─────────────── MODULE *CSComm* ───────────────

EXTENDS *SequenceUtils*

───

CONSTANTS
    Client,   the set of clients
    Server, the (unique) server
    Msg   the set of messages

───

VARIABLES
    cincoming, $cincoming[c]$: incoming *FIFO* channel at client $c \in Client$
    sincoming   incoming *FIFO* channel at the *Server*

───

$TypeOK \triangleq$
    $\wedge \quad cincoming \in [Client \rightarrow Seq(Msg)]$
    $\wedge \quad sincoming \in Seq(Msg)$

$Init \triangleq$
    $\wedge cincoming = [c \in Client \mapsto \langle\rangle]$
    $\wedge sincoming = \langle\rangle$

$EmptyChannel \triangleq Init$

───

$CSend(m) \triangleq$ A client sends a message m to the *Server*.
    $\wedge sincoming' = Append(sincoming, m)$
    $\wedge$ UNCHANGED *cincoming*

$CRev(c) \triangleq$ Client c receives and consumes a message from the *Server*.
    $\wedge cincoming[c] \neq \langle\rangle$
    $\wedge cincoming' = [cincoming$ EXCEPT $![c] = Tail(@)]$
    $\wedge$ UNCHANGED *sincoming*

───

$SRev \triangleq$ The *Server* receives and consumes a message.
    $\wedge sincoming \neq \langle\rangle$
    $\wedge sincoming' = Tail(sincoming)$

$SSend(c, cm) \triangleq$ The *Server* sents a message $cm[cl]$ to client cl (other than c).
    $cincoming' = [cl \in Client \mapsto$ IF $cl = c$ THEN $cincoming[cl]$
                              ELSE $Append(cincoming[cl], cm[cl])]$

$SSendSame(c, m) \triangleq$ The *Server* broadcasts the message m to all clients other than c.
    $SSend(c, [cl \in Client \mapsto m])$

───

* Modification History
* Last modified Sat *Jan* 19 16:28:48 *CST* 2019 by anonymous
* Created Sun *Jun* 24 10:25:34 *CST* 2018 by anonymous

1