―――――――――――― MODULE $xApp$ ――――――――――――

LOCAL INSTANCE $Naturals$

LOCAL INSTANCE $Sequences$

LOCAL INSTANCE $FiniteSets$

LOCAL INSTANCE TLC

───

An empty value
CONSTANT Nil

Node states
CONSTANT $Stopped$, $Started$

A set of $E2T$ node identifiers
CONSTANT $E2TNodes$

ASSUME $\land IsFiniteSet(E2TNodes)$
$\land \forall n \in E2TNodes : n \in$ STRING

A mapping of node states
VARIABLE $nodes$

Connections to $E2T$ nodes
VARIABLE $conns$

$vars \triangleq \langle nodes, conns \rangle$

LOCAL $E2T \triangleq$ INSTANCE $E2TService$

───

$StartNode(n) \triangleq$
$\quad \land nodes[n] = Stopped$
$\quad \land nodes' = [nodes$ EXCEPT $![n] = Started]$
$\quad \land$ UNCHANGED $\langle conns \rangle$

$StopNode(n) \triangleq$
$\quad \land nodes[n] = Started$
$\quad \land nodes' = [nodes$ EXCEPT $![n] = Stopped]$
$\quad \land$ UNCHANGED $\langle conns \rangle$

───

$SendControlRequest(n, c) \triangleq$
$\quad \land$ UNCHANGED $\langle nodes \rangle$

1

$HandleControlResponse(n, c, r) \triangleq$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$SendSubscribeRequest(n, c) \triangleq$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$HandleSubscribeResponse(n, c, r) \triangleq$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$SendUnsubscribeRequest(n, c) \triangleq$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$HandleUnsubscribeResponse(n, c, r) \triangleq$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$HandleE2TResponse(n, c) \triangleq$
 $\wedge \vee E2T!Client!Handle!SubscribeResponse(c,$ LAMBDA $m : HandleSubscribeResponse(n, c, m))$
  $\vee E2T!Client!Handle!UnsubscribeResponse(c,$ LAMBDA $m : HandleUnsubscribeResponse(n, c, m))$
  $\vee E2T!Client!Handle!ControlResponse(c,$ LAMBDA $m : HandleControlResponse(n, c, m))$
 $\wedge$ UNCHANGED $\langle nodes \rangle$

$Init \triangleq$
 $\wedge E2T!Init$
 $\wedge nodes = [n \in E2TNodes \mapsto Stopped]$

$Next \triangleq$
 $\vee \exists n \in E2TNodes : StartNode(n)$
 $\vee \exists n \in E2TNodes : StopNode(n)$
 $\vee \exists n \in E2TNodes, c \in E2T!Connections : SendSubscribeRequest(n, c)$
 $\vee \exists n \in E2TNodes, c \in E2T!Connections : SendUnsubscribeRequest(n, c)$
 $\vee \exists n \in E2TNodes, c \in E2T!Connections : SendControlRequest(n, c)$
 $\vee \exists n \in E2TNodes, c \in E2T!Connections : HandleE2TResponse(n, c)$

\ * Modification History
\ * Last modified *Mon Sep* 13 19:33:23 *PDT* 2021 by *jordanhalterman*
\ * *Created Mon Sep* 13 15:19:07 *PDT* 2021 by *jordanhalterman*