*mastership*  $\mapsto$  *mastership'*,

*conns*  $\mapsto$  *conn'*],

*Enabled*  $\leftarrow$  *LogEnabled*

LOCAL *ProposalLog*  $\triangleq$  INSTANCE *Log* WITH

*File*  $\leftarrow$  "Proposal.log",

*CurrState*  $\leftarrow$  [

*proposals*  $\mapsto$   $[i \in \{i \in \text{DOMAIN } \text{proposal} : \text{proposal}[i].\text{phase} \neq \text{None}\} \mapsto \text{proposal}[i]]$ ,

*configuration*  $\mapsto$  *configuration*,

*target*  $\mapsto$  *target*,

$$\begin{aligned}
& \text{mastership} \mapsto \text{mastership}, \\
& \text{conns} \mapsto \text{conn}], \\
\text{SuccState} \leftarrow [ & \\
& \text{proposals} \mapsto [i \in \{i \in \text{DOMAIN } \text{proposal}' : \text{proposal}'[i].\text{phase} \neq \text{None}\} \mapsto \text{proposal}'[i]], \\
& \text{configuration} \mapsto \text{configuration}', \\
& \text{target} \mapsto \text{target}', \\
& \text{mastership} \mapsto \text{mastership}', \\
& \text{conns} \mapsto \text{conn'}], \\
& \text{Enabled} \leftarrow \text{LogEnabled}
\end{aligned}$$


---

This section models configuration target.

$$\begin{aligned}
\text{StartTarget} &\triangleq \\
&\wedge \neg \text{target.running} \\
&\wedge \text{target}' = [\text{target} \text{ EXCEPT } !.\text{id} = \text{target.id} + 1, \\
&\quad \quad \quad !.\text{running} = \text{TRUE}] \\
&\wedge \text{UNCHANGED } \langle \text{proposal}, \text{configuration}, \text{mastership}, \text{conn}, \text{history} \rangle \\
\text{StopTarget} &\triangleq \\
&\wedge \text{target.running} \\
&\wedge \text{target}' = [\text{target} \text{ EXCEPT } !.\text{running} = \text{FALSE}, \\
&\quad \quad \quad !.\text{values} = [p \in \{\} \mapsto [\text{value} \mapsto \text{None}]]] \\
&\wedge \text{conn}' = [n \in \text{Node} \mapsto [\text{conn}[n] \text{ EXCEPT } !.\text{connected} = \text{FALSE}]] \\
&\wedge \text{UNCHANGED } \langle \text{proposal}, \text{configuration}, \text{mastership}, \text{history} \rangle
\end{aligned}$$


---

This section models nodes connection to the configuration target.

$$\begin{aligned}
\text{ConnectNode}(n) &\triangleq \\
&\wedge \neg \text{conn}[n].\text{connected} \\
&\wedge \text{target.running} \\
&\wedge \text{conn}' = [\text{conn} \text{ EXCEPT } ![n].\text{id} = \text{conn}[n].\text{id} + 1, \\
&\quad \quad \quad ![n].\text{connected} = \text{TRUE}] \\
&\wedge \text{UNCHANGED } \langle \text{proposal}, \text{configuration}, \text{mastership}, \text{target}, \text{history} \rangle \\
\text{DisconnectNode}(n) &\triangleq \\
&\wedge \text{conn}[n].\text{connected} \\
&\wedge \text{conn}' = [\text{conn} \text{ EXCEPT } ![n].\text{connected} = \text{FALSE}] \\
&\wedge \text{UNCHANGED } \langle \text{proposal}, \text{configuration}, \text{mastership}, \text{target}, \text{history} \rangle
\end{aligned}$$


---

This section models *mastership* reconciliation.

$$\begin{aligned}
\text{ReconcileMastership}(n) &\triangleq \\
&\wedge \vee \wedge \text{conn}[n].\text{connected} \\
&\quad \wedge \text{mastership.master} = \text{None}
\end{aligned}$$





until that rollback is complete. The index represents the index to which the proposal at  $changeIndex + 1$  rolls back.

$\wedge \text{UNCHANGED } \langle \text{proposal}, \text{target}, \text{history} \rangle$

$\text{CommitRollback}(n, i) \triangleq$

'index' is the current index committed to the configuration

'changeIndex' is the maximum change index committed to the configuration

'targetIndex' is the index of the proposal currently being committed

*targetIndex* is always changed first. Once the rollback is committed, the index will be decremented to match the *targetIndex*. The next time a change is committed, the index will increase again. If the committed index is equal to this proposal index, this proposal is the next to be rolled back. To roll back a proposal, the target index is set to the proposal's rollback index. When the rollback is committed, the committed index is set to the proposal's rollback index, thus matching the *targetIndex*. This unblocks new changes to be committed.

$\wedge \vee \wedge \text{proposal}[i].\text{rollback.commit} = \text{Pending}$   
 $\wedge \text{configuration.committed.changeIndex} \geq i$   
 $\wedge \text{configuration.committed.index} = i$   
 $\wedge \vee \wedge \text{configuration.committed.targetIndex} = i$   
 $\wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{committed.targetIndex} = \text{proposal}[i].\text{rollback.index}]$   
 $\wedge \text{UNCHANGED } \langle \text{proposal} \rangle$   
 $\vee \wedge \text{configuration.committed.targetIndex} = \text{proposal}[i].\text{rollback.index}$   
 $\wedge \vee \wedge \text{proposal}[i].\text{change.commit} \neq \text{Aborted}$   
 $\wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{InProgress}]$   
 $\vee \wedge \text{proposal}[i].\text{change.commit} = \text{Aborted}$   
 $\wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{Complete}]$   
 $\wedge \text{UNCHANGED } \langle \text{configuration} \rangle$   
 $\wedge \text{UNCHANGED } \langle \text{history} \rangle$   
 $\vee \wedge \text{proposal}[i].\text{rollback.commit} = \text{InProgress}$   
 $\wedge \vee \wedge \text{configuration.committed.index} = i$   
 $\wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{committed.index} = \text{proposal}[i].\text{rollback.index},$   
 $\quad !.\text{committed.values} = \text{proposal}[i].\text{rollback.values} @@$   
 $\quad \text{configuration.committed.values}]$   
 $\wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Rollback}, \text{phase} \mapsto \text{Commit}, \text{index} \mapsto i])$   
 $\wedge \text{UNCHANGED } \langle \text{proposal} \rangle$   
 $\vee \wedge \text{configuration.committed.index} = \text{proposal}[i].\text{rollback.index}$   
 $\wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{Complete}]$   
 $\wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle$   
 $\vee \wedge \text{proposal}[i].\text{rollback.commit} = \text{Complete}$   
 $\wedge \text{proposal}[i].\text{change.commit} = \text{Aborted}$   
 $\wedge \text{configuration.committed.targetIndex} = \text{proposal}[i].\text{rollback.index}$   
 $\wedge \text{configuration.committed.index} \neq \text{proposal}[i].\text{rollback.index}$   
 $\wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{committed.index} = \text{proposal}[i].\text{rollback.index}]$   
 $\wedge \text{UNCHANGED } \langle \text{proposal}, \text{history} \rangle$   
 $\wedge \text{UNCHANGED } \langle \text{target} \rangle$



$ApplyRollback(n, i) \triangleq$

'index' is the current index applied to the configuration  
 'changeIndex' is the maximum change index applied to the configuration  
 'targetIndex' is the index of the proposal currently being applied  
*targetIndex* is always changed first. Once the rollback is applied, the  
 index will be decremented to match the *targetIndex*. The next time a change  
 is applied, the index will increase again. If the applied index is equal  
 to this proposal index, this proposal is the next to be rolled back. To roll  
 back a proposal, the target index is set to the proposal's rollback index.  
 When the rollback is applied, the applied index is set to the proposal's  
 rollback index, thus matching the *targetIndex*. This unblocks new changes  
 to be applied.

$\wedge \vee \wedge proposal[i].rollback.apply = Pending$   
 $\wedge configuration.committed.index \leq proposal[i].rollback.index$   
 $\wedge configuration.applied.changeIndex \geq i$   
 $\wedge configuration.applied.index = i$   
 $\wedge \vee \wedge configuration.applied.targetIndex = i$   
 $\wedge configuration' = [configuration \text{ EXCEPT } !.applied.targetIndex = proposal[i].rollback.index]$   
 $\wedge UNCHANGED \langle proposal \rangle$   
 $\vee \wedge configuration.applied.targetIndex = proposal[i].rollback.index$   
 $\wedge proposal' = [proposal \text{ EXCEPT } ![i].rollback.apply = InProgress]$   
 $\wedge UNCHANGED \langle configuration \rangle$   
 $\wedge UNCHANGED \langle target, history \rangle$   
 $\vee \wedge proposal[i].rollback.apply = InProgress$   
 $\wedge \vee \wedge configuration.applied.index = i$   
 Verify the applied term is the current *mastership* term to ensure the  
 configuration has been synchronized following restarts.  
 $\wedge configuration.applied.term = mastership.term$   
 Verify the node's connection to the target.  
 $\wedge conn[n].connected$   
 $\wedge target.running$   
 $\wedge target' = [target \text{ EXCEPT } !.values = proposal[i].rollback.values @@ target.values]$   
 $\wedge configuration' = [configuration \text{ EXCEPT } !.applied.index = proposal[i].rollback.index,$   
 $!.applied.values = proposal[i].rollback.values @@$   
 $configuration.applied.values]$   
 $\wedge history' = Append(history, [type \mapsto Rollback, phase \mapsto Apply, index \mapsto i])$   
 $\wedge UNCHANGED \langle proposal \rangle$   
 $\vee \wedge configuration.applied.index \neq i$   
 $\wedge proposal' = [proposal \text{ EXCEPT } ![i].rollback.apply = Complete]$   
 $\wedge UNCHANGED \langle configuration, target, history \rangle$

$ReconcileProposal(n, i) \triangleq$

$\wedge mastership.master = n$   
 $\wedge \vee CommitChange(n, i)$   
 $\vee ApplyChange(n, i)$

$$\begin{aligned}
& \vee \text{CommitRollback}(n, i) \\
& \vee \text{ApplyRollback}(n, i) \\
& \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn} \rangle
\end{aligned}$$


---

This section models changes to the proposal queue.

$$\begin{aligned}
& \text{Propose change at index 'i'} \\
& \text{ProposeChange}(i) \triangleq \\
& \quad \wedge \text{proposal}[i].\text{phase} = \text{None} \\
& \quad \wedge i - 1 \in \text{DOMAIN } \text{proposal} \Rightarrow \text{proposal}[i - 1].\text{phase} \neq \text{None} \\
& \quad \wedge \exists p \in \text{Path}, v \in \text{AllValues} : \\
& \quad \quad \wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } \begin{array}{l} ! [i].\text{phase} = \text{Change}, \\ ! [i].\text{change.values} = (p :> [\text{value} \mapsto v]), \\ ! [i].\text{change.commit} = \text{Pending}, \\ ! [i].\text{change.apply} = \text{Pending} \end{array} \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{mastership}, \text{conn}, \text{target}, \text{history} \rangle
\end{aligned}$$

$$\begin{aligned}
& \text{Rollback proposed change at index 'i'} \\
& \text{ProposeRollback}(i) \triangleq \\
& \quad \wedge \text{proposal}[i].\text{phase} = \text{Change} \\
& \quad \wedge \text{proposal}' = [\text{proposal} \text{ EXCEPT } \begin{array}{l} ! [i].\text{phase} = \text{Rollback}, \\ ! [i].\text{rollback.commit} = \text{Pending}, \\ ! [i].\text{rollback.apply} = \text{Pending} \end{array} \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{mastership}, \text{conn}, \text{target}, \text{history} \rangle
\end{aligned}$$


---

Formal specification, constraints, and theorems.

$$\begin{aligned}
& \text{Init} \triangleq \\
& \quad \wedge \text{proposal} = [ \\
& \quad \quad i \in 1 \dots \text{NumProposals} \mapsto [ \\
& \quad \quad \quad \text{phase} \mapsto \text{None}, \\
& \quad \quad \quad \text{change} \mapsto [ \\
& \quad \quad \quad \quad \text{values} \mapsto [p \in \{\} \mapsto [\text{index} \mapsto 0, \text{value} \mapsto \text{None}]], \\
& \quad \quad \quad \quad \text{commit} \mapsto \text{None}, \\
& \quad \quad \quad \quad \text{apply} \mapsto \text{None}], \\
& \quad \quad \quad \text{rollback} \mapsto [ \\
& \quad \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \quad \text{values} \mapsto [p \in \{\} \mapsto [\text{index} \mapsto 0, \text{value} \mapsto \text{None}]], \\
& \quad \quad \quad \quad \text{commit} \mapsto \text{None}, \\
& \quad \quad \quad \quad \text{apply} \mapsto \text{None}]] \\
& \quad \wedge \text{configuration} = [ \\
& \quad \quad \text{committed} \mapsto [ \\
& \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \text{changeIndex} \mapsto 0,
\end{aligned}$$

$$\begin{aligned}
& \text{targetIndex} \mapsto 0, \\
& \text{values} \mapsto [p \in \{\} \mapsto [\text{index} \mapsto 0, \text{value} \mapsto \text{None}]]], \\
& \text{applied} \mapsto [ \\
& \quad \text{index} \mapsto 0, \\
& \quad \text{changeIndex} \mapsto 0, \\
& \quad \text{targetIndex} \mapsto 0, \\
& \quad \text{term} \mapsto 0, \\
& \quad \text{target} \mapsto 0, \\
& \quad \text{values} \mapsto [p \in \{\} \mapsto [\text{index} \mapsto 0, \text{value} \mapsto \text{None}]]], \\
& \quad \text{status} \mapsto \text{Pending}] \\
& \wedge \text{mastership} = [\text{master} \mapsto \text{None}, \text{term} \mapsto 0, \text{conn} \mapsto 0] \\
& \wedge \text{conn} = [n \in \text{Node} \mapsto [\text{id} \mapsto 0, \text{connected} \mapsto \text{FALSE}]] \\
& \wedge \text{target} = [ \\
& \quad \text{id} \mapsto 0, \\
& \quad \text{values} \mapsto [p \in \{\} \mapsto [\text{index} \mapsto 0, \text{value} \mapsto \text{None}]], \\
& \quad \text{running} \mapsto \text{FALSE}] \\
& \wedge \text{history} = \langle \rangle \\
\text{Next} & \triangleq \\
& \vee \exists i \in 1 \dots \text{NumProposals} : \\
& \quad \vee \text{ProposeChange}(i) \\
& \quad \vee \text{ProposeRollback}(i) \\
& \vee \exists n \in \text{Node}, i \in \text{DOMAIN } \text{proposal} : \\
& \quad \text{ProposalLog! Action}(\text{ReconcileProposal}(n, i), [\text{node} \mapsto n, \text{index} \mapsto i]) \\
& \vee \exists n \in \text{Node} : \\
& \quad \text{ConfigurationLog! Action}(\text{ReconcileConfiguration}(n), [\text{node} \mapsto n]) \\
& \vee \exists n \in \text{Node} : \\
& \quad \text{MastershipLog! Action}(\text{ReconcileMastership}(n), [\text{node} \mapsto n]) \\
& \vee \exists n \in \text{Node} : \\
& \quad \vee \text{ConnectNode}(n) \\
& \quad \vee \text{DisconnectNode}(n) \\
& \vee \text{StartTarget} \\
& \vee \text{StopTarget} \\
\text{Spec} & \triangleq \\
& \wedge \text{Init} \\
& \wedge \Box [\text{Next}]_{\text{vars}} \\
& \wedge \forall i \in 1 \dots \text{NumProposals} : \text{WF}_{\langle \text{proposal}, \text{configuration}, \text{mastership}, \text{conn}, \text{target}, \text{history} \rangle} (\text{ProposeChange}(i) \vee \text{ProposeRollback}(i)) \\
& \wedge \forall n \in \text{Node}, i \in 1 \dots \text{NumProposals} : \text{WF}_{\langle \text{proposal}, \text{configuration}, \text{mastership}, \text{conn}, \text{target}, \text{history} \rangle} (\text{ReconcileProposal}(n, i)) \\
& \wedge \forall n \in \text{Node} : \text{WF}_{\langle \text{configuration}, \text{mastership}, \text{conn}, \text{target} \rangle} (\text{ReconcileConfiguration}(n)) \\
& \wedge \forall n \in \text{Node} : \text{WF}_{\langle \text{mastership}, \text{conn}, \text{target} \rangle} (\text{ReconcileMastership}(n)) \\
& \wedge \forall n \in \text{Node} : \text{WF}_{\langle \text{conn}, \text{target} \rangle} (\text{ConnectNode}(n) \vee \text{DisconnectNode}(n)) \\
& \wedge \text{WF}_{\langle \text{target} \rangle} (\text{StartTarget}) \\
& \wedge \text{WF}_{\langle \text{target} \rangle} (\text{StopTarget}) \\
\text{Alias} & \triangleq [
\end{aligned}$$

$$\begin{aligned}
& \text{configuration} \mapsto [ \\
& \quad \text{committed} \mapsto [ \\
& \quad \quad \text{values} \mapsto \text{configuration.committed.values}, \\
& \quad \text{applied} \mapsto [ \\
& \quad \quad \text{term} \mapsto \text{configuration.applied.term}, \\
& \quad \quad \text{target} \mapsto \text{configuration.applied.target}, \\
& \quad \quad \text{values} \mapsto \text{configuration.applied.values}, \\
& \quad \text{status} \mapsto \text{configuration.status}], \\
& \text{proposal} \mapsto [i \in \text{DOMAIN proposal} \mapsto [ \\
& \quad \text{phase} \mapsto \text{proposal}[i].\text{phase}, \\
& \quad \text{values} \mapsto [p \in \text{DOMAIN proposal}[i].\text{change.values} \mapsto \text{proposal}[i].\text{change.values}[p].\text{value}], \\
& \quad \text{change} \mapsto [ \\
& \quad \quad \text{commit} \mapsto \text{IF } \wedge \text{proposal}[i].\text{change.commit} = \text{InProgress} \\
& \quad \quad \quad \wedge \text{configuration.committed.changeIndex} \geq i \\
& \quad \quad \quad \text{THEN Complete} \\
& \quad \quad \quad \text{ELSE proposal}[i].\text{change.commit}, \\
& \quad \quad \text{apply} \mapsto \text{IF } \wedge \text{proposal}[i].\text{change.apply} = \text{InProgress} \\
& \quad \quad \quad \wedge \text{configuration.applied.changeIndex} \geq i \\
& \quad \quad \quad \text{THEN Complete} \\
& \quad \quad \quad \text{ELSE proposal}[i].\text{change.apply}], \\
& \quad \text{rollback} \mapsto [ \\
& \quad \quad \text{commit} \mapsto \text{IF } \wedge \text{proposal}[i].\text{rollback.commit} = \text{InProgress} \\
& \quad \quad \quad \wedge \text{configuration.committed.index} \neq i \\
& \quad \quad \quad \text{THEN Complete} \\
& \quad \quad \quad \text{ELSE proposal}[i].\text{rollback.commit}, \\
& \quad \quad \text{apply} \mapsto \text{IF } \wedge \text{proposal}[i].\text{rollback.apply} = \text{InProgress} \\
& \quad \quad \quad \wedge \text{configuration.applied.index} \neq i \\
& \quad \quad \quad \text{THEN Complete} \\
& \quad \quad \quad \text{ELSE proposal}[i].\text{rollback.apply}]], \\
& \text{mastership} \mapsto \text{mastership}, \\
& \text{conn} \mapsto \text{conn}, \\
& \text{target} \mapsto \text{target}] \\
\\
& \text{Mapping} \triangleq \text{INSTANCE Config WITH} \\
& \quad \text{proposal} \leftarrow \text{Alias.proposal}, \\
& \quad \text{configuration} \leftarrow \text{Alias.configuration} \\
\\
& \text{Refinement} \triangleq \text{Mapping!Spec} \\
\\
& \text{Order} \triangleq \text{Mapping!Order} \\
\\
& \text{Consistency} \triangleq \text{Mapping!Consistency} \\
\\
& \text{Liveness} \triangleq \text{Mapping!Liveness}
\end{aligned}$$


---