

---

MODULE *Config*

---

INSTANCE *Naturals*

INSTANCE *FiniteSets*

INSTANCE *Sequences*

INSTANCE *TLC*

---

*GenerateTestCases*  $\triangleq$  TRUE

*Nil*  $\triangleq$  "<nil>"

*Change*  $\triangleq$  "Change"

*Rollback*  $\triangleq$  "Rollback"

*Commit*  $\triangleq$  "Commit"

*Apply*  $\triangleq$  "Apply"

*Pending*  $\triangleq$  "Pending"

*InProgress*  $\triangleq$  "InProgress"

*Complete*  $\triangleq$  "Complete"

*Aborted*  $\triangleq$  "Aborted"

*Failed*  $\triangleq$  "Failed"

*Done*  $\triangleq$  { *Complete*, *Aborted*, *Failed* }

*Node*  $\triangleq$  { "node1" }

*NumTransactions*  $\triangleq$  4

*NumTerms*  $\triangleq$  2

*NumConns*  $\triangleq$  2

*NumStarts*  $\triangleq$  2

*Path*  $\triangleq$  { "path1" }

*Value*  $\triangleq$  { "value1", "value2" }

---

A transaction log. Transactions may either request a set of changes to a set of targets or rollback a prior change.

VARIABLE *transaction*

A record of per-target proposals

VARIABLE *proposal*

A record of per-target configurations

VARIABLE *configuration*

A record of target masterships  
VARIABLE *mastership*

A record of node connections to the target  
VARIABLE *conn*

The target state  
VARIABLE *target*

A sequence of state changes used for model checking.  
VARIABLE *history*

$vars \triangleq \langle transaction, proposal, configuration, mastership, conn, target, history \rangle$

---

LOCAL *Transaction*  $\triangleq$  INSTANCE *Transaction*

LOCAL *Proposal*  $\triangleq$  INSTANCE *Proposal*

LOCAL *Configuration*  $\triangleq$  INSTANCE *Configuration*

LOCAL *Mastership*  $\triangleq$  INSTANCE *Mastership*

LOCAL *Target*  $\triangleq$  INSTANCE *Target*

---

*RequestChange*(*p*, *v*)  $\triangleq$   
 $\wedge$  *Transaction*! *RequestChange*(*p*, *v*)  
 $\wedge$  UNCHANGED  $\langle mastership, conn, target, history \rangle$

*RequestRollback*(*i*)  $\triangleq$   
 $\wedge$  *Transaction*! *RequestRollback*(*i*)  
 $\wedge$  UNCHANGED  $\langle mastership, conn, target, history \rangle$

*ReconcileTransaction*(*n*, *i*)  $\triangleq$   
 $\wedge$  *i*  $\in$  DOMAIN *transaction*  
 $\wedge$  *Transaction*! *ReconcileTransaction*(*n*, *i*)  
 $\wedge$  UNCHANGED  $\langle mastership, conn, target, history \rangle$   
 $\wedge$  *GenerateTestCases*  $\Rightarrow$  *Transaction*! *Test*! *Log*([*node*  $\mapsto$  *n*, *index*  $\mapsto$  *i*])

*ReconcileProposal*(*n*, *i*)  $\triangleq$   
 $\wedge$  *i*  $\in$  DOMAIN *proposal*  
 $\wedge$  *Proposal*! *ReconcileProposal*(*n*, *i*)  
 $\wedge$  UNCHANGED  $\langle transaction \rangle$   
 $\wedge$  *GenerateTestCases*  $\Rightarrow$  *Proposal*! *Test*! *Log*([*node*  $\mapsto$  *n*, *index*  $\mapsto$  *i*])

*ReconcileConfiguration*(*n*)  $\triangleq$   
 $\wedge$  *Configuration*! *ReconcileConfiguration*(*n*)

$$\begin{aligned}
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{history} \rangle \\
& \wedge \text{GenerateTestCases} \Rightarrow \text{Configuration!Test!Log}([node \mapsto n]) \\
\text{ReconcileMastership}(n) & \triangleq \\
& \wedge \text{Mastership!ReconcileMastership}(n) \\
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{configuration}, \text{target}, \text{history} \rangle \\
& \wedge \text{GenerateTestCases} \Rightarrow \text{Mastership!Test!Log}([node \mapsto n]) \\
\text{ConnectNode}(n) & \triangleq \\
& \wedge \text{Target!Connect}(n) \\
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{configuration}, \text{mastership}, \text{history} \rangle \\
\text{DisconnectNode}(n) & \triangleq \\
& \wedge \text{Target!Disconnect}(n) \\
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{configuration}, \text{mastership}, \text{history} \rangle \\
\text{StartTarget} & \triangleq \\
& \wedge \text{Target!Start} \\
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{configuration}, \text{mastership}, \text{history} \rangle \\
\text{StopTarget} & \triangleq \\
& \wedge \text{Target!Stop} \\
& \wedge \text{UNCHANGED } \langle \text{transaction}, \text{proposal}, \text{configuration}, \text{mastership}, \text{history} \rangle
\end{aligned}$$


---

Formal specification, constraints, and theorems.

$$\begin{aligned}
\text{Init} & \triangleq \\
& \wedge \text{transaction} = [ \\
& \quad i \in \{\} \mapsto [ \\
& \quad \quad \text{type} \mapsto \text{Change}, \\
& \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \text{values} \mapsto [p \in \{\} \mapsto \text{Nil}], \\
& \quad \quad \text{commit} \mapsto \text{Pending}, \\
& \quad \quad \text{apply} \mapsto \text{Pending}] \\
& \wedge \text{proposal} = [ \\
& \quad i \in \{\} \mapsto [ \\
& \quad \quad \text{change} \mapsto [ \\
& \quad \quad \quad \text{phase} \mapsto \text{Nil}, \\
& \quad \quad \quad \text{state} \mapsto \text{Nil}, \\
& \quad \quad \quad \text{values} \mapsto [ \\
& \quad \quad \quad \quad p \in \{\} \mapsto [ \\
& \quad \quad \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \quad \quad \text{value} \mapsto \text{Nil}]]], \\
& \quad \text{rollback} \mapsto [ \\
& \quad \quad \text{phase} \mapsto \text{Nil}, \\
& \quad \quad \text{state} \mapsto \text{Nil},
\end{aligned}$$

$$\begin{aligned}
& \text{values} \mapsto [ \\
& \quad p \in \{\} \mapsto [ \\
& \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \text{value} \mapsto \text{Nil}]]]] \\
\wedge \text{configuration} = [ \\
& \quad \text{state} \mapsto \text{InProgress}, \\
& \quad \text{term} \mapsto 0, \\
& \quad \text{committed} \mapsto [ \\
& \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \text{revision} \mapsto 0, \\
& \quad \quad \text{values} \mapsto [ \\
& \quad \quad \quad p \in \{\} \mapsto [ \\
& \quad \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \quad \text{value} \mapsto \text{Nil}]]], \\
& \quad \text{applied} \mapsto [ \\
& \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \text{revision} \mapsto 0, \\
& \quad \quad \text{target} \mapsto 0, \\
& \quad \quad \text{values} \mapsto [ \\
& \quad \quad \quad p \in \{\} \mapsto [ \\
& \quad \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \quad \text{value} \mapsto \text{Nil}]]]] \\
\wedge \text{target} = [ \\
& \quad \text{id} \mapsto 0, \\
& \quad \text{running} \mapsto \text{FALSE}, \\
& \quad \text{values} \mapsto [ \\
& \quad \quad p \in \{\} \mapsto [ \\
& \quad \quad \quad \text{index} \mapsto 0, \\
& \quad \quad \quad \text{value} \mapsto \text{Nil}]] \\
\wedge \text{mastership} = [ \\
& \quad \text{master} \mapsto \text{Nil}, \\
& \quad \text{term} \mapsto 0, \\
& \quad \text{conn} \mapsto 0] \\
\wedge \text{conn} = [ \\
& \quad n \in \text{Node} \mapsto [ \\
& \quad \quad \text{id} \mapsto 0, \\
& \quad \quad \text{connected} \mapsto \text{FALSE}] \\
\wedge \text{history} = \langle \rangle \\
\text{Next} \triangleq \\
& \vee \exists p \in \text{Path}, v \in \text{Value} : \\
& \quad \text{RequestChange}(p, v) \\
& \vee \exists i \in \text{DOMAIN transaction} : \\
& \quad \text{RequestRollback}(i) \\
& \vee \exists n \in \text{Node} :
\end{aligned}$$

$$\begin{aligned}
& \exists i \in \text{DOMAIN } transaction : \\
& \quad ReconcileTransaction(n, i) \\
\vee \exists n \in Node : \\
& \quad \exists i \in \text{DOMAIN } proposal : \\
& \quad \quad ReconcileProposal(n, i) \\
\vee \exists n \in Node : \\
& \quad ReconcileConfiguration(n) \\
\vee \exists n \in Node : \\
& \quad ReconcileMastership(n) \\
\vee \exists n \in Node : \\
& \quad \vee ConnectNode(n) \\
& \quad \vee DisconnectNode(n) \\
\vee StartTarget \\
\vee StopTarget \\
Spec & \triangleq \\
& \wedge Init \\
& \wedge \Box[Next]_{vars} \\
& \wedge \forall p \in Path, v \in Value : \\
& \quad WF_{\langle transaction, proposal, configuration, mastership, target \rangle}(Transaction!RequestChange(p, v)) \\
& \wedge \forall i \in 1 \dots NumTransactions : i \in \text{DOMAIN } transaction \Rightarrow \\
& \quad WF_{\langle transaction, proposal, configuration, mastership, target \rangle}(Transaction!RequestRollback(i)) \\
& \wedge \forall n \in Node, i \in 1 \dots NumTransactions : \\
& \quad WF_{\langle transaction, proposal, configuration, mastership, target \rangle}(Transaction!ReconcileTransaction(n, i)) \\
& \wedge \forall n \in Node, i \in 1 \dots NumTransactions : \\
& \quad WF_{\langle proposal, configuration, mastership, conn, target, history \rangle}(Proposal!ReconcileProposal(n, i)) \\
& \wedge \forall n \in Node : \\
& \quad WF_{\langle configuration, mastership, conn, target \rangle}(Configuration!ReconcileConfiguration(n)) \\
& \wedge \forall n \in Node : \\
& \quad WF_{\langle mastership, conn \rangle}(Mastership!ReconcileMastership(n)) \\
& \wedge \forall n \in Node : \\
& \quad WF_{\langle conn, target \rangle}(Target!Connect(n) \vee Target!Disconnect(n)) \\
& \wedge WF_{\langle conn, target \rangle}(Target!Start \vee Target!Stop)
\end{aligned}$$

---


$$LimitTransactions \triangleq Len(transaction) \leq NumTransactions$$

$$\begin{aligned}
LimitTerms & \triangleq \\
& \vee mastership.term < NumTerms \\
& \vee \wedge mastership.term = NumTerms \\
& \wedge mastership.master \neq Nil
\end{aligned}$$

$$\begin{aligned}
LimitConns & \triangleq \\
& \forall n \in \text{DOMAIN } conn : \\
& \quad \vee conn[n].id < NumConns
\end{aligned}$$

$$\begin{aligned} & \vee \wedge \text{conn}[n].id = \text{NumConns} \\ & \wedge \text{conn}[n].connected \end{aligned}$$

$$\begin{aligned} \text{LimitStarts} & \triangleq \\ & \vee \text{target}.id < 2 \\ & \vee \wedge \text{target}.id = 2 \\ & \wedge \text{target}.running \end{aligned}$$

---


$$\begin{aligned} \text{TypeOK} & \triangleq \\ & \wedge \text{Transaction! TypeOK} \\ & \wedge \text{Proposal! TypeOK} \\ & \wedge \text{Configuration! TypeOK} \\ & \wedge \text{Mastership! TypeOK} \end{aligned}$$

$$\begin{aligned} \text{LOCAL } \text{IsOrderedChange}(p, i) & \triangleq \\ & \wedge \text{history}[i].type = \text{Change} \\ & \wedge \text{history}[i].phase = p \\ & \wedge \neg \exists j \in \text{DOMAIN } \text{history} : \\ & \quad \wedge j < i \\ & \quad \wedge \text{history}[j].type = \text{Change} \\ & \quad \wedge \text{history}[j].phase = p \\ & \quad \wedge \text{history}[j].index \geq \text{history}[i].index \end{aligned}$$

$$\begin{aligned} \text{LOCAL } \text{IsOrderedRollback}(p, i) & \triangleq \\ & \wedge \text{history}[i].type = \text{Rollback} \\ & \wedge \text{history}[i].phase = p \\ & \wedge \neg \exists j \in \text{DOMAIN } \text{history} : \\ & \quad \wedge j < i \\ & \quad \wedge \text{history}[j].type = \text{Change} \\ & \quad \wedge \text{history}[j].phase = p \\ & \quad \wedge \text{history}[j].index > \text{history}[i].index \\ & \wedge \neg \exists k \in \text{DOMAIN } \text{history} : \\ & \quad \wedge k > j \\ & \quad \wedge k < i \\ & \quad \wedge \text{history}[k].type = \text{Rollback} \\ & \quad \wedge \text{history}[k].phase = p \\ & \quad \wedge \text{history}[k].index = \text{history}[j].index \end{aligned}$$

$$\begin{aligned} \text{Order} & \triangleq \\ & \wedge \forall i \in \text{DOMAIN } \text{history} : \\ & \quad \vee \text{IsOrderedChange}(\text{Commit}, i) \\ & \quad \vee \text{IsOrderedChange}(\text{Apply}, i) \\ & \quad \vee \text{IsOrderedRollback}(\text{Commit}, i) \\ & \quad \vee \text{IsOrderedRollback}(\text{Apply}, i) \\ & \wedge \forall i \in \text{DOMAIN } \text{proposal} : \end{aligned}$$

$$\begin{aligned}
& \wedge \text{proposal}[i].\text{change.phase} = \text{Apply} \\
& \wedge \text{proposal}[i].\text{change.state} = \text{Failed} \\
& \wedge \text{proposal}[i].\text{rollback.phase} = \text{Apply} \Rightarrow \text{proposal}[i].\text{rollback.state} \neq \text{Complete} \\
& \Rightarrow \forall j \in \text{DOMAIN } \text{proposal} : (j > i \Rightarrow \\
& \quad (\text{proposal}[j].\text{change.phase} = \text{Apply} \Rightarrow \\
& \quad \text{proposal}[j].\text{change.state} \in \{\text{Nil}, \text{Pending}, \text{Aborted}\}))
\end{aligned}$$

*Consistency*  $\triangleq$

$$\begin{aligned}
& \wedge \forall i \in \text{DOMAIN } \text{proposal} : \\
& \quad \vee \text{configuration.committed.index} < i \\
& \quad \vee \text{configuration.committed.revision} < i \\
& \quad \Rightarrow \neg \exists p \in \text{DOMAIN } \text{configuration.committed.values} : \\
& \quad \quad \text{configuration.committed.values}[p].\text{index} = i \\
& \wedge \forall i \in \text{DOMAIN } \text{proposal} : \\
& \quad \vee \text{configuration.applied.index} < i \\
& \quad \vee \text{configuration.applied.revision} < i \\
& \quad \Rightarrow \wedge \neg \exists p \in \text{DOMAIN } \text{configuration.applied.values} : \\
& \quad \quad \text{configuration.applied.values}[p].\text{index} = i \\
& \quad \wedge \neg \exists p \in \text{DOMAIN } \text{target.values} : \\
& \quad \quad \text{target.values}[p].\text{index} = i \\
& \wedge \wedge \text{target.running} \\
& \wedge \text{configuration.applied.target} = \text{target.id} \\
& \wedge \text{configuration.state} = \text{Complete} \\
& \Rightarrow \forall i \in \text{DOMAIN } \text{proposal} : \\
& \quad \wedge \text{configuration.applied.index} \geq i \\
& \quad \wedge \text{configuration.applied.revision} \geq i \\
& \quad \Rightarrow \forall p \in \text{DOMAIN } \text{proposal}[i].\text{change.values} : \\
& \quad \quad \wedge \neg \exists j \in \text{DOMAIN } \text{proposal} : \\
& \quad \quad \quad \wedge j > i \\
& \quad \quad \quad \wedge \text{configuration.applied.index} \geq j \\
& \quad \quad \quad \wedge \text{configuration.applied.revision} \geq j \\
& \quad \Rightarrow \wedge p \in \text{DOMAIN } \text{target.values} \\
& \quad \quad \wedge \text{target.values}[p].\text{value} = \text{proposal}[i].\text{change.values}[p].\text{value} \\
& \quad \quad \wedge \text{target.values}[p].\text{index} = \text{proposal}[i].\text{change.values}[p].\text{index}
\end{aligned}$$

*Safety*  $\triangleq \Box(\text{Order} \wedge \text{Consistency})$

THEOREM  $\text{Spec} \Rightarrow \text{Safety}$

*Terminates*(*i*)  $\triangleq$

$$\begin{aligned}
& \wedge i \in \text{DOMAIN } \text{transaction} \\
& \wedge \text{transaction}[i].\text{commit} \in \text{Done} \\
& \wedge \text{transaction}[i].\text{apply} \in \text{Done} \\
& \wedge \text{transaction}[i].\text{index} \in \text{DOMAIN } \text{proposal} \\
& \wedge \vee \wedge \text{transaction}[i].\text{type} = \text{Change} \\
& \quad \wedge \vee \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{change.phase} = \text{Commit}
\end{aligned}$$

$$\begin{aligned}
& \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{change.state} \in \{\text{Aborted}, \text{Failed}\} \\
& \vee \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{change.phase} = \text{Apply} \\
& \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{change.state} \in \text{Done} \\
& \vee \wedge \text{transaction}[i].\text{type} = \text{Rollback} \\
& \wedge \vee \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{rollback.phase} = \text{Commit} \\
& \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{rollback.state} \in \{\text{Aborted}, \text{Failed}\} \\
& \vee \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{rollback.phase} = \text{Apply} \\
& \wedge \text{proposal}[\text{transaction}[i].\text{index}].\text{rollback.state} \in \text{Done}
\end{aligned}$$

$$\begin{aligned}
\text{Termination} & \triangleq \\
& \forall i \in 1 \dots \text{NumTransactions} : \Diamond \text{Terminates}(i)
\end{aligned}$$

$$\text{Liveness} \triangleq \text{Termination}$$

THEOREM  $\text{Spec} \Rightarrow \text{Liveness}$