

```

|----- MODULE Protocols -----|

EXTENDS TLC

|----- MODULE E2AP -----|

  Message type constants
CONSTANTS
  E2Setup,
  E2SetupResponse,
  E2SetupFailure
CONSTANTS
  ResetRequest,
  ResetResponse
CONSTANTS
  RICSubscriptionRequest,
  RICSubscriptionResponse,
  RICSubscriptionFailure
CONSTANTS
  RICSubscriptionDeleteRequest,
  RICSubscriptionDeleteResponse,
  RICSubscriptionDeleteFailure
CONSTANTS
  RICControlRequest,
  RICControlResponse,
  RICControlFailure,
  RICServiceUpdate
CONSTANTS
  E2ConnectionUpdate,
  E2ConnectionUpdateAcknowledge,
  E2ConnectionUpdateFailure
CONSTANTS
  E2NodeConfigurationUpdate,
  E2NodeConfigurationUpdateAcknowledge,
  E2NodeConfigurationUpdateFailure

LOCAL messageTypes  $\triangleq$ 
  {E2Setup,
   E2SetupResponse,
   E2SetupFailure,
   ResetRequest,
   ResetResponse,
   RICSubscriptionRequest,
   RICSubscriptionResponse,
   RICSubscriptionFailure,
   RICSubscriptionDeleteRequest,
```

RICSubscriptionDeleteResponse,
RICSubscriptionDeleteFailure,
RICControlRequest,
RICControlResponse,
RICControlFailure,
RICServiceUpdate,
E2ConnectionUpdate,
E2ConnectionUpdateAcknowledge,
E2ConnectionUpdateFailure,
E2NodeConfigurationUpdate,
E2NodeConfigurationUpdateAcknowledge,
E2NodeConfigurationUpdateFailure}

ASSUME $\forall m \in \text{messageTypes} : m \in \text{STRING}$

Failure cause constants

CONSTANTS

MiscFailureUnspecified,
MiscFailureControlProcessingOverload,
MiscFailureHardwareFailure,
MiscFailureOMIntervention

CONSTANTS

ProtocolFailureUnspecified,
ProtocolFailureTransferSyntaxError,
ProtocolFailureAbstractSyntaxErrorReject,
ProtocolFailureAbstractSyntaxErrorIgnoreAndNotify,
ProtocolFailureMessageNotCompatibleWithReceiverState,
ProtocolFailureSemanticError,
ProtocolFailureAbstractSyntaxErrorFalselyConstructedMessage

CONSTANTS

RICFailureUnspecified,
RICFailureRANFunctionIDInvalid,
RICFailureActionNotSupported,
RICFailureExcessiveActions,
RICFailureDuplicateAction,
RICFailureDuplicateEvent,
RICFailureFunctionResourceLimit,
RICFailureRequestIDUnknown,
RICFailureInconsistentActionSubsequentActionSequence,
RICFailureControlMessageInvalid,
RICFailureCallProcessIDInvalid

CONSTANTS

RICServiceFailureUnspecified,
RICServiceFailureFunctionNotRequired,
RICServiceFailureExcessiveFunctions,

$RICServiceFailureRICResourceLimit$
 CONSTANTS
 $TransportFailureUnspecified,$
 $TransportFailureTransportResourceUnavailable$
 LOCAL $failureCauses \triangleq$
 $\{$ $MiscFailureUnspecified,$
 $MiscFailureControlProcessingOverload,$
 $MiscFailureHardwareFailure,$
 $MiscFailureOMIntervention,$
 $ProtocolFailureUnspecified,$
 $ProtocolFailureTransferSyntaxError,$
 $ProtocolFailureAbstractSyntaxErrorReject,$
 $ProtocolFailureAbstractSyntaxErrorIgnoreAndNotify,$
 $ProtocolFailureMessageNotCompatibleWithReceiverState,$
 $ProtocolFailureSemanticError,$
 $ProtocolFailureAbstractSyntaxErrorFalselyConstructedMessage,$
 $RICFailureUnspecified,$
 $RICFailureRANFunctionIDInvalid,$
 $RICFailureActionNotSupported,$
 $RICFailureExcessiveActions,$
 $RICFailureDuplicateAction,$
 $RICFailureDuplicateEvent,$
 $RICFailureFunctionResourceLimit,$
 $RICFailureRequestIDUnknown,$
 $RICFailureInconsistentActionSubsequentActionSequence,$
 $RICFailureControlMessageInvalid,$
 $RICFailureCallProcessIDInvalid,$
 $RICServiceFailureUnspecified,$
 $RICServiceFailureFunctionNotRequired,$
 $RICServiceFailureExcessiveFunctions,$
 $RICServiceFailureRICResourceLimit,$
 $TransportFailureUnspecified,$
 $TransportFailureTransportResourceUnavailable\}$
 ASSUME $\forall c \in failureCauses : c \in \text{STRING}$
 $Protocol \triangleq [t \in messageTypes \mapsto t]$
 $Cause \triangleq [c \in failureCauses \mapsto c]$

 LOCAL $IsValidE2Setup(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2SetupResponse(m) \triangleq \text{TRUE}$

LOCAL $IsValidE2SetupFailure(m) \triangleq \text{TRUE}$
 LOCAL $IsValidResetRequest(m) \triangleq \text{TRUE}$
 LOCAL $IsValidResetResponse(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionRequest(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionResponse(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionFailure(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionDeleteRequest(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionDeleteResponse(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICSubscriptionDeleteFailure(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICControlRequest(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICControlResponse(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICControlFailure(m) \triangleq \text{TRUE}$
 LOCAL $IsValidRICServiceUpdate(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2ConnectionUpdate(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2ConnectionUpdateAcknowledge(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2ConnectionUpdateFailure(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2NodeConfigurationUpdate(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2NodeConfigurationUpdateAcknowledge(m) \triangleq \text{TRUE}$
 LOCAL $IsValidE2NodeConfigurationUpdateFailure(m) \triangleq \text{TRUE}$

LOCAL $IsValidMessage(m) \triangleq$
 $\vee \wedge m.type = E2Setup$
 $\wedge IsValidE2Setup(m)$
 $\vee \wedge m.type = E2SetupResponse$
 $\wedge IsValidE2SetupResponse(m)$
 $\vee \wedge m.type = E2SetupFailure$
 $\wedge IsValidE2SetupFailure(m)$
 $\vee \wedge m.type = ResetRequest$
 $\wedge IsValidResetRequest(m)$
 $\vee \wedge m.type = ResetResponse$
 $\wedge IsValidResetResponse(m)$

$\vee \wedge m.type = RICSubscriptionRequest$
 $\wedge IsValidRICSubscriptionRequest(m)$
 $\vee \wedge m.type = RICSubscriptionResponse$
 $\wedge IsValidRICSubscriptionResponse(m)$
 $\vee \wedge m.type = RICSubscriptionFailure$
 $\wedge IsValidRICSubscriptionFailure(m)$
 $\vee \wedge m.type = RICSubscriptionDeleteRequest$
 $\wedge IsValidRICSubscriptionDeleteRequest(m)$
 $\vee \wedge m.type = RICSubscriptionDeleteResponse$
 $\wedge IsValidRICSubscriptionDeleteResponse(m)$
 $\vee \wedge m.type = RICSubscriptionDeleteFailure$
 $\wedge IsValidRICSubscriptionDeleteFailure(m)$
 $\vee \wedge m.type = RICControlRequest$
 $\wedge IsValidRICControlRequest(m)$
 $\vee \wedge m.type = RICControlResponse$
 $\wedge IsValidRICControlResponse(m)$
 $\vee \wedge m.type = RICControlFailure$
 $\wedge IsValidRICControlFailure(m)$
 $\vee \wedge m.type = RICServiceUpdate$
 $\wedge IsValidRICServiceUpdate(m)$
 $\vee \wedge m.type = E2ConnectionUpdate$
 $\wedge IsValidE2ConnectionUpdate(m)$
 $\vee \wedge m.type = E2ConnectionUpdateAcknowledge$
 $\wedge IsValidE2ConnectionUpdateAcknowledge(m)$
 $\vee \wedge m.type = E2ConnectionUpdateFailure$
 $\wedge IsValidE2ConnectionUpdateFailure(m)$
 $\vee \wedge m.type = E2NodeConfigurationUpdate$
 $\wedge IsValidE2NodeConfigurationUpdate(m)$
 $\vee \wedge m.type = E2NodeConfigurationUpdateAcknowledge$
 $\wedge IsValidE2NodeConfigurationUpdateAcknowledge(m)$
 $\vee \wedge m.type = E2NodeConfigurationUpdateFailure$
 $\wedge IsValidE2NodeConfigurationUpdateFailure(m)$

VARIABLE *servers*, *conns*

INSTANCE *Messaging* WITH *Nil* \leftarrow "<nil>", *IsValid* \leftarrow *IsValidMessage*

VARIABLE *e2apServers*, *e2apConns*

$E2AP \triangleq$ INSTANCE *E2AP* WITH
servers \leftarrow *e2apServers*,
conns \leftarrow *e2apConns*,
E2Setup \leftarrow "E2Setup",
E2SetupResponse \leftarrow "E2SetupResponse",

E2SetupFailure \leftarrow "E2SetupFailure",
ResetRequest \leftarrow "ResetRequest",
ResetResponse \leftarrow "ResetResponse",
RICSubscriptionRequest \leftarrow "RICSubscriptionRequest",
RICSubscriptionResponse \leftarrow "RICSubscriptionResponse",
RICSubscriptionFailure \leftarrow "RICSubscriptionFailure",
RICSubscriptionDeleteRequest \leftarrow "RICSubscriptionDeleteRequest",
RICSubscriptionDeleteResponse \leftarrow "RICSubscriptionDeleteResponse",
RICSubscriptionDeleteFailure \leftarrow "RICSubscriptionDeleteFailure",
RICControlRequest \leftarrow "RICControlRequest",
RICControlResponse \leftarrow "RICControlResponse",
RICControlFailure \leftarrow "RICControlFailure",
RICServiceUpdate \leftarrow "RICServiceUpdate",
E2ConnectionUpdate \leftarrow "E2ConnectionUpdate",
E2ConnectionUpdateAcknowledge \leftarrow "E2ConnectionUpdateAcknowledge",
E2ConnectionUpdateFailure \leftarrow "E2ConnectionUpdateFailure",
E2NodeConfigurationUpdate \leftarrow "E2NodeConfigurationUpdate",
E2NodeConfigurationUpdateAcknowledge \leftarrow "E2NodeConfigurationUpdateAcknowledge",
E2NodeConfigurationUpdateFailure \leftarrow "E2NodeConfigurationUpdateFailure",
MiscFailureUnspecified \leftarrow "MiscFailureUnspecified",
MiscFailureControlProcessingOverload \leftarrow "MiscFailureControlProcessingOverload",
MiscFailureHardwareFailure \leftarrow "MiscFailureHardwareFailure",
MiscFailureOMIntervention \leftarrow "MiscFailureOMIntervention",
ProtocolFailureUnspecified \leftarrow "ProtocolFailureUnspecified",
ProtocolFailureTransferSyntaxError \leftarrow "ProtocolFailureTransferSyntaxError",
ProtocolFailureAbstractSyntaxErrorReject \leftarrow "ProtocolFailureAbstractSyntaxErrorReject",
ProtocolFailureAbstractSyntaxErrorIgnoreAndNotify \leftarrow "ProtocolFailureAbstractSyntaxErrorIgnoreAndNotify",
ProtocolFailureMessageNotCompatibleWithReceiverState \leftarrow "ProtocolFailureMessageNotCompatibleWithReceiverState",
ProtocolFailureSemanticError \leftarrow "ProtocolFailureSemanticError",
ProtocolFailureAbstractSyntaxErrorFalselyConstructedMessage \leftarrow "ProtocolFailureAbstractSyntaxErrorFalselyConstructedMessage",
RICFailureUnspecified \leftarrow "RICFailureUnspecified",
RICFailureRANFunctionIDInvalid \leftarrow "RICFailureRANFunctionIDInvalid",
RICFailureActionNotSupported \leftarrow "RICFailureActionNotSupported",
RICFailureExcessiveActions \leftarrow "RICFailureExcessiveActions",
RICFailureDuplicateAction \leftarrow "RICFailureDuplicateAction",
RICFailureDuplicateEvent \leftarrow "RICFailureDuplicateEvent",
RICFailureFunctionResourceLimit \leftarrow "RICFailureFunctionResourceLimit",
RICFailureRequestIDUnknown \leftarrow "RICFailureRequestIDUnknown",
RICFailureInconsistentActionSubsequentActionSequence \leftarrow "RICFailureInconsistentActionSubsequentActionSequence",
RICFailureControlMessageInvalid \leftarrow "RICFailureControlMessageInvalid",
RICFailureCallProcessIDInvalid \leftarrow "RICFailureCallProcessIDInvalid",
RICServiceFailureUnspecified \leftarrow "RICServiceFailureUnspecified",
RICServiceFailureFunctionNotRequired \leftarrow "RICServiceFailureFunctionNotRequired",
RICServiceFailureExcessiveFunctions \leftarrow "RICServiceFailureExcessiveFunctions",
RICServiceFailureRICResourceLimit \leftarrow "RICServiceFailureRICResourceLimit",

$TransportFailureUnspecified \leftarrow \text{"TransportFailureUnspecified"},$
 $TransportFailureTransportResourceUnavailable \leftarrow \text{"TransportFailureTransportResourceUnavailable"}$

MODULE $E2T$

Message type constants

CONSTANT

$SubscribeRequest,$
 $SubscribeResponse$

CONSTANTS

$UnsubscribeRequest,$
 $UnsubscribeResponse$

CONSTANTS

$ControlRequest,$
 $ControlResponse$

LOCAL $messageTypes \triangleq$
 $\{SubscribeRequest,$
 $SubscribeResponse,$
 $UnsubscribeRequest,$
 $UnsubscribeResponse,$
 $ControlRequest,$
 $ControlResponse\}$

ASSUME $\forall m \in messageTypes : m \in \text{STRING}$

$Protocol \triangleq [t \in messageTypes \mapsto t]$

LOCAL $IsValidSubscribeRequest(m) \triangleq \text{TRUE}$

LOCAL $IsValidSubscribeResponse(m) \triangleq \text{TRUE}$

LOCAL $IsValidUnsubscribeRequest(m) \triangleq \text{TRUE}$

LOCAL $IsValidUnsubscribeResponse(m) \triangleq \text{TRUE}$

LOCAL $IsValidMessage(m) \triangleq$
 $\vee \wedge m.type = SubscribeRequest$
 $\wedge IsValidSubscribeRequest(m)$
 $\vee \wedge m.type = SubscribeResponse$
 $\wedge IsValidSubscribeResponse(m)$
 $\vee \wedge m.type = UnsubscribeRequest$
 $\wedge IsValidUnsubscribeRequest(m)$
 $\vee \wedge m.type = UnsubscribeResponse$
 $\wedge IsValidUnsubscribeResponse(m)$

VARIABLE *servers*, *conns*

INSTANCE *Messaging* WITH *Nil* \leftarrow "<nil>", *IsValid* \leftarrow *IsValidMessage*

VARIABLES *e2tServers*, *e2tConns*

$E2T \triangleq$ INSTANCE *E2T* WITH
 servers \leftarrow *e2tServers*,
 conns \leftarrow *e2tConns*,
 SubscribeRequest \leftarrow "SubscribeRequest",
 SubscribeResponse \leftarrow "SubscribeResponse",
 UnsubscribeRequest \leftarrow "UnsubscribeRequest",
 UnsubscribeResponse \leftarrow "UnsubscribeResponse",
 ControlRequest \leftarrow "ControlRequest",
 ControlResponse \leftarrow "ControlResponse"

Init \triangleq
 \wedge *E2AP!Init*
 \wedge *E2T!Init*

Next \triangleq
 \vee *E2AP!Next*
 \vee *E2T!Next*

\ * Modification History
\ * Last modified *Fri Aug 13 06:03:48 PDT 2021* by *jordanhalterman*
\ * Created *Fri Aug 13 04:37:35 PDT 2021* by *jordanhalterman*