MODULE *AJupiter*

Specification of the *Jupiter* protocol presented by *Attiya* et al.

EXTENDS *JupiterInterface*, *OT*, *BufferStateSpace*

VARIABLES

cbuf, *cbuf*[*c*]: buffer for locally generated operations at client $c \in Client$

crec, *crec*[*c*]: number of remote operations received by client $c \in Client$ since the last time a local operation was generated

sbuf, *sbuf*[*c*]: buffer for transformed remote operations *w.r.t* client $c \in Client$

srec *srec*[*c*]: number of locally generated operations by client $c \in Client$ since the last time a remote operation was transformed at the *Server*

$vars \triangleq \langle intVars, cbuf, crec, sbuf, srec \rangle$

$AJMsg \triangleq$
- $[c : Client, ack : Nat, op : Op \cup \{Nop\}] \cup$ messages sent to the *Server* from client $c \in Client$
- $[ack : Nat, op : Op \cup \{Nop\}]$ messages broadcast to Clients from the *Server*

$TypeOK \triangleq$
- $\land$ $TypeOKInt$
- $\land$ $cbuf \in [Client \to Seq(Op \cup \{Nop\})]$
- $\land$ $crec \in [Client \to Nat]$
- $\land$ $sbuf \in [Client \to Seq(Op \cup \{Nop\})]$
- $\land$ $srec \in [Client \to Nat]$

$Init \triangleq$
- $\land$ $InitInt$
- $\land$ $cbuf = [c \in Client \mapsto \langle\rangle]$
- $\land$ $crec = [c \in Client \mapsto 0]$
- $\land$ $sbuf = [c \in Client \mapsto \langle\rangle]$
- $\land$ $srec = [c \in Client \mapsto 0]$

$ClientPerform(c, m) \triangleq$
- LET $xform \triangleq xFormShift(OT, m.op, cbuf[c], m.ack)$ [*xop*, *xops*]
- IN $\land$ $cbuf' = [cbuf$ EXCEPT $![c] = xform.xops]$
  - $\land$ $crec' = [crec$ EXCEPT $![c] = @ + 1]$
  - $\land$ $SetNewAop(c, xform.xop)$

$ServerPerform(m) \triangleq$
- LET $c \triangleq m.c$
  - $xform \triangleq xFormShift(OT, m.op, sbuf[c], m.ack)$ [*xop*, *xops*]
  - $xop \triangleq xform.xop$
- IN $\land$ $srec' = [cl \in Client \mapsto$ IF $cl = c$ THEN $srec[cl] + 1$ ELSE $0]$
  - $\land$ $sbuf' = [cl \in Client \mapsto$ IF $cl = c$ THEN $xform.xops$ ELSE $Append(sbuf[cl], xop)]$

$\land\ SetNewAop(Server,\ xop)$
$\land\ Comm!SSend(c,\ [cl \in Client \mapsto [ack \mapsto srec[cl],\ op \mapsto xop]])$

$DoOp(c,\ op) \triangleq$
$\land\ SetNewAop(c,\ op)$
$\land\ cbuf' = [cbuf \text{ EXCEPT } ![c] = Append(@,\ op)]$
$\land\ crec' = [crec \text{ EXCEPT } ![c] = 0]$
$\land\ Comm!CSend([c \mapsto c,\ ack \mapsto crec[c],\ op \mapsto op])$

$Do(c) \triangleq$
$\land\ DoInt(DoOp,\ c)$
$\land$ UNCHANGED $\langle sbuf,\ srec\rangle$

$Rev(c) \triangleq$
$\land\ RevInt(ClientPerform,\ c)$
$\land$ UNCHANGED $\langle sbuf,\ srec\rangle$

$SRev \triangleq$
$\land\ SRevInt(ServerPerform)$
$\land$ UNCHANGED $\langle cbuf,\ crec\rangle$

$Next \triangleq$
$\lor \exists\, c \in Client : Do(c) \lor Rev(c)$
$\lor\ SRev$

$Fairness \triangleq$
$\mathrm{WF}_{vars}(SRev \lor \exists\, c \in Client : Rev(c))$

$Spec \triangleq Init \land \Box[Next]_{vars}$ $\land\ Fairness$

$QC \triangleq$ Quiescent Consistency
$Comm!EmptyChannel \Rightarrow Cardinality(Range(state)) = 1$

THEOREM $Spec \Rightarrow \Box QC$

---

* Modification History
* Last modified *Thu Jan* 17 10:30:39 *CST* 2019 by *hengxin*
* Created *Satchins*, Jun 23 17:14:18 *CST* 2018 by *hengxin*