TSG-RAN Meeting #22 **RP-030626**
Maui, USA, 09-12 December 2003

Title: **25.921 Rel-5 CR**

Source: **TSG-RAN WG2**

Agenda item: **7.3.5**

Spec	CR	Rev	Phase	Subject	Cat	Version-Current	Version-New	Doc-2nd-Level	Workitem
25.921	048	-	Rel-5	Incorporation of PCAP	F	5.2.0	5.3.0	R2-032610	TEI5

3GPP TSG-RAN WG3 Meeting #38 **Tdoc ⌘R2-032610**
Sophia-Antipolis, France, 6th – 10th October 2003

CR-Form-v7

CHANGE REQUEST

⌘ **25.921** **CR** **048** ⌘**rev** **-** ⌘ Current version: **5.2.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network **X** Core Network ☐

Title:	⌘	Incorporation of PCAP
Source:	⌘	RAN WG2
Work item code:	⌘	TEI5 ***Date:*** ⌘ 6-10/10/2003
Category:	⌘	**F** ***Release:*** ⌘ Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	PCAP is missing in spec
Summary of change:	⌘	Addition of PCAP to the respective protocol description
Consequences if not approved:	⌘	PCAP remains missing in spec

Clauses affected:	⌘	1, 9.1a, 9.1a.1.1, 9.1a.1.1.7, 9.1a.2, 9.1a.3.1, 9.2, 10, 10.5, 10.5.1, 10.5.2, 10.5.3.1, 10.5.4, 11.1

	Y	N		
Other specs affected: ⌘		X	Other core specifications	⌘
		X	Test specifications	
		X	O&M Specifications	

Other comments: ⌘

How to create CRs using this form:
Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.
2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.
3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

1 1 Scope

The present document provides a guideline for protocol specification of UMTS stage 2 and 3 including the usage of formal languages and rules for error handling. The present document covers control-plane and user-plane protocols specified in TSG-RAN such as RRC, RLC, RANAP, RNSAP, NBAP, PCAP, and SABP.

/* partly omitted */

1.1 9.1a Tabular description of messages and IEs in RANAP, RNSAP, NBAP, PCAP, and SABP

1.1.1 9.1a.1 Message description

A 'Message description' subclause includes one subclause per message.

A message is described with, in this order:

- a table describing a list of information elements;
- explanatory clauses, mainly for describing textually conditions for presence or absence and range bounds for some IEs/IE groups.

1.1.1.1 9.1a.1.1 The Information Element table

The table used in RANAP, RNSAP, NBAP, PCAP, and SABP is composed of 7 columns, labelled and presented as shown below.

IE/Group Name	Presence	Range	IE type and reference	Semantics description	Criticality	Assigned Criticality

NOTE: Indentations are used to visualise the embedding level of an "IE/Group".

Indentations are explicitly written with the character ">" as well as by use of ruler indentations, one per level of indentation. Indentations of lines can be found in the IE/Group Name column.

Each line corresponds either to an IE or to a IE group. An IE group includes all the IEs in following lines until, and not including, a line with the same indentation as the group line.

/* partly omitted */

1.1.1.1.1 9.1a.1.1.7 Assigned Criticality column

This column provides the actual criticality information as defined in subclause 10.3.2 in RANAP, RNSAP, NBAP, PCAP, and SABP.

If an IE/IE group is not understood or missing, the receiving node shall take different actions depending on the value of the Criticality Information. The three possible values of the Criticality Information for an IE/IE group are:

1. Reject IE;
2. Ignore IE and Notify Sender;
3. Ignore IE.

1.1.1.2 9.1a.1.2 Explanatory clauses

This includes the subclauses needed to elaborate conditions and symbolic names (e.g., range bounds). There must be one explanatory clause for each named condition, and for each symbolic name. The text must give the information sufficient to decide whether the IE/IE group is to be included or not, or the value of the symbolic name. The text shall be given in separate tables for the conditions and range bounds.

1.1.2 9.1a.2 IE type description

This describes IE types referred elsewhere, either in the description of a message or in the description of another IE type. The description of an IE type must be as generic as possible, i.e. independent of any specific use. A type should as far as possible not be defined in multiple places in a specification.

An 'IE description' subclause includes one subclause per IE type.

The description of an IE type is done as a table similar to that used for the description of messages.

In RANAP, RNSAP, NBAP, PCAP, and SABP this table has the layout as shown below.

IE/Group Name	Presence	Range	IE type and reference	Semantics description	Criticality	Assigned Criticality

The different columns are filled as message description columns are filled with the addition that in the IE Type and reference column also the use of a basic types defined in subclause 9.2 in the present document is allowed. These basic types shall be considered as pre-defined and for those no reference is necessary. For IE type descriptions, explanatory clauses should also be used as described in subclause 9.1a.1.2. The inclusion of the criticality and assigned criticality columns is optional, but shall be included if separate criticality needs to be indicated.

1.1.3 9.1a.3 Extension for further releases

1.1.3.1 9.1a.3.1 Basic principle

Added elements or choice branches are included where they fit most naturally according to their semantics if the ASN.1 allows (e.g. there exist an extension container in this place). For further information on handling of extensions in RANAP, RNSAP, NBAP, PCAP, and SABP see subclause 10.5.

1.2 9.2 Basic types

To reduce the text in tabular descriptions, some basic abstract types of IE are defined in the present document.

NOTE: The tabular description in this subclause used to describe different formats of the basic types follow the layout applicable to RRC. However, the basic types as such are applicable also to RANAP, RNSAP, NBAP, PCAP, and SABP.

/* partly omitted */

10 Usage of ASN.1

The following clauses contain guidelines for specification of protocol messages with ASN.1.The purpose of ASN.1 is to make it possible to specify message contents description of a message (i.e. what is the contents of a message) separately from its transfer syntax (i.e. how a message is encoded for transmission).

The clause 11 specifies how message transfer syntax is specified. It should be noted that importance of some transfer syntax properties must be determined early during specification because of their effect on message contents description specification possibilities. The properties are **compactness** and **extensibility.** If extreme compactness is required then extensibility must be restricted. If good extensibility is required then compromises must be done regarding compactness. The sections concerning these issues are marked in the following clauses as COMPACTNESS and EXTENSIBILITY.

Identifiers that could be keywords of some language (e.g.: SDL, C, ASN.1, JAVA, C++, ...) should be avoided.

In the current version of the ASN.1 specifications, user-defined constraints are not used.

RANAP, ~~SABP,~~ RNSAP, NBAP, PCAP, and ~~NBAP~~ SABP specifications refers to the 1997 versions of X.680, X.681 and X.691, but the protocols shall only make use of the feature set available in the 1994 versions of X.680, X.681 and X.691.

/* partly omitted */

10.5 Extensions for future releases in RANAP, RNSAP, NBAP; PCAP and SABP

The following clauses contain rules for extension mechanisms of ASN.1 for RANAP, ~~SABP,~~ RNSAP, NBAP, PCAP, and ~~NBAP~~SABP. The purpose of these rules is to guarantee backward compatibility for ASN.1.

1.2.1 10.5.1 Allowed Extension

The allowed extension for ASN.1 description in RANAP, ~~SABP,~~ RNSAP, NBAP, PCAP, and ~~NBAP~~ SABP are:

1) adding New IEs or IE groups which should be achieved by using the protocol extension container (extension by using of ellipsis notation (...) should be avoided) for:

   - adding at the top level of message; and
   - adding in the SEQUENCE type,

2) extending the range of already define IEs which has ellipsis notation(...);

3) changing the assigned criticality information of already defined IEs; and

4) adding new IEs of IE groups after ellipsis notation (...) in the CHOICE type if the ellipsis notation (...) is present.

1.2.2 10.5.2 Not Allowed Extension

The not allowed extension for ASN.1 description in RANAP, ~~SABP,~~ RNSAP, NBAP, PCAP, and ~~NBAP~~ SABP are:

1) deleting the already defined IEs or IE groups when no individual criticality information is defined;

2) adding or deleting the criticality information of existing IEs;

3) deleting the already defined values in the ASN.1 type. Instead, a semantic description is added in order to clarify the behaviour; and

4) changing the presence of already defined IEs with no assigned criticality.

This is because above changes do not guarantee the backward compatibility.

1.2.3 10.5.3 Recommendations for extensions for further releases

1.2.3.1 10.5.3.1 General

This subclause gives recommendations for future extensions in versions of the RANAP, RNSAP, NBAP, PCAP, and SABP where non-backward compatible changes are not acceptable.

1.2.3.2 10.5.3.2 Usage of Presence and Assigned Criticality in Future Releases

1.2.3.2.1 10.5.3.2.1 New Procedures

For procedures introduced when the backward compatibility mechanisms are taken into use the following recommendation applies to the Assigned Criticality of the procedure (in the tabular description of messages visible as the Assigned Criticality of the IE Message Type).

Assigned Criticality	Recommendation	Typical usage
Ignore	Should be used if: - the sender does not care whether or not the procedure is supported; or - if the sender "already knows" that the procedure is supported.	Typically used for procedures where: - the sender do not care whether or not the procedure is supported; or - where the usage is dependent on previously exchanged information.
Ignore and Notify	Should be used if: - the sender does not care whether or not the procedure is supported; or - if the sender "already knows" that the procedure is supported; but need to know whether or not the procedure was understood.	Typically not used.
Reject	Should be used if: - the procedure shall be rejected when not supported.	Typically used for new procedures where the sender has no prior knowledge on whether or not the procedure will be understood.

1.2.3.2.2 10.5.3.2.2 New IEs

For new IEs introduced when the backward compatibility mechanisms are taken into use the following recommendation applies to the Assigned Criticality of the IE.

Presence	Assigned Criticality	Recommendation	Typical usage
Optional	Ignore	Should be used if the sender does not care whether or not the function related to the IE is supported.	Typically used for "non core" features (specification text; "… shall, if supported, …".
	Ignore and Notify	Should be used if: - the sender does not care whether or not the function related to the IE is supported. but need to know whether or not the IE was understood.	Typically used for "non core" features (specification text; "… shall, if supported, …").
	Reject	Should be used if: - the alternative to executing the feature related to the IE is rejecting the procedure.	Typically used for "core" features (specification text; "… shall …").
Mandatory / Conditional	Ignore	Should be used for "core" features where: - it is essential that all implementations of future releases support the feature related to the IE; - it is possible to inter-work with nodes implementing older releases (not understanding the IE related to the feature and consequently not supporting the feature).	Typically not used. Note 1.
	Ignore and Notify	Should be used for "core" features where: - it is essential that all implementations of future releases support the feature related to the IE; - it is possible to inter-work with nodes implementing older releases (not understanding the IE related to the feature and consequently not supporting the feature); but the sending node need to know whether or not the IE was understood.	Typically not used. Note 1.
	Reject	Should be used for "core" features where: - it is essential that all implementations of future releases support the feature related to the IE; - it is not possible to inter-work with nodes implementing older releases (not supporting the feature).	Typically not used. Note 2.

NOTE 1: This combination (presence + assigned criticality) could be used as an intermediate state, i.e. when the Assigned Criticality is expected/planned to be changed to "Reject" in the future.

NOTE 2: This combination (presence + assigned criticality) should be avoided since it prevents inter-working with older version of a specification.

1.2.3.2.3 10.5.3.2.3 Changing the Presence of an IE

The Presence can always be changed in future version of a specification.

NOTE: Mandatory and Conditional IEs with Assigned Criticality "Reject" will still cause rejection when missing in a node based on a previous version of the specification (even though changed to Optional).

Recommendation:

The Presence of Mandatory IEs with Assigned Criticality "Reject" should not be changed in future versions of a specification.

The Presence of Conditional IEs with Assigned Criticality "Reject" should not be changed in future versions of a specification, unless it is also ensured that the condition will not result in a requirement to include the IE.

1.2.3.2.4 10.5.3.2.4 Changing the Assigned Criticality of an IE

The Assigned Criticality can always be changed in future version of a specification.

NOTE: The behaviour for missing IEs will remain unchanged when inter-working with a node based on a previous version of the specification.

Recommendation:

When changing the Assigned Criticality of Mandatory and Conditional IEs with Assigned Criticality "Reject" in future versions of a specification special attention should be paid to inter-working between different versions of the specification.

1.2.3.2.5 10.5.3.2.5 Removing IEs

Any IE (with Assigned Criticality) can be removed in future version of a specification.

NOTE: Mandatory and Conditional IEs with Assigned Criticality "Reject" will still cause rejection when missing in a node based on a previous version of the specification (even though changed to Optional).

Recommendation:

Mandatory IEs with Assigned Criticality "Reject" should not be removed in future versions of a specification.

Conditional IEs with Assigned Criticality "Reject" should not be removed in future versions of a specification, unless it is also ensured that the condition, if evaluated for the message where the IE is removed by a node based on a previous version of the specification, will not result in a requirement to include the IE.

1.2.4 10.5.4 Use of extensions

The following rules apply to the use of extensions in RANAP, ~~SABP,~~ RNSAP, NBAP, PCAP, and ~~NBAP~~SABP:

1) IEs added in the extension containers shall be added before the ellipsis notation.

2) IEs added in the extension containers shall be placed in the extension containers in the chronological order of their introduction e.g. a late correction of Release '99 may appear after Release 4 corrections in a Release 4 version of the specification.

3) When new values are added into an ENUMERATED type after the ellipsis notation in the later release (e.g. Release 4), the same amount of new values need to be added as "dummy" in the same IE in the correspondent older release of the specification (e.g. Release '99).

   The "dummy" has to be added into the older release (e.g. Release '99) of the specifications when needed, i.e. if the later release (e.g. Release 4) of specification is considered as not stable, the "dummy" shall not be added in the same IE in the correspondent older release (e.g. Release '99) of the specification. When the older release (e.g. Release '99) adds new values for later version (e.g. Release '99 June version), the same values shall be added in the same position of the correspondent later release (e.g. Release 4) in order to have backward compatibility when in the future the later release (e.g. Release 4) is considered as stable. Further explanation with an example is shown below:

   - Release '99 June version: *Example* ::= ENUMERATED {a, b,…};
   - Release 4 June version: *Example* ::= ENUMERATED {a, b,…, c, d};

NOTE 1: In the June version the Release 4 is considered as not stable, therefore no "dummy" was added in Release '99).

   - Release '99 September version, *Example* ::= ENUMERATED {a, b,…, e};
   - Release 4 September version, *Example* ::= ENUMERATED {a, b,…, e, c, d};

NOTE 2: In the September version the value "e" was added in Release '99, the value "e" shall be added at the same position in Release 4 since the Release 4 is yet considered as not stable.

   - Release '99 December version, *Example* ::= ENUMERATED {a, b,…, e, dummy1, dummy2, f};
   - Release 4 December version, *Example* ::= ENUMERATED {a, b,…, e, c, d, f}.

NOTE 3: In the December version, when a new value "f" has to be added in Release '99 and the Release 4 is considered as stable, "dummy1" and "dummy2" are added.

4) When new choice tags are added into a CHOICE type after the ellipsis notation, new tags shall need to be added in the protocol container.

- Any addition of a choice tag in Release x for a CHOICE not existing in Release y (y<x) shall be made by introducing a normal choice tag (not included in the protocol container) after the ellipses if no release z (z >= x) has included a protocol container for this CHOICE yet. Further explanation with an example is shown below:
  - Release '99 June version, *Example* ::= CHOICE {a, b,…};
  - Release 4 June version, *Example* ::= CHOICE {a, b,…};
  - Release '99 September version, *Example* ::= CHOICE {a, b,…, c};
  - Release 4 September version, *Example* ::= CHOICE {a, b,…, c}.
- Any addition of a choice tag in Release x for a CHOICE already existing in Release y (y<x) shall be made inside a protocol container after the ellipses. Further explanation with an example is shown below:
  - Release '99 June version: *Example* ::= CHOICE {a, b,…};
  - Release 4 June version: *Example* ::= CHOICE {a, b,…};
  - Release '99 September version: *Example* ::= CHOICE {a, b,…};
  - Release 4 September version: *Example* ::= CHOICE {a, b,…, protocol container {d}}.
- Any addition of a choice tag in Release x for a CHOICE already containing a protocol container in Release y (y>x) shall be made inside a protocol container after the ellipses. Further explanation with an example is shown below:
  - Release '99 June version: *Example* ::= CHOICE {a, b,…};
  - Release 4 June version: *Example* ::= CHOICE {a, b,…, protocol container {d}};
  - Release '99 September version: *Example* ::= CHOICE {a, b,…, protocol container {e}};
  - Release 4 September version: *Example* ::= CHOICE {a, b,…, protocol container {d, e}}.
- If any Release has already included a protocol container in this CHOICE, then all future changes to this CHOICE shall be made by introducing IE's in the protocol container. Further explanation with an example is shown below:
  - Release 4 June version: *Example* ::= CHOICE {a, b,…, protocol container {d, e}};
  - Release 4 September version: *Example* ::= CHOICE {a, b,…, protocol container {d, e, f}}.

/* partly omitted */

2 11 Message transfer syntax specification

2.1 11.1 Selection of transfer syntax specification method

For RRC Basic Packed Encoding Rules (BASIC-PER) PER Unaligned Variant and possible use of specialised encoding is chosen.

For RANAP, RNSAP, NBAP, PCAP and SABP Basic Packed Encoding Rules (BASIC-PER) Aligned Variant is chosen.

/* partly omitted */