MODULE $U2PC$

This models a a deployment of $U2PC$ with a reliable network (eventual and non-corrupt delivery).

Crash faults are not explicitly modelled, instead relying on asynchrony (both in nodes and the network) to provide equivalent executions. Thus an execution where a replica crashes is equivalent to one where that replica takes no further action.

EXTENDS $FiniteSets$, $Integers$, $Apalache$, TLC

@$typeAlias$: $key = Str$;
@$typeAlias$: rid $= Str$;
@$typeAlias$: $tid = Str$;
@$typeAlias$: $version =$ \$$tid$;
@$typeAlias$: $txn = Set($ \$$key)$;
@$typeAlias$: $txnstate =$ \$$key \to$ \$version;
$U2PC\_ALIAS \triangleq$ TRUE

CONSTANTS
@type: \$$key \to Set($ \$$rid)$;
$Shards$,
@type: \$$tid \to$ \$$txn$;
$Txns$

ASSUME "Init" $\notin$ DOMAIN $Txns$

Replicas are unique across shards.

In full implementations, if a server is a replica for multiple shards, it must have separate state for each shard.

ASSUME $\forall k1, k2 \in$ DOMAIN $Shards : k1 \neq k2 \Rightarrow Shards[k1] \cap Shards[k2] = \{\}$

$msg\_read = Str$;
$msg\_read\_resp = \{key :$ \$$key, ver :$ \$$version\}$;
$msg\_lock = \{txn :$ \$$txn, key :$ \$$key, ver :$ \$$version\}$;
$msg\_lock\_resp = Bool$;
$msg\_unlock = Bool$;
$msg\_unlock\_resp = Bool$;

VARIABLES
@type: \$rid $\to \{locked : Bool, version :$ \$$version, logged :$ \$$version\}$;
$Replicas$,
@type: \$$tid \to Str$;
$Coordinator\_state$,
@type: \$$tid \to$ \$txnstate;
$Coordinator\_txn\_state$,
@type: $Set(\{src :$ \$$tid, key :$ \$$key\})$;
M_read,
@type: $Set(\{src :$ \$$rid, dst :$ \$$tid, ver :$ \$$version\})$;

M_read_resp,
@type: $Set(\{tid : \$\, tid,\ txn : \$\, txn,\ state : \$\, txnstate\})$;
M_lock,
@type: $Set(\{src : \$\, rid,\ dst : \$\, tid,\ locked : Bool\})$;
M_lock_resp,
@type: $Set(\{src : \$\, tid,\ apply : Bool\})$;
M_unlock,
@type: $Set(\{src : \$\, rid,\ tid : \$\, tid\})$;
M_unlock_resp,
The set of transactions committed before the given transaction started.
NOTE: only used to check linearisability
@type: $\$tid \to Set(\$\, tid)$;
$Linearisability\_rt$

$$\begin{aligned} Vars \triangleq\ & \langle Replicas, \\ & Coordinator\_state,\ Coordinator\_txn\_state, \\ & M\_read,\ M\_read\_resp, \\ & M\_lock,\ M\_lock\_resp, \\ & M\_unlock,\ M\_unlock\_resp, \\ & Linearisability\_rt\rangle \end{aligned}$$

$$\begin{aligned} Var\_M\_read &\triangleq \langle M\_read,\ M\_read\_resp\rangle \\ Var\_M\_lock &\triangleq \langle M\_lock,\ M\_lock\_resp\rangle \\ Var\_M\_unlock &\triangleq \langle M\_unlock,\ M\_unlock\_resp\rangle \end{aligned}$$

$$Var\_Msgs \triangleq \langle Var\_M\_read,\ Var\_M\_lock,\ Var\_M\_unlock\rangle$$

@type: $(a \to b) \Rightarrow Set(b)$;
$$Range(F) \triangleq \{F[x] : x \in \text{DOMAIN } F\}$$

@type: $Set(\$\, rid)$;
$$\begin{aligned} RIDs &\triangleq \text{UNION } Range(Shards) \\ TIDs &\triangleq \text{DOMAIN } Txns \end{aligned}$$

$$KeyLookup \triangleq [r \in RIDs \mapsto \text{CHOOSE } k \in \text{DOMAIN } Shards : r \in Shards[k]]$$

$$\begin{aligned} Init \triangleq\ & \\ & \land Replicas = [r \in RIDs \mapsto \\ & \quad [locked \mapsto \text{FALSE},\ version \mapsto \text{“Init”},\ logged \mapsto \text{“NULL”}]] \\ & \land Coordinator\_state = [t \in TIDs \mapsto \text{“Start”}] \\ & \land Coordinator\_txn\_state = [t \in TIDs \mapsto SetAsFun(\{\})] \\ & \land M\_read = \{\} \land M\_read\_resp = \{\} \\ & \land M\_lock = \{\} \land M\_lock\_resp = \{\} \\ & \land M\_unlock = \{\} \land M\_unlock\_resp = \{\} \\ & \land Linearisability\_rt = [t \in TIDs \mapsto \{\}] \end{aligned}$$

$$RelevantReplicas(t) \triangleq \text{UNION } \{Shards[k] : k \in Txns[t]\}$$

$CoordinatorStart(t) \triangleq$
$\land\ Coordinator\_state[t] = \text{“Start”}$
$\land\ M\_read' = M\_read \cup \{[src \mapsto t,\ key \mapsto k] : k \in Txns[t]\}$
$\land$ UNCHANGED $\langle M\_read\_resp,\ Var\_M\_lock,\ Var\_M\_unlock\rangle$
$\land\ Coordinator\_state' = [Coordinator\_state$ EXCEPT $![t] = \text{“Read”}]$
$\land$ UNCHANGED $\langle Coordinator\_txn\_state,\ Replicas\rangle$
$\land\ Linearisability\_rt' = [Linearisability\_rt$ EXCEPT $![t] =$
$\quad \{t1 \in TIDs : Coordinator\_state[t1] = \text{“Commit”}\}]$

$ReplicaRead(r) \triangleq$
$\land\ Replicas[r].locked =$ FALSE
$\land \exists\, m \in M\_read :$
$\quad \land \neg\exists\, m1 \in M\_read\_resp : m1.src = r \land m1.dst = m.src$
$\quad \land\ M\_read\_resp' = M\_read\_resp \cup$
$\quad\quad \{[src \mapsto r,\ dst \mapsto m.src,\ ver \mapsto Replicas[r].version]\}$
$\quad \land$ UNCHANGED $\langle M\_read,\ Var\_M\_lock,\ Var\_M\_unlock\rangle$
$\land$ UNCHANGED $\langle Replicas\rangle$
$\land$ UNCHANGED $\langle Coordinator\_state,\ Coordinator\_txn\_state,\ Linearisability\_rt\rangle$

$CoordinatorRead(t) \triangleq$
$\land\ Coordinator\_state[t] = \text{“Read”}$
$\land \forall\, k \in Txns[t] : \exists\, m \in M\_read\_resp : KeyLookup[m.src] = k$
$\land \exists\, F \in [Txns[t] \to RIDs] :$
$\quad \land \forall\, k \in Txns[t] : \land\ k = KeyLookup[F[k]]$
$\quad\quad\quad\quad\quad\quad\quad \land \exists\, m \in M\_read\_resp : m.dst = t \land m.src = F[k]$
$\quad \land\ Coordinator\_txn\_state' = [Coordinator\_txn\_state$ EXCEPT $![t] = [$
$\quad\quad k \in Txns[t] \mapsto$
$\quad\quad\quad$ (CHOOSE $m \in M\_read\_resp : m.dst = t \land m.src = F[k]).ver$
$\quad\quad]]$
$\land\ Coordinator\_state' = [Coordinator\_state$ EXCEPT $![t] = \text{“Lock”}]$
$\land$ UNCHANGED $\langle Replicas,\ Var\_Msgs,\ Linearisability\_rt\rangle$

$CoordinatorLock(t) \triangleq$
$\land\ Coordinator\_state[t] = \text{“Lock”}$
$\land\ M\_lock' = M\_lock \cup$
$\quad \{[tid \mapsto t,\ txn \mapsto Txns[t],\ state \mapsto Coordinator\_txn\_state[t]]\}$
$\land$ UNCHANGED $\langle M\_lock\_resp,\ Var\_M\_read,\ Var\_M\_unlock\rangle$
$\land\ Coordinator\_state' = [Coordinator\_state$ EXCEPT $![t] = \text{“Decide”}]$
$\land$ UNCHANGED $\langle Coordinator\_txn\_state,\ Replicas,\ Linearisability\_rt\rangle$

$ReplicaLock(r) \triangleq$
$\land \exists\, m \in M\_lock :$
$\quad \land\ KeyLookup[r] \in m.txn$
$\quad \land \neg\exists\, m1 \in M\_lock\_resp : m1.src = r \land m1.dst = m.tid$
$\quad \land$ IF $(\neg Replicas[r].locked) \land Replicas[r].version = m.state[KeyLookup[r]]$
$\quad\quad$ THEN

$\land\ Replicas' = [Replicas \text{ EXCEPT } ![r] = [$
$locked \mapsto \text{TRUE},\ version \mapsto Replicas[r].version,\ logged \mapsto m.tid]]$
$\land\ M\_lock\_resp' = M\_lock\_resp \cup$
$\{[src \mapsto r,\ dst \mapsto m.tid,\ locked \mapsto \text{TRUE}]\}$
ELSE
$\land\ M\_lock\_resp' = M\_lock\_resp \cup$
$\{[src \mapsto r,\ dst \mapsto m.tid,\ locked \mapsto \text{FALSE}]\}$
$\land\ \text{UNCHANGED}\ Replicas$
$\land\ \text{UNCHANGED}\ \langle M\_lock,\ Var\_M\_read,\ Var\_M\_unlock\rangle$
$\land\ \text{UNCHANGED}\ \langle Coordinator\_state,\ Coordinator\_txn\_state,\ Linearisability\_rt\rangle$

$CoordinatorCommit(t) \triangleq$
$\land\ Coordinator\_state[t] =$ "Decide"
$\land\ \forall\, k \in Txns[t] : \forall\, r \in Shards[k] : \exists\, m \in M\_lock\_resp :$
$\land\ m.src = r \land m.dst = t$
$\land\ m.locked$
$\land\ Coordinator\_state' = [Coordinator\_state \text{ EXCEPT } ![t] =$ "Commit"$]$
$\land\ M\_unlock' = M\_unlock \cup \{[src \mapsto t,\ apply \mapsto \text{TRUE}]\}$
$\land\ \text{UNCHANGED}\ \langle Var\_M\_read,\ Var\_M\_lock,\ M\_unlock\_resp\rangle$
$\land\ \text{UNCHANGED}\ \langle Replicas,\ Coordinator\_txn\_state,\ Linearisability\_rt\rangle$

$CoordinatorStartAbort(t) \triangleq$
$\land\ Coordinator\_state[t] =$ "Decide"
$\land\ \exists\, k \in Txns[t] : \exists\, r \in Shards[k] : \exists\, m \in M\_lock\_resp :$
$\land\ m.src = r \land m.dst = t$
$\land\ \neg m.locked$
$\land\ Coordinator\_state' = [Coordinator\_state \text{ EXCEPT } ![t] =$ "TryAbort"$]$
$\land\ M\_unlock' = M\_unlock \cup \{[src \mapsto t,\ apply \mapsto \text{FALSE}]\}$
$\land\ \text{UNCHANGED}\ \langle Var\_M\_read,\ Var\_M\_lock,\ M\_unlock\_resp\rangle$
$\land\ \text{UNCHANGED}\ \langle Replicas,\ Coordinator\_txn\_state,\ Linearisability\_rt\rangle$

$ReplicaUnlock(r) \triangleq$
$\exists\, m \in M\_unlock :$
$\land\ Replicas[r].locked$
$\land\ m.src = Replicas[r].logged$
$\land\ \text{IF}\ m.apply$
$\text{THEN}\ Replicas' = [Replicas \text{ EXCEPT } ![r] = [$
$locked \mapsto \text{FALSE},\ version \mapsto Replicas[r].logged,\ logged \mapsto$ "NULL"$]]$
$\text{ELSE}\ Replicas' = [Replicas \text{ EXCEPT } ![r] = [$
$locked \mapsto \text{FALSE},\ version \mapsto Replicas[r].version,\ logged \mapsto$ "NULL"$]]$
$\land\ M\_unlock\_resp' = M\_unlock\_resp \cup \{[src \mapsto r,\ tid \mapsto Replicas[r].logged]\}$
$\land\ \text{UNCHANGED}\ \langle Var\_M\_read,\ Var\_M\_lock,\ M\_unlock\rangle$
$\land\ \text{UNCHANGED}\ \langle Coordinator\_state,\ Coordinator\_txn\_state,\ Linearisability\_rt\rangle$

$CoordinatorAbort(t) \triangleq$
$\land\ Coordinator\_state[t] =$ "TryAbort"

$\land \exists\, k \in \mathit{Txns}[t] : \forall\, r \in \mathit{Shards}[k] :$
$\quad \lor \exists\, m \in \mathit{M\_unlock\_resp} : m.\mathit{src} = r \land m.\mathit{tid} = t$
$\quad \lor \exists\, m \in \mathit{M\_lock\_resp} : m.\mathit{src} = r \land m.\mathit{dst} = t \land \neg m.\mathit{locked}$
$\land\ \mathit{Coordinator\_state}' = [\mathit{Coordinator\_state}\ \text{EXCEPT}\ ![t] = \text{“Abort”}]$
$\land\ \text{UNCHANGED}\ \langle \mathit{Replicas},\ \mathit{Var\_Msgs},\ \mathit{Coordinator\_txn\_state},\ \mathit{Linearisability\_rt} \rangle$

$\mathit{Next} \triangleq$
$\lor \exists\, r \in \mathit{RIDs} : \lor \mathit{ReplicaRead}(r)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{ReplicaLock}(r)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{ReplicaUnlock}(r)$
$\lor \exists\, t \in \mathit{TIDs} : \lor \mathit{CoordinatorStart}(t)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{CoordinatorRead}(t)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{CoordinatorLock}(t)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{CoordinatorCommit}(t)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{CoordinatorStartAbort}(t)$
$\qquad\qquad\qquad\qquad\ \lor \mathit{CoordinatorAbort}(t)$

$\mathit{Spec} \triangleq \mathit{Init} \land \Box[\mathit{Next}]_{\mathit{Vars}}$

$\mathit{Linearisability}(C) \triangleq$
$\lor\ \mathit{Cardinality}(C) < 2$
$\lor \exists\, R \in \text{SUBSET}\ (C \times C) :$
Irreflexive
$\land \forall\, t1 \in C : \langle t1,\ t1 \rangle \notin R$
Transitive
$\land \forall\, t1,\ t2,\ t3 \in C : (\langle t1,\ t2 \rangle \in R \land \langle t2,\ t3 \rangle \in R) \Rightarrow \langle t1,\ t3 \rangle \in R$
Above 2 ensure there are no cycles
R respects observed order
$\land \forall\, t1,\ t2 \in C :$
$\quad (\exists\, k \in \mathit{Txns}[t2] : \mathit{Coordinator\_txn\_state}[t2][k] = t1) \Rightarrow \langle t1,\ t2 \rangle \in R$
If two transactions interfere, there is an order
$\land \forall\, t1,\ t2 \in C :$
$\quad (t1 \neq t2 \land \mathit{Txns}[t1] \cap \mathit{Txns}[t2] \neq \{\}) \Rightarrow \langle t1,\ t2 \rangle \in R \lor \langle t2,\ t1 \rangle \in R$
Strict serialisability / *Linearisability* check
$\land \forall\, t1,\ t2 \in C :$
$\quad (t1 \in \mathit{Linearisability\_rt}[t2]) \Rightarrow \langle t1,\ t2 \rangle \in R$

$\mathit{CommittedTIDs} \triangleq \{t \in \mathit{TIDs} : \mathit{Coordinator\_state}[t] = \text{“Commit”}\}$
$\mathit{AbortedTIDs} \triangleq \{t \in \mathit{TIDs} : \mathit{Coordinator\_state}[t] = \text{“Abort”}\}$

$\mathit{Safety\_non\_recovery} \triangleq \mathit{Linearisability}(\mathit{CommittedTIDs})$