
MODULE *Transaction*

INSTANCE *Naturals*

INSTANCE *FiniteSets*

INSTANCE *Sequences*

INSTANCE *TLC*

An empty constant

CONSTANT *Nil*

Transaction phase constants

CONSTANTS

Change,

Rollback

Transaction phase constants

CONSTANTS

Commit,

Apply

Status constants

CONSTANTS

Pending,

InProgress,

Complete,

Aborted,

Canceled,

Failed

$Status \triangleq \{Pending, InProgress, Complete, Aborted, Canceled, Failed\}$

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

The set of all nodes

CONSTANT *Node*

The set of possible paths and values

CONSTANT *Path, Value*

$Empty \triangleq [p \in \{\} \mapsto Nil]$

Variables defined by other modules.

VARIABLES

configuration,
mastership,
conn,
target

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

VARIABLE *transactions*

A history of transaction change/rollback commit/apply events used for model checking.

VARIABLE *history*

TypeOK \triangleq

$\forall i \in \text{DOMAIN } transactions :$
 $\wedge transactions[i].index \in Nat$
 $\wedge transactions[i].phase \in \{Change, Rollback\}$
 $\wedge transactions[i].change.commit \in Status$
 $\wedge transactions[i].change.apply \in Status$
 $\wedge \forall p \in \text{DOMAIN } transactions[i].change.values :$
 $\quad transactions[i].change.values[p] \neq Nil \Rightarrow$
 $\quad \quad transactions[i].change.values[p] \in STRING$
 $\wedge transactions[i].rollback.commit \neq Nil \Rightarrow$
 $\quad transactions[i].rollback.commit \in Status$
 $\wedge transactions[i].rollback.apply \neq Nil \Rightarrow$
 $\quad transactions[i].rollback.apply \in Status$
 $\wedge \forall p \in \text{DOMAIN } transactions[i].rollback.values :$
 $\quad transactions[i].rollback.values[p] \neq Nil \Rightarrow$
 $\quad \quad transactions[i].rollback.values[p] \in STRING$

LOCAL *State* \triangleq [
 transactions $\mapsto transactions$,
 configuration $\mapsto configuration$,
 mastership $\mapsto mastership$,
 conn $\mapsto conn$,
 target $\mapsto target$]

LOCAL *Transitions* \triangleq

LET
 indexes $\triangleq \{i \in \text{DOMAIN } transactions' : i \in \text{DOMAIN } transactions \Rightarrow transactions'[i] \neq transactions[i]\}$
 IN [*transactions* $\mapsto [i \in indexes \mapsto transactions'[i]]$] @@
 (IF *configuration'* $\neq configuration$ THEN [*configuration* $\mapsto configuration'$] ELSE *Empty*) @@
 (IF *target'* $\neq target$ THEN [*target* $\mapsto target'$] ELSE *Empty*)

Test \triangleq INSTANCE *Test* WITH
 File \leftarrow "Transaction.log"

$$\begin{aligned}
\text{CommitChange}(n, i) \triangleq & \\
& \wedge \vee \wedge \text{transactions}[i].\text{change.commit} = \text{Pending} \\
& \wedge \text{configuration.committed.maxIndex} = i - 1 \\
& \wedge \vee \wedge \text{configuration.committed.target} \neq i \\
& \quad \wedge \text{configuration.committed.index} = \text{configuration.committed.target} \\
& \quad \wedge \text{configuration.committed.transaction} \in \text{DOMAIN transactions} \Rightarrow \\
& \quad \quad \vee \wedge \text{configuration.committed.target} = \text{configuration.committed.transaction} \\
& \quad \quad \quad \wedge \text{transactions}[\text{configuration.committed.transaction}].\text{change.commit} \in \text{Done} \\
& \quad \quad \vee \wedge \text{configuration.committed.target} < \text{configuration.committed.transaction} \\
& \quad \quad \quad \wedge \text{transactions}[\text{configuration.committed.transaction}].\text{rollback.commit} \in \text{Done} \\
& \wedge \text{configuration}' = [\text{configuration EXCEPT } !.\text{committed.target} = i] \\
& \wedge \text{history}' = \text{Append}(\text{history}, [\\
& \quad \quad \text{type} \mapsto \text{Change}, \\
& \quad \quad \text{phase} \mapsto \text{Commit}, \\
& \quad \quad \text{index} \mapsto i, \\
& \quad \quad \text{status} \mapsto \text{InProgress}]) \\
& \wedge \text{UNCHANGED } \langle \text{transactions} \rangle \\
& \vee \wedge \text{configuration.committed.target} = i \\
& \quad \wedge \text{transactions}' = [\text{transactions EXCEPT } ! [i].\text{change.commit} = \text{InProgress}, \\
& \quad \quad \quad ! [i].\text{rollback.index} = \text{configuration.committed.revision}, \\
& \quad \quad \quad ! [i].\text{rollback.values} = [\\
& \quad \quad \quad \quad p \in \text{DOMAIN transactions}[i].\text{change.values} \mapsto \\
& \quad \quad \quad \quad \text{IF } p \in \text{DOMAIN configuration.committed.values THEN} \\
& \quad \quad \quad \quad \quad \text{configuration.committed.values}[p] \\
& \quad \quad \quad \quad \text{ELSE Nil}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \vee \wedge \text{transactions}[i].\text{change.commit} = \text{InProgress} \\
& \quad \wedge \vee \wedge \text{configuration.committed.index} \neq i \\
& \quad \quad \wedge \vee \wedge \text{configuration}' = [\text{configuration EXCEPT } !.\text{committed.transaction} = i, \\
& \quad \quad \quad !.\text{committed.index} = i, \\
& \quad \quad \quad !.\text{committed.maxIndex} = i, \\
& \quad \quad \quad !.\text{committed.revision} = i, \\
& \quad \quad \quad !.\text{committed.seqnum} = \text{configuration.commit}, \\
& \quad \quad \quad !.\text{committed.values} = \text{transactions}[i].\text{change.values}, \\
& \quad \quad \quad \text{configuration.committed.target} = i, \\
& \quad \quad \quad \text{configuration.committed.index} = i, \\
& \quad \quad \quad \text{configuration.committed.maxIndex} = i, \\
& \quad \quad \quad \text{configuration.committed.revision} = i, \\
& \quad \quad \quad \text{configuration.committed.seqnum} = \text{configuration.commit}, \\
& \quad \quad \quad \text{configuration.committed.values} = \text{transactions}[i].\text{change.values}] \\
& \quad \quad \wedge \text{history}' = \text{Append}(\text{history}, [\\
& \quad \quad \quad \text{type} \mapsto \text{Change}, \\
& \quad \quad \quad \text{phase} \mapsto \text{Commit}, \\
& \quad \quad \quad \text{index} \mapsto i, \\
& \quad \quad \quad \text{status} \mapsto \text{Complete}]) \\
& \quad \vee \wedge \text{configuration}' = [\text{configuration EXCEPT } !.\text{committed.transaction} = i, \\
& \quad \quad \quad !.\text{committed.index} = i, \\
& \quad \quad \quad !.\text{committed.maxIndex} = i] \\
& \quad \quad \wedge \text{history}' = \text{Append}(\text{history}, [\\
& \quad \quad \quad \text{type} \mapsto \text{Change},
\end{aligned}$$

$$\begin{aligned}
& \text{index} \mapsto i, \\
& \text{status} \mapsto \text{Complete}] \\
& \wedge \text{UNCHANGED } \langle \text{transactions} \rangle \\
& \vee \wedge \text{configuration.committed.revision} = \text{transactions}[i].\text{rollback.index} \\
& \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{Complete}, \\
& \hspace{15em} ![i].\text{rollback.seqnum} = \text{configuration.committed.seqnum}] \\
& \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn}, \text{target} \rangle \\
& - \text{configuration} \\
& - \text{applied} \\
& \quad - \text{transaction: the index of the last applied transaction} \\
& \quad - \text{seqnum: the last applied sequence number} \\
& \quad - \text{target: the current target revision (index)} \\
& \quad - \text{revision: the last successfully applied revision} \\
& * \text{ If } \text{target} = \text{transaction} \text{ then the current transaction is being changed} \\
& * \text{ If } \text{target} < \text{transaction} \text{ then the current transaction is being rolled back} \\
& \text{ApplyChange}(n, i) \triangleq \\
& \quad \wedge \text{transactions}[i].\text{change.commit} = \text{Complete} \\
& \quad \wedge \vee \wedge \text{transactions}[i].\text{change.apply} = \text{Pending} \\
& \quad \wedge \vee \wedge \text{configuration.applied.seqnum} = \text{transactions}[i].\text{change.seqnum} - 1 \\
& \quad \wedge \vee \wedge \text{configuration.applied.target} \neq i \\
& \quad \wedge \text{configuration.applied.transaction} \in \text{DOMAIN } \text{transactions} \Rightarrow \\
& \quad \quad \vee \wedge \text{configuration.applied.target} = \text{configuration.applied.transaction} \\
& \quad \quad \wedge \text{transactions}[\text{configuration.applied.transaction}].\text{change.apply} \in \text{Done} \\
& \quad \quad \vee \wedge \text{configuration.applied.target} < \text{configuration.applied.transaction} \\
& \quad \quad \wedge \text{transactions}[\text{configuration.applied.transaction}].\text{rollback.apply} \in \text{Done} \\
& \quad \wedge \vee \wedge \text{configuration.applied.revision} = \text{transactions}[i].\text{rollback.index} \\
& \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.target} = i, \\
& \hspace{15em} !.\text{applied.index} = i] \\
& \quad \wedge \text{history}' = \text{Append}(\text{history}, [\\
& \hspace{4em} \text{type} \mapsto \text{Change}, \\
& \hspace{4em} \text{phase} \mapsto \text{Apply}, \\
& \hspace{4em} \text{index} \mapsto i, \\
& \hspace{4em} \text{status} \mapsto \text{InProgress}]) \\
& \quad \vee \wedge \vee \text{transactions}[i].\text{phase} = \text{Rollback} \\
& \quad \vee \text{configuration.applied.revision} < \text{transactions}[i].\text{rollback.index} \\
& \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.target} = i, \\
& \hspace{15em} !.\text{applied.transaction} = i, \\
& \hspace{15em} !.\text{applied.seqnum} = \text{transactions}[i].\text{change.seqnum}] \\
& \quad \wedge \text{history}' = \text{Append}(\text{history}, [\\
& \hspace{4em} \text{type} \mapsto \text{Change}, \\
& \hspace{4em} \text{phase} \mapsto \text{Apply}, \\
& \hspace{4em} \text{index} \mapsto i, \\
& \hspace{4em} \text{status} \mapsto \text{Aborted}])
\end{aligned}$$

$$\begin{aligned}
& \text{type} \mapsto \text{Rollback}, \\
& \text{phase} \mapsto \text{Apply}, \\
& \text{index} \mapsto i, \\
& \text{status} \mapsto \text{Complete}] \\
& \wedge \text{UNCHANGED } \langle \text{transactions} \rangle \\
& \text{If the change has been applied, update the transaction status.} \\
& \vee \wedge \text{configuration.applied.seqnum} = \text{transactions}[i].\text{rollback.seqnum} \\
& \wedge \text{configuration.applied.revision} = \text{transactions}[i].\text{rollback.index} \\
& \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.apply} = \text{Complete}] \\
& \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
& \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn} \rangle \\
\text{ReconcileTransaction}(n, i) & \triangleq \\
& \wedge i \in \text{DOMAIN } \text{transactions} \\
& \wedge \text{mastership.master} = n \\
& \wedge \vee \text{CommitChange}(n, i) \\
& \quad \vee \text{ApplyChange}(n, i) \\
& \quad \vee \text{CommitRollback}(n, i) \\
& \quad \vee \text{ApplyRollback}(n, i)
\end{aligned}$$

$$\begin{aligned}
\text{CommitChange}(n, i) & \triangleq \\
& \wedge \vee \wedge \text{transactions}[i].\text{change.commit} = \text{Pending} \\
& \quad \wedge \forall j \in \text{DOMAIN } \text{transactions} : j < i \Rightarrow \\
& \quad \quad \wedge \text{transactions}[j].\text{change.commit} \in \text{Done} \\
& \quad \quad \wedge \text{transactions}[j].\text{rollback.commit} \neq \text{InProgress} \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.commit} = \text{InProgress}, \\
& \quad \quad \quad ![i].\text{rollback.values} = [\\
& \quad \quad \quad \quad p \in \text{DOMAIN } \text{transactions}[i].\text{change.values} \mapsto \\
& \quad \quad \quad \quad \text{IF } p \in \text{DOMAIN } \text{configuration.committed.values} \text{ THEN} \\
& \quad \quad \quad \quad \quad \text{configuration.committed.value}[p] \\
& \quad \quad \quad \quad \text{ELSE Nil}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \vee \wedge \text{transactions}[i].\text{change.commit} = \text{InProgress} \\
& \quad \backslash * \text{Changes are validated during the commit phase. If a change fails validation,} \\
& \quad \backslash * \text{it will be marked failed before being applied to the configuration.} \\
& \quad \backslash * \text{If all the change values are valid, record the changes required to roll} \\
& \quad \backslash * \text{back the transactions and the index to which the rollback changes} \\
& \quad \backslash * \text{will roll back the configuration.} \quad \wedge \vee \wedge \text{configuration}' = [\text{configuration} \\
& \quad \text{EXCEPT } !.\text{committed.values} = \text{transactions}[i].\text{change.values} @@ \\
& \quad \quad \quad \text{configuration.committed.values}] \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.commit} = \text{Complete}] \\
& \quad \wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Change}, \text{phase} \mapsto \text{Commit}, \text{index} \mapsto i]) \\
& \quad \vee \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.commit} = \text{Failed}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn}, \text{target} \rangle \\
\text{CommitRollback}(n, i) & \triangleq
\end{aligned}$$

$$\begin{aligned}
& \wedge \vee \wedge \text{transactions}[i].\text{rollback.commit} = \text{Pending} \\
& \wedge \forall j \in \text{DOMAIN } \text{transactions} : \\
& \quad \wedge j > i \\
& \quad \wedge \text{transactions}[j].\text{phase} \neq \text{Nil} \\
& \quad \wedge \text{transactions}[j].\text{change.commit} \neq \text{Pending} \\
& \quad \Rightarrow \text{transactions}[j].\text{rollback.commit} = \text{Complete} \\
& \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{InProgress}] \\
& \wedge \text{UNCHANGED } \langle \text{configuration}, \text{history} \rangle \\
& \vee \wedge \text{transactions}[i].\text{rollback.commit} = \text{InProgress} \\
& \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{committed.values} = \text{transactions}[i].\text{rollback.values} @@ \text{configuration.committed.values}] \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.commit} = \text{Complete}] \\
& \quad \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{mastership}, \text{conn}, \text{target} \rangle \\
& \text{ApplyChange}(n, i) \triangleq \\
& \quad \wedge \vee \wedge \text{transactions}[i].\text{change.apply} = \text{Pending} \\
& \quad \wedge \vee \wedge \text{transactions}[i].\text{change.commit} = \text{Complete} \\
& \quad \wedge \forall j \in \text{DOMAIN } \text{transactions} : j < i \Rightarrow \\
& \quad \quad \vee \wedge \text{transactions}[j].\text{change.apply} = \text{Complete} \\
& \quad \quad \wedge \text{transactions}[j].\text{rollback.apply} \neq \text{InProgress} \\
& \quad \quad \vee \wedge \text{transactions}[j].\text{change.apply} = \text{Failed} \\
& \quad \quad \wedge \text{transactions}[j].\text{rollback.apply} = \text{Complete} \\
& \quad \wedge i - 1 \in \text{DOMAIN } \text{transactions} \wedge \text{transactions}[i - 1].\text{change.apply} = \text{Failed} \Rightarrow \\
& \quad \quad \text{transactions}[i - 1].\text{rollback.apply} = \text{Complete} \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.apply} = \text{InProgress}] \\
& \quad \vee \wedge \text{transactions}[i].\text{change.commit} = \text{Failed} \\
& \quad \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.apply} = \text{Aborted}] \\
& \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
& \quad \vee \wedge \text{transactions}[i].\text{change.apply} = \text{InProgress} \\
& \quad \quad \backslash * \text{Verify the applied term is the current } \text{mastership} \text{ term to ensure the} \\
& \quad \quad \backslash * \text{configuration has been synchronized following restarts.} \\
& \quad \quad \wedge \text{configuration.applied.term} = \text{mastership.term} \\
& \quad \quad \backslash * \text{Verify the node's connection to the target.} \\
& \quad \quad \wedge \text{conn}[n].\text{connected} \\
& \quad \quad \wedge \text{mastership.conn} = \text{conn}[n].\text{id} \\
& \quad \quad \wedge \text{target.running} \\
& \quad \quad \backslash * \text{Model successful and failed target update requests.} \\
& \quad \quad \wedge \vee \wedge \text{target}' = [\text{target} \text{ EXCEPT } !.\text{values} = \text{transactions}[i].\text{change.values} @@ \text{target.values}] \\
& \quad \quad \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.values} = \text{transactions}[i].\text{change.values} @@ \text{configuration.applied.values}] \\
& \quad \quad \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.apply} = \text{Complete}] \\
& \quad \quad \quad \wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Change}, \text{phase} \mapsto \text{Apply}, \text{index} \mapsto i]) \\
& \quad \quad \vee \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.apply} = \text{Failed}] \\
& \quad \quad \quad \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
& \quad \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn} \rangle \\
& \text{ApplyRollback}(n, i) \triangleq \\
& \quad \wedge \vee \wedge \text{transactions}[i].\text{rollback.apply} = \text{Pending}
\end{aligned}$$

$$\begin{aligned}
& \wedge \text{transactions}[i].\text{rollback.commit} = \text{Complete} \\
& \wedge \forall j \in \text{DOMAIN } \text{transactions} : \\
& \quad \wedge j > i \\
& \quad \wedge \text{transactions}[j].\text{phase} \neq \text{Nil} \\
& \quad \wedge \text{transactions}[j].\text{change.apply} \neq \text{Pending} \\
& \quad \Rightarrow \text{transactions}[j].\text{rollback.apply} \in \text{Done} \\
& \wedge \vee \wedge \text{transactions}[i].\text{change.apply} = \text{Pending} \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{change.apply} = \text{Aborted}, \\
& \quad \quad \quad ![i].\text{rollback.apply} = \text{Complete}] \\
& \vee \wedge \text{transactions}[i].\text{change.apply} \in \text{Done} \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.apply} = \text{InProgress}] \\
& \wedge \text{UNCHANGED } \langle \text{configuration}, \text{target}, \text{history} \rangle \\
& \vee \wedge \text{transactions}[i].\text{rollback.apply} = \text{InProgress} \\
& \quad \backslash * \text{ Verify the applied term is the current } \text{mastership} \text{ term to ensure the} \\
& \quad \backslash * \text{ configuration has been synchronized following restarts.} \\
& \quad \wedge \text{configuration.applied.term} = \text{mastership.term} \\
& \quad \backslash * \text{ Verify the node's connection to the target.} \\
& \quad \wedge \text{conn}[n].\text{connected} \\
& \quad \wedge \text{target.running} \\
& \quad \wedge \text{target}' = [\text{target} \text{ EXCEPT } !.\text{values} = \text{transactions}[i].\text{rollback.values} @@ \text{target.values}] \\
& \quad \wedge \text{configuration}' = [\text{configuration} \text{ EXCEPT } !.\text{applied.values} = \text{transactions}[i].\text{rollback.values} @@ \text{configuration.applied.values}] \\
& \quad \wedge \text{transactions}' = [\text{transactions} \text{ EXCEPT } ![i].\text{rollback.apply} = \text{Complete}] \\
& \quad \wedge \text{history}' = \text{Append}(\text{history}, [\text{type} \mapsto \text{Rollback}, \text{phase} \mapsto \text{Apply}, \text{index} \mapsto i]) \\
& \wedge \text{UNCHANGED } \langle \text{mastership}, \text{conn} \rangle \\
& \text{ReconcileChange}(n, i) \triangleq \\
& \quad \wedge \text{transactions}[i].\text{phase} = \text{Change} \\
& \quad \wedge \vee \text{CommitChange}(n, i) \\
& \quad \vee \text{ApplyChange}(n, i) \\
& \text{ReconcileRollback}(n, i) \triangleq \\
& \quad \wedge \text{transactions}[i].\text{phase} = \text{Rollback} \\
& \quad \wedge \vee \text{CommitRollback}(n, i) \\
& \quad \vee \text{ApplyRollback}(n, i) \\
& \text{ReconcileTransaction}(n, i) \triangleq \\
& \quad \wedge i \in \text{DOMAIN } \text{transactions} \\
& \quad \wedge \text{mastership.master} = n \\
& \quad \wedge \vee \text{ReconcileChange}(n, i) \\
& \quad \vee \text{ReconcileRollback}(n, i)
\end{aligned}$$

TODO:

- Change :
- if $\text{configuration.committed.index} = i - 1$ and associated transaction change commit is done
 - increment $\text{configuration.committed.index}$
 - change transaction commit status to *InProgress* and store rollback revision and *values*
 - if $\text{configuration.committed.index} = i$ then validate and commit *change*
 - increment $\text{configuration.committed.revision}$ to *i* if valid

Rollback :

- if *configuration.committed.revision* = *i*
- set *configuration.committed.index* to *i* and change rollback commit status to *InProgress*
- change *configuration.committed.values* to rollback values and revision to rollback revision
- if *configuration.committed.index* = *i* + 1 and associated transaction change commit failed or rollback is *committed*
- or if *configuration.committed.index* = *i* and transaction change is *committed*
- if *transactions[configuration.committed.index].commit* *i*
- decrement *configuration.committed.index*

TRANSACTION 1 [*index* = 1, *phase* = *Change*, *change* = (*commit* = *Pending*, *apply* = *Pending*), *rollback* = (*commit* = *Pending*, *apply* = *Pending*)]

CHANGE

commit = *Pending*

- ∨ ∧ *transactions[configuration.committed.index].phase* = *Change*
- ∧ *transactions[configuration.committed.index].change.commit* ∈ *Done*
- ∨ ∧ *transactions[configuration.committed.index].phase* = *Rollback*
- ∧ *transactions[configuration.committed.index].rollback.commit* ∈ *Done*

else if *transactions[configuration.committed.index].phase* = *Rollback*

apply = *Pending*

ROLLBACK

configuration \triangleq [
index \mapsto 0,
version \mapsto 0, \ * transaction version number
committed \mapsto [
index \mapsto 0, \ * committed transaction number
version \mapsto 0, \ * committed serial number
revision \mapsto 0, \ * committed revision number
values \mapsto [...]],
applied \mapsto [
index \mapsto 0, \ * applied transaction number
version \mapsto 0, \ * applied serial number
revision \mapsto 0, \ * applied revision number
values \mapsto [...]]]

TODO: Serialize transactions by managing the next transaction for changes and the previous transaction for rollbacks

When change is committed, set next transaction's change commit to *InProgress*

When rollback is committed, set previous transaction's rollback commit to *InProgress*

CHANGE [*index* = 1, *change* = (*prev* = 0), *rollback* = ()]

configuration.transaction = 1

configuration.revision = 1

CHANGE [*index* = 1, *change* = (*prev* = 0), *rollback* = ()]

CHANGE [*index* = 2, *change* = (*prev* = 1), *rollback* = ()]

configuration.transaction = 2

```

configuration.revision = 2

CHANGE [index = 1, change = (prev = 0), rollback = ()]
CHANGE [index = 2, change = (prev = 1), rollback = ()]
CHANGE [index = 3, change = (prev = 2), rollback = ()]

configuration.transaction = 3
configuration.revision = 3

CHANGE [index = 1, change = (prev = 0), rollback = ()]
CHANGE [index = 2, change = (prev = 1), rollback = ()]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]

configuration.transaction = 3
configuration.revision = 2

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]

configuration.transaction = 2
configuration.revision = 1

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)] CHANGE [index = 4,
change = (prev = 2), rollback = ()]

configuration.transaction = 4
configuration.revision = 4

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
CHANGE [index = 4, change = (prev = 2), rollback = ()]
CHANGE [index = 5, change = (prev = 4), rollback = ()]

configuration.transaction = 5
configuration.revision = 5

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = ()] CHANGE [index = 5, change =
(prev = 4), rollback = ()]

configuration.transaction = 5
configuration.revision = 5

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = (prev = 5)]
ROLLBACK [index = 5, change = (prev = 4), rollback = (prev = 5)]

configuration.transaction = 5
configuration.revision = 4

```

```

configuration.transaction = 4
configuration.revision = 1

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = (prev = 5)]
ROLLBACK [index = 5, change = (prev = 4), rollback = (prev = 5)]
CHANGE [index = 6, change = (prev = 2, commit = Complete, apply = Failed), rollback = ()]
CHANGE [index = 7, change = (prev = 6, commit = Complete, apply = Aborted), rollback =
()]
CHANGE [index = 8, change = (prev = 7, commit = Complete, apply = Aborted), rollback =
()]

configuration.transaction = 6
configuration.revision = 6

configuration.transaction = 7
configuration.revision = 7

configuration.transaction = 8
configuration.revision = 8

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = (prev = 5)]
ROLLBACK [index = 5, change = (prev = 4), rollback = (prev = 5)]
CHANGE [index = 6, change = (prev = 2, commit = Complete, apply = Failed), rollback = ()]
CHANGE [index = 7, change = (prev = 6, commit = Complete, apply = Aborted), rollback =
()]
ROLLBACK [index = 8, change = (prev = 7, commit = Complete, apply = Aborted), rollback =
(prev = 8)]

configuration.transaction = 8
configuration.revision = 7

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = (prev = 5)]
ROLLBACK [index = 5, change = (prev = 4), rollback = (prev = 5)] CHANGE [index = 6,
change = (prev = 2, commit = Complete, apply = Failed), rollback = ()]
ROLLBACK [index = 7, change = (prev = 6, commit = Complete, apply = Aborted), rollback =
(prev = 8)]
ROLLBACK [index = 8, change = (prev = 7, commit = Complete, apply = Aborted), rollback =
(prev = 8)]

configuration.transaction = 7
configuration.revision = 6

CHANGE [index = 1, change = (prev = 0), rollback = ()]
ROLLBACK [index = 2, change = (prev = 1), rollback = (prev = 3)]
ROLLBACK [index = 3, change = (prev = 2), rollback = (prev = 3)]
ROLLBACK [index = 4, change = (prev = 2), rollback = (prev = 5)]

```

```
ROLLBACK [index = 5, change = (prev = 4), rollback = (prev = 5)]
ROLLBACK [index = 6, change = (prev = 2, commit = Complete, apply = Failed), rollback =
(prev = 7)]
ROLLBACK [index = 7, change = (prev = 6, commit = Complete, apply = Aborted), rollback =
(prev = 8)]
ROLLBACK [index = 8, change = (prev = 7, commit = Complete, apply = Aborted), rollback =
(prev = 8)] CHANGE [index = 9, change = (prev = 6), rollback = ()]

configuration.transaction = 6
configuration.revision = 1
```
