

---

MODULE *RIC*

---

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 *RIC* node identifiers  
 CONSTANT *RICNode*

ASSUME  $\wedge IsFiniteSet(RICNode)$   
 $\wedge \forall n \in RICNode : n \in \text{STRING}$

A mapping of node states  
 VARIABLE *state*

Network state  
 VARIABLE *network*

A global store of *E2* node states  
 VARIABLE *nodes*

A set of *E2* node management connections  
 VARIABLE *mgmtConn*

A transaction *ID* counter  
 VARIABLE *txId*

A set of *E2* node data connections  
 VARIABLE *dataConn*

A request *ID* counter  
 VARIABLE *reqId*

$vars \triangleq \langle state, network, nodes, mgmtConn, txId, dataConn, reqId \rangle$

LOCAL *E2AP*  $\triangleq$  INSTANCE *E2AP* WITH *conns*  $\leftarrow network$

---

$$\begin{aligned}
\text{StartNode}(\text{node}) &\triangleq \\
&\wedge \text{state}[\text{node}] = \text{Stopped} \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Start} \\
&\wedge \text{state}' = [\text{nodes} \text{ EXCEPT } ![\text{node}] = \text{Started}] \\
&\wedge \text{UNCHANGED } \langle \text{nodes} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{StopNode}(\text{node}) &\triangleq \\
&\wedge \text{state}[\text{node}] = \text{Started} \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Stop} \\
&\wedge \text{state}' = [\text{nodes} \text{ EXCEPT } ![\text{node}] = \text{Stopped}] \\
&\wedge \text{UNCHANGED } \langle \text{network}, \text{nodes} \rangle
\end{aligned}$$


---


$$\begin{aligned}
\text{HandleE2SetupRequest}(\text{node}, \text{conn}, \text{req}) &\triangleq \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Receive!E2SetupRequest}(\text{conn}, \text{req}) \\
&\wedge \vee \wedge \text{conn.id} \notin \text{DOMAIN } \text{mgmtConn} \\
&\quad \wedge \text{mgmtConn}' = \text{mgmtConn} @ @ (\text{conn.id} :> \text{conn}) \\
&\quad \vee \wedge \text{conn.id} \in \text{DOMAIN } \text{mgmtConn} \\
&\quad \wedge \text{UNCHANGED } \langle \text{mgmtConn} \rangle \\
&\wedge \vee \wedge \text{req.e2NodeId} \notin \text{DOMAIN } \text{nodes} \\
&\quad \wedge \text{nodes}' = \text{nodes} @ @ (\text{req.e2NodeId} :> [\text{e2NodeId} \mapsto \text{req.e2NodeId}, \\
&\quad \quad \quad \text{sms} \quad \quad \mapsto \text{req.sms}]) \\
&\quad \vee \wedge \text{req.e2NodeId} \in \text{DOMAIN } \text{nodes} \\
&\quad \wedge \text{nodes}' = [\text{nodes} \text{ EXCEPT } ![\text{req.e2NodeId}] = [ \\
&\quad \quad \quad \text{nodes}[\text{req.e2NodeId}] \text{ EXCEPT } !.\text{sms} = \text{req.sms}]] \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Reply!E2SetupResponse}(\text{conn}, [\text{txId} \mapsto \text{req.txId}]) \\
&\wedge \text{UNCHANGED } \langle \text{state}, \text{txId}, \text{reqId} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{HandleRICControlResponse}(\text{node}, \text{conn}, \text{res}) &\triangleq \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Receive!RICControlResponse}(\text{conn}, \text{res}) \\
&\wedge \text{UNCHANGED } \langle \text{state} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{HandleRICSubscriptionResponse}(\text{node}, \text{conn}, \text{res}) &\triangleq \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Receive!RICSubscriptionResponse}(\text{conn}, \text{res}) \\
&\wedge \text{UNCHANGED } \langle \text{state} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{HandleRICSubscriptionDeleteResponse}(\text{node}, \text{conn}, \text{res}) &\triangleq \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Receive!RICSubscriptionDeleteResponse}(\text{conn}, \text{res}) \\
&\wedge \text{UNCHANGED } \langle \text{state} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{HandleRICIndication}(\text{node}, \text{conn}, \text{res}) &\triangleq \\
&\wedge \text{E2AP!Server}(\text{node})!\text{Receive!RICIndication}(\text{conn}, \text{res}) \\
&\wedge \text{UNCHANGED } \langle \text{state} \rangle
\end{aligned}$$

$$\begin{aligned}
\text{E2ConnectionUpdate}(\text{node}) &\triangleq \\
&\wedge \text{node} \in \text{DOMAIN } \text{mgmtConn}
\end{aligned}$$

$$\begin{aligned}
& \wedge E2AP!Server(node)!Send!E2ConnectionUpdate(mgmtConn[node], [txId \mapsto txId[node] + 1]) \\
& \wedge txId' = [txId \text{ EXCEPT } ![node] = txId[node] + 1] \\
& \wedge \text{UNCHANGED } \langle state \rangle
\end{aligned}$$

$$\begin{aligned}
HandleE2NodeConfigurationUpdate(node, conn, req) & \triangleq \\
& \wedge E2AP!Server(node)!Receive!E2NodeConfigurationUpdate(conn, req) \\
& \wedge E2AP!Server(node)!Reply!E2NodeConfigurationUpdateAcknowledge(conn, [txId \mapsto req.txId]) \\
& \wedge \text{UNCHANGED } \langle state \rangle
\end{aligned}$$


---


$$Init \triangleq$$

$$\begin{aligned}
& \wedge E2AP!Init \\
& \wedge state = [n \in RICNode \mapsto Stopped] \\
& \wedge nodes = [n \in \{\} \mapsto [id \mapsto Nil]] \\
& \wedge mgmtConn = [n \in \{\} \mapsto Nil] \\
& \wedge txId = [n \in RICNode \mapsto 0] \\
& \wedge dataConn = [n \in \{\} \mapsto Nil] \\
& \wedge reqId = [n \in RICNode \mapsto 0]
\end{aligned}$$

$$Next \triangleq$$

$$\begin{aligned}
& \vee \exists node \in RICNode : StartNode(node) \\
& \vee \exists node \in RICNode : StopNode(node) \\
& \vee \exists node \in RICNode : E2ConnectionUpdate(node) \\
& \vee \exists node \in RICNode : \\
& \quad \exists conn \in E2AP!Server(node)!Connections : \\
& \quad \vee E2AP!Server(node)!Handle!E2SetupRequest(conn, LAMBDA c, m : HandleE2SetupRequest(node, \\
& \quad \vee E2AP!Server(node)!Handle!RICControlResponse(conn, LAMBDA c, m : HandleRICControlRespon \\
& \quad \vee E2AP!Server(node)!Handle!RICSubscriptionResponse(conn, LAMBDA c, m : HandleRICSubscrip \\
& \quad \vee E2AP!Server(node)!Handle!RICSubscriptionDeleteResponse(conn, LAMBDA c, m : HandleRICSu \\
& \quad \vee E2AP!Server(node)!Handle!RICIndication(conn, LAMBDA c, m : HandleRICIndication(node, con \\
& \quad \vee E2AP!Server(node)!Handle!E2NodeConfigurationUpdate(conn, LAMBDA c, m : HandleE2NodeC
\end{aligned}$$


---

\\* Modification History  
\\* Last modified Tue Sep 21 10:06:48 PDT 2021 by jordanhalterman  
\\* Created Tue Sep 21 02:14:49 PDT 2021 by jordanhalterman