Tdoc: 99c55

20th .. 24th September 99

Author: Lucent Technologies
Subject: RANAP protocol principles and error handling
For discussion & action

Introduction

Before a discussion can be made with respect to error handling the basic principles of protocols for UMTS UTRAN must be agreed.
This Tdoc proposes to discuss the basic protocol principles and from this the error handling principles can be defined.
Lucent suggest that once decisions are made then a working document will be produced during the meeting in order to reflect the discussion and decide about text within a section for RANAP. Lucent believe that the principles decided upon for Iu would also reflect for the use on Iub and Iur and thus the results of the discussion have wider scope. Lucent believe that the discussion may be better done in the Iu SWG and the results forwarded to the Iub/Iur for acceptance. In this way we will have a common error handling policy for all interfaces.

Presently the only principles defined are the use of EP and a single error handling procedure for all procedures.

RANAP presently defines EP and RANAP functional procedures. One or more EP makes up a RANAP functional procedure.

EP and RANAP functional procedures may occur on either connection oriented and connectionless SCCP service.

This Tdoc outlines the principles for error handling issues. However, in order that we can decide what functionality and thus what error handling is needed, we need to split error handling into categories:

- Basic error handling;
  This falls in to: Syntax errors; Syntactic transfer errors; & Semantic errors.
- Backward compatibility;
- & Forward compatibility.

Elementary Procedure Protocol handling

Is it necessary for backward compatibility to be able to change one EP to another EP type, in the case where a Class 2 becomes a Class 1 or 3?

Is it necessary that an unknown procedure be able to respond to an unknown Class 1 or 3?

These questions need to be asked, discussed and documented as it would seem that this is the basis upon the error handling can eventually be defined, for the treatment of unknown messages for both connection orientated and connectionless procedures.

Protocol IE handling

Should it be possible that a receiving entity decode and use information which previously wasn't available in a message?
This feature should be discussed as this would allow considerable flexibility in the protocol for forward compatibility. If this is the case the sending entity would have to know what was understood in the received message, this is the general philosophy of inter working on the A interface and has defined the way in which the coding/design of the basic protocol.
Conversely should it be possible for a sending entity to know what the receiving entity didn't *understand*.

What means doesn't understand & not supported

One of the major problems on the GSM A interface was to know the difference between these two concepts.
They should not be confused for UMTS and the behavior of a receiving entity when indicating to the sender should be specified in order to not cause any confusion for the sender.
Below is an analysis of the receivers requirement:

- If a receiver understands the coding of a message or IE but doesn't support the operation specified, then the receiver responds indicating that it is not supported.
- If a receiver doesn't understand the coding of the message or IE but doesn't support the service then the receiver responds indicating that it is not supported.
- If the receiver doesn't understand the coding of the message or IE but doesn't need to support service then the receiver responds indicating that the message or IE is not understood.

The second bullet is an interesting concept as this will/may require a form on comprehension required mechanism in the protocol – this is for discussion.

Comprehension required concept

Lucent supports the concept of comprehension required and believes that it should be supported in the protocol to give freedom to the transmitter to construct the protocols as it wishes.
Comprehension required allows a sender of a message to build a message where the receiver is required to indicate that the sent message/IE was not supported (see above).

Do we need this type of flexibility?

Protocol version concept

Obviously, the adoption of a protocol concept in the protocol has a significant effect on complexity for error handling. Lucent's point of view, is that it is against the concept of a protocol version. Below is an idealized protocol concept which Lucent believes can be adopted for Iu.

Transmitter entity

The transmitter should be able to talk to all network nodes with a single version of the protocol. This means that the message built by the sender is independent of what the node has implemented.

Only when there is significant shift in functionality and the protocols have to change will the transmitter need to behave differently. This may be controlled by O&M parameter or learnt by the Transmitter, this is implementation dependent and shall not be specified.

The transmitter will build messages in order that the built message is always backward compatible. This will be achieved by either duplicating Ies where the receiver throws away those Ies that are not understood (see above) or by introducing new Ies which are automatically discarded by the receiver if not supported (se comprehension required concept).

Receiver entity

The receiver shall attempt to decode all Ies within the message. If having completely decoded the message the receiver deems that the information is not enough to fulfill the procedure then it will (if otherwise advised by the CN, via comprehension required mechanism) perform the procedure.