MODULE *AWSet*

EXTENDS *Naturals*, *Sequences*, *SEC*

CONSTANTS

Data the set of data

VARIABLES

aSet, *aSet*[*r*]: set of active *Instance*(*s*) maintained by $r \in$ *Replica*

tSet, *tSet*[*r*]: set of tombstone *Instance*(*s*) maintained by $r \in$ *Replica*

seq, *seq*[*r*]: local sequence number at replica $r \in$ *Replica*

incoming, *incoming*[*r*]: incoming messages at replica $r \in$ *Replica*

msg,

messageSet

$vars \triangleq \langle aSet, tSet, seq, incoming, msg, messageSet, SECvars \rangle$

$Instance \triangleq [d : Data, r : Replica, k : Nat]$

$Msg \triangleq [r : Replica, seq : Nat, update : \text{SUBSET } Update, A : \text{SUBSET } Instance, T : \text{SUBSET } Instance]$

$Network \triangleq \text{INSTANCE } Network$

$TypeOK \triangleq$

$\land \quad aSet \in [Replica \to \text{SUBSET } Instance]$

$\land \quad tSet \in [Replica \to \text{SUBSET } Instance]$

$\land \quad seq \in [Replica \to Nat]$

$Init \triangleq$

$\land\, Network\,!\,NInit$

$\land\, SECInit$

$\land\, seq = [r \in Replica \mapsto 0]$

$\land\, aSet = [r \in Replica \mapsto \{\}]$

$\land\, tSet = [r \in Replica \mapsto \{\}]$

$Add(d, r) \triangleq$

$\land\, seq' = [seq \text{ EXCEPT } ![r] = @ + 1]$

$\land\, SECUpdate(r, seq[r])$

$\land\, aSet' = [aSet \text{ EXCEPT } ![r] = @ \cup \{[d \mapsto d, r \mapsto r, k \mapsto seq'[r]]\}]$

$\land \text{ UNCHANGED } \langle tSet, incoming, msg, messageSet \rangle$

$Remove(d, r) \triangleq$

$\land\, seq' = [seq \text{ EXCEPT } ![r] = @ + 1]$

$\land\, SECUpdate(r, seq[r])$

$\land \text{ LET } D \triangleq \{ins \in aSet[r] : ins.d = d\}$

IN $\land\, aSet' = [aSet \text{ EXCEPT } ![r] = @ \setminus D]$

$\land\, tSet' = [tSet \text{ EXCEPT } ![r] = @ \cup D]$

$\land \text{ UNCHANGED } \langle incoming, msg, messageSet \rangle$

$$
\begin{array}{l}
\textit{Broadcast}(s,\, m) \triangleq \\
\quad [r \in \textit{Replica} \mapsto \text{IF } s = r \text{ THEN } \textit{incoming}[s] \\
\qquad\qquad\qquad\qquad\quad \text{ELSE } \textit{incoming}[r] \circ \langle m \rangle] \\
\\
\textit{Send}(r) \triangleq \\
\quad \land \textit{Network}!\textit{NBroadcast}(r,\, [r \mapsto r,\, \textit{seq} \mapsto \textit{seq}[r],\, \textit{update} \mapsto \textit{StateUpdate}(r),\, A \mapsto \textit{aSet}[r],\, T \mapsto \textit{tSet}[r]]) \\
\quad \land \textit{SECSend}(r) \\
\quad \land \text{UNCHANGED } \langle \textit{aSet},\, \textit{tSet},\, \textit{seq} \rangle \\
\\
\textit{Deliver}(r) \triangleq \\
\quad \land \ \textit{Network}!\textit{NDeliver}(r) \\
\quad \land \ \textit{SECDeliver}(r,\, \textit{msg}'[r]) \\
\quad \land \ \textit{seq}' = [\textit{seq} \text{ EXCEPT } ![r] = @ + 1] \\
\quad \land \ \textit{tSet}' = [\textit{tSet} \text{ EXCEPT } ![r] = @ \cup \textit{msg}'[r].T] \\
\quad \land \ \textit{aSet}' = [\textit{aSet} \text{ EXCEPT } ![r] = (@ \cup \textit{msg}'[r].A) \setminus \textit{tSet}'[r]] \\
\quad \land \ \text{UNCHANGED } \langle\rangle
\end{array}
$$

$$
\begin{array}{l}
\textit{Next} \triangleq \\
\quad \lor \exists\, r \in \textit{Replica} : \exists\, a \in \textit{Data} : \\
\qquad \textit{Add}(a,\, r) \lor \textit{Remove}(a,\, r) \\
\quad \lor \exists\, r \in \textit{Replica} : \\
\qquad \textit{Send}(r) \lor \textit{Deliver}(r) \\
\\
\textit{Spec} \triangleq \textit{Init} \land \Box[\textit{Next}]_{\textit{vars}}
\end{array}
$$

$$
\textit{Read}(r) \triangleq \{\textit{ins}.d : \textit{ins} \in \textit{aSet}[r]\}
$$

QC: Quiescent Consistency

$$
\begin{array}{l}
\textit{Quiescence} \triangleq \\
\quad \forall\, r \in \textit{Replica} : \textit{incoming}[r] = \langle\rangle \\
\\
\textit{Convergence} \triangleq \\
\quad \forall\, r,\, s \in \textit{Replica} : \textit{Read}(r) = \textit{Read}(s) \\
\\
\textit{QC} \triangleq \textit{Quiescence} \Rightarrow \textit{Convergence}
\end{array}
$$

SEC: Strong Eventual Consistency

$$
\begin{array}{l}
\textit{SEC} \triangleq \forall\, r1,\, r2 \in \textit{Replica} : \\
\qquad \textit{Sameupdate}(r1,\, r2) \Rightarrow \textit{Read}(r1) = \textit{Read}(r2)
\end{array}
$$

* Modification History
* Last modified *Tue* May 14 11:24:42 *CST* 2019 by *zfwang*
* Last modified Sun *Apr* 28 15:09:12 *CST* 2019 by *jywellin*
* Created Sun *Apr* 28 14:02:54 *CST* 2019 by *jywellin*