

ETSI TS 125 419 V3.3.0 (2000-12)

Technical Specification

**Universal Mobile Telecommunications System (UMTS);
UTRAN Iu Interface: Service Area Broadcast Protocol SABP
(3GPP TS 25.419 version 3.3.0 Release 1999)**



Reference

RTS/TSGR-0325419UR3

Keywords

UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at <http://www.etsi.org/tb/status/>

If you find errors in the present document, send your comment to:
editor@etsi.fr

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2000.

All rights reserved.

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<http://www.etsi.org/ipr>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by the ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under www.etsi.org/key.

Contents

Foreword	6
1 Scope	7
2 References	7
3 Definitions and abbreviations	7
3.1 Definitions	7
3.2 Abbreviations	8
4 General	8
4.1 Specification Notations	9
5 Services provided by SABP	9
6 Services expected from the Transport layer	9
7 Functions of SABP	9
8 SABP Procedures	10
8.1 Elementary Procedures	10
8.2 Write-Replace	10
8.2.1 General	10
8.2.2 Successful Operation	10
8.2.3 Unsuccessful Operation	11
8.2.4 Abnormal Conditions	11
8.3 Kill	11
8.3.1 General	11
8.3.2 Successful Operation	12
8.3.3 Unsuccessful Operation	12
8.3.4 Abnormal Conditions	12
8.4 Load Status Enquiry	12
8.4.1 General	12
8.4.2 Successful Operation	13
8.4.3 Unsuccessful Operation	13
8.4.4 Abnormal Conditions	13
8.5 Message Status Query	13
8.5.1 General	13
8.5.2 Successful Operation	14
8.5.3 Unsuccessful Operation	14
8.5.4 Abnormal Conditions	15
8.6 Reset	15
8.6.1 General	15
8.6.2 Successful Operation	15
8.6.3 Unsuccessful Operation	15
8.6.4 Abnormal Conditions	16
8.7 Restart Indication	16
8.7.1 General	16
8.7.2 Successful Operation	16
8.7.3 Abnormal Conditions	16
8.8 Failure Indication	16
8.8.1 General	16
8.8.2 Successful Operation	17
8.8.3 Abnormal Conditions	17
8.9 Error Indication	17
8.9.1 General	17
8.9.2 Successful Operation	17
8.9.3 Abnormal Conditions	18

9	Elements for SABP Communication.....	18
9.1	Message Functional Definition and Content	18
9.1.1	General.....	18
9.1.2	Message Contents	18
9.1.2.1	Presence	18
9.1.2.2	Criticality	18
9.1.2.3	Range	18
9.1.2.4	Assigned Criticality.....	19
9.1.3	WRITE-REPLACE.....	19
9.1.4	WRITE-REPLACE COMPLETE.....	19
9.1.5	WRITE-REPLACE FAILURE.....	19
9.1.6	KILL	19
9.1.7	KILL COMPLETE	20
9.1.8	KILL FAILURE	20
9.1.9	LOAD QUERY	20
9.1.10	LOAD QUERY COMPLETE	20
9.1.11	LOAD QUERY FAILURE.....	21
9.1.12	MESSAGE STATUS QUERY	21
9.1.13	MESSAGE STATUS QUERY COMPLETE.....	21
9.1.14	MESSAGE STATUS QUERY FAILURE	21
9.1.15	RESET	22
9.1.16	RESET COMPLETE.....	22
9.1.17	RESET FAILURE	22
9.1.18	RESTART	22
9.1.19	FAILURE	23
9.1.20	ERROR INDICATION	23
9.2	Information Element Definitions.....	23
9.2.1	MessageType	23
9.2.2	Broadcast Message Content.....	24
9.2.3	Serial Number	24
9.2.4	Old Serial Number	24
9.2.5	New Serial Number	25
9.2.6	Service Areas List.....	25
9.2.7	Category.....	25
9.2.8	Repetition Period	25
9.2.9	No of Broadcasts Requested	25
9.2.10	No of Broadcasts Completed List.....	26
9.2.11	Service Area Identifier.....	26
9.2.12	Failure List.....	26
9.2.13	Radio Resource Loading List	27
9.2.14	Cause	27
9.2.15	Data Coding Scheme	30
9.2.16	Recovery Indication.....	30
9.2.17	Criticality Diagnostics	31
9.2.18	Available Bandwidth	31
9.2.19	Message Identifier	31
9.3	Message and Information Element Abstract Syntax (with ASN.1).....	33
9.3.0	General.....	33
9.3.1	Usage of protocol extension mechanism for non-standard use.....	33
9.3.2	Elementary Procedure Definitions	34
9.3.3	PDU Definitions	38
9.3.4	Information Element Definitions	47
9.3.5	Common Definitions	52
9.3.6	Constant Definitions	52
9.3.7	Container Definitions.....	54
9.4	Message Transfer Syntax	56
10	Handling of Unknown, Unforeseen or Erroneous Protocol Data.....	57
10.1	General	57
10.2	Transfer Syntax Error.....	57
10.3	Abstract Syntax Error.....	57
10.3.1	General.....	57

10.3.2 Criticality Information57
10.3.3 Presence Information58
10.3.4 Not comprehended IE/IE group58
10.3.4.1 Procedure Code.....58
10.3.4.2 IEs other than the Procedure Code.....59
10.3.5 Missing IE or IE group59
10.3.6 IEs or IE groups received in wrong order or with too many occurrences.....60
10.4 Logical Error61

Annex A (informative): Change history62

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document specifies the *Service Area Broadcast Protocol (SABP)* between the Cell Broadcast Centre (CBC) and the Radio Network Controller (RNC). It fulfils the CBC - RNC communication requirements specified in [5] and is defined over the Iu-BC – reference point.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

- [1] 3GPP TR 23.930: "Iu Principles".
- [2] 3GPP TS 25.410: "UTRAN Iu Interface; General Aspects and Principles".
- [3] 3GPP TS 25.401: "UTRAN Overall Description".
- [4] 3GPP TS 25.931: "UTRAN Functions, Examples on Signalling Procedures".
- [5] 3GPP TS 23.041: "Technical realization of Cell Broadcast Service (CBS)".
- [6] 3GPP TS 25.414: "UTRAN Iu Interface Data Transport and Transport Signalling".
- [7] ITU-T Recommendation X.680 (12/94): "Information Technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".
- [8] ITU-T Recommendation X.681 (12/94): "Information Technology - Abstract Syntax Notation One (ASN.1): Information object specification".
- [9] ITU-T Recommendation X.691 (12/94): "Information Technology - ASN.1 encoding rules - Specification of Packed Encoding Rules (PER)".
- [10] 3GPP TR 25.921: "Guidelines and Principles for Protocol Description and Error Handling".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

Elementary Procedure: The SABP consists of Elementary Procedures (EPs). An Elementary Procedure is a unit of interaction between the CN (CBC) and the RNC. These EPs are defined separately and are intended to be used to build up complete sequences in a flexible manner. If the independence between some EPs is restricted, it is described under the relevant EP description. Unless otherwise stated by the restrictions, the EPs may be invoked independently of each other as stand alone procedures, which can be active in parallel. Examples on using several SABP EPs together with each other and EPs from other interfaces can be found in reference [4].

An EP consists of an initiating message and possibly a response message. Two kinds of EPs are used:

- **Class 1:** Elementary Procedures with response (success or failure).

- **Class 2:** Elementary Procedures without response.

For Class 1 EPs, the types of responses can be as follows:

Successful

- A signalling message explicitly indicates that the elementary procedure successfully completed with the receipt of the response.

Unsuccessful

- A signalling message explicitly indicates that the EP failed.
- On time supervision expiry (i.e. absence of expected response).

Class 2 EPs are considered always successful.

Message Reference: This is defined as consisting of the following parameters: Message Identifier, Serial Number, and SAI (Service Area Identifier).

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CBC	Cell Broadcast Centre
CBS	Cell Broadcast Service
CN	Core Network
EP	Elementary Procedure
FP	Frame Protocol
PDU	Protocol Data Unit
RNC	Radio Network Controller
SA	Service Area
SABP	Service Area Broadcast Protocol

4 General

The protocol described in the present document is the protocol between CN (CBC) and RNC needed for the CBC Application. The CBC Application is described in [5].

4.1 Procedure Specification Principles

The principle for specifying the procedure logic is to specify the functional behaviour of the RNC exactly and completely. The CN functional behaviour is left unspecified.

The following specification principles have been applied for the procedure text in chapter 8:

- The procedure text discriminates between:
 - 1) Functionality which "shall" be executed

The procedure text indicates that the receiving node "shall" perform a certain function Y under a certain condition. If the receiving node supports procedure X but cannot perform functionality Y requested in the REQUEST message of a Class 1 EP, the receiving node shall respond with the message used to report unsuccessful outcome for this procedure, containing an appropriate cause value.

- 2) Functionality which "shall, if supported" be executed

The procedure text indicates that the receiving node "shall, if supported," perform a certain function Y under a certain condition. If the receiving node supports procedure X, but does not support functionality Y, the receiving node shall proceed with the execution of the EP, possibly informing the requesting node about the not supported functionality.

- Any required inclusion of an optional IE in a response message is explicitly indicated in the procedure text. If the procedure text does not explicitly indicate that an optional IE shall be included in a response message, the optional IE shall not be included.

4.2 Forwards and Backwards Compatibility

The forwards and backwards compatibility of the protocol is assured by mechanism where all current and future messages, and IEs or groups of related IEs, include Id and criticality fields that are coded in a standard format that will not be changed in the future. These parts can always be decoded regardless of the standard version.

4.3 Specification Notations

For the purposes of the present document, the following notations apply:

Procedure	When referring to an elementary procedure in the specification the Procedure Name is written with the first letters in each word in upper case characters followed by the word "procedure", e.g. Write-Replace procedure.
Message	When referring to a message in the specification the MESSAGE NAME is written with all letters in upper case characters followed by the word "message", e.g. WRITE-REPLACE message.
IE	When referring to an information element (IE) in the specification the <i>Information Element Name</i> is written with the first letters in each word in upper case characters and all letters in Italic font followed by the abbreviation "IE", e.g. <i>Old Serial Number</i> IE.
Value of an IE	When referring to the value of an information element (IE) in the specification the "Value" is written as it is specified in subclause 9.2 enclosed by quotation marks, e.g. "Abstract Syntax Error (Reject)" or "Background".

5 Services provided by SABP

- During normal operation the CN (CBC) initiates all message transfer and query operations. The RNC responds to the message transfer and query operations initiated by the CBC.
- The RNC will open the connection only in case an error (Failure Indication Procedure) or recovery (Restart Indication Procedure) is to be reported.
- The initiator of a connection is responsible for the termination of the connection.

6 Services expected from the Transport layer

Following service is expected from the transport layer:

- in sequence delivery of FP PDU [6].

7 Functions of SABP

The SABP has the following functions:

- Message Handling. This function is responsible for the broadcast of new messages, amend existing broadcasted messages and to stop the broadcasting of specific messages.
- Load Handling. This function is responsible for determining the loading of the broadcast channels at any particular point in time.
- Reset. This function permits the CBC to end broadcasting in one or more Service Areas.

- Error Handling. This function allows the reporting of general error situations, for which function specific error messages have not been defined.

These functions are implemented by one or several SABP elementary procedures described in the following clauses.

8 SABP Procedures

8.1 Elementary Procedures

In the following tables, all EPs are divided into Class 1, and Class 2 Procedures:

Table 1: Class 1

Elementary Procedure	Initiating Message	Successful Outcome	Unsuccessful Outcome
		Response message	Response message
Write-Replace	WRITE-REPLACE	WRITE-REPLACE COMPLETE	WRITE-REPLACE FAILURE
Kill	KILL	KILL COMPLETE	KILL FAILURE
Status Load Enquiry	LOAD QUERY	LOAD QUERY COMPLETE	LOAD QUERY FAILURE
Status Message Query	MESSAGE QUERY	MESSAGE QUERY COMPLETE	MESSAGE QUERY FAILURE
Reset	RESET	RESET COMPLETE	RESET FAILURE

Table 2: Class 2

Elementary Procedure	Message
Restart Indication	RESTART
Failure Indication	FAILURE
Error Indication	ERROR INDICATION

8.2 Write-Replace

8.2.1 General

The purpose of this Write-Replace procedure is to broadcast new information or replace a message already broadcast to a chosen Service Area(s).

8.2.2 Successful Operation

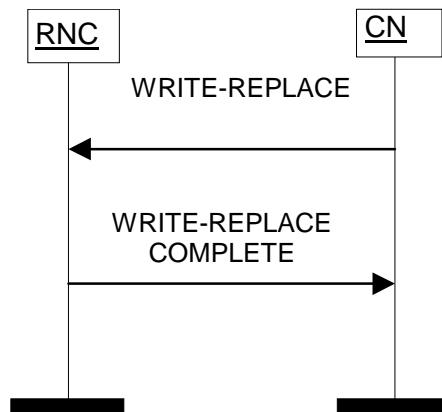


Figure 1: Write-Replace Procedure: Successful Operation

The CN shall initiate the procedure by sending a WRITE-REPLACE message to the RNC.

The presence of a *New Serial Number* IE will indicate that this is a new broadcast. The presence of both the *Old Serial Number* IE and a *New Serial Number* IE will indicate that this message is a replacement of an existing broadcast. The RNC will initiate broadcasting of a new message or replace a message already broadcast as requested.

Upon receipt of the WRITE-REPLACE message the RNC shall respond using the WRITE-REPLACE COMPLETE message containing a *New Serial Number* IE indicating that resources are available as requested for the Service Area(s) specified.

8.2.3 Unsuccessful Operation

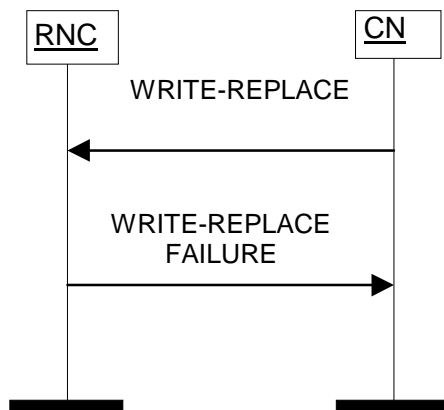


Figure 2: Write-Replace Procedure: Un-Successful Operation

If the RNC cannot allocate all the resources requested for the Service Area(s) specified in the WRITE-REPLACE message, then the RNC shall return a WRITE-REPLACE FAILURE message to the CN. A list of Service Area(s) where the requested resources are unavailable will be provided in this WRITE-REPLACE FAILURE message in the *Failure List* IE.

This WRITE-REPLACE FAILURE message may also include those Service Area(s) where the requested resources were available and will indicate in the *Number of Broadcasts Completed* IE those Service Area(s) which completed the request.

8.2.4 Abnormal Conditions

8.3 Kill

8.3.1 General

The purpose of the Kill procedure is to stop the broadcast of the indicated message.

8.3.2 Successful Operation

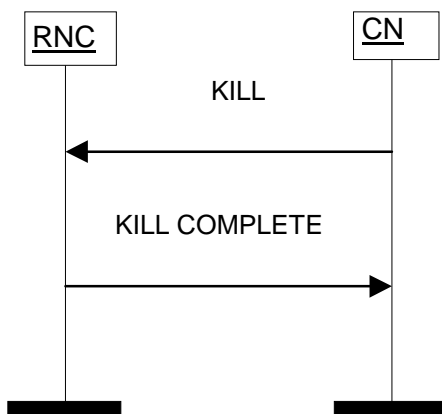


Figure 3: Kill Procedure: Successful Operation

The CN shall initiate the procedure by sending a KILL message to the RNC.

Upon receipt of the KILL message the RNC shall stop broadcasting the indicated message in the indicated Service Area(s). The RNC shall respond using the KILL COMPLETE message, containing the *Old Serial Number* IE.

8.3.3 Unsuccessful Operation

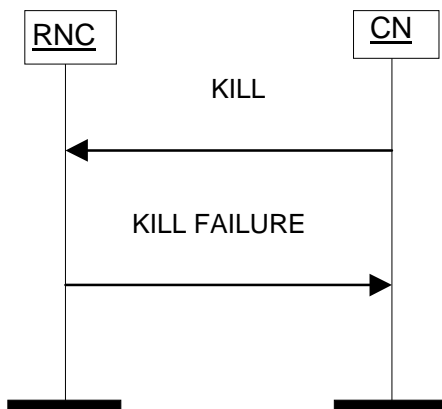


Figure 4: Kill Procedure: Un-Successful Operation

A *Failure List* IE indicating the list of Service Area(s) where the message reference is not valid will be provided in a KILL FAILURE message. This response message may also – if applicable - indicate in the *Number of Broadcasts Completed List* IE those Service Area(s) which completed the request where the KILL message was successful.

8.3.4 Abnormal Conditions

8.4 Load Status Enquiry

8.4.1 General

The purpose of this Load Status Enquiry procedure is to obtain the current permissible bandwidth available for broadcast within particular Service Area(s).

8.4.2 Successful Operation

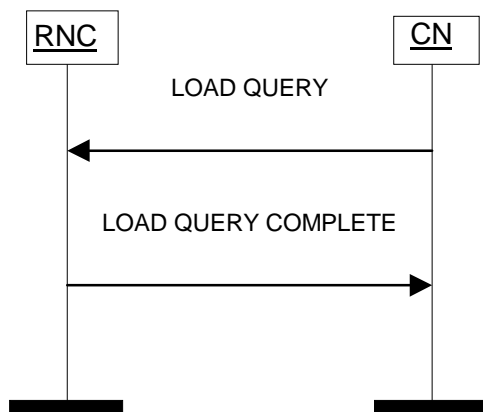


Figure 5: Load Status Enquiry Procedure: Successful Operation

The CN shall initiate the procedure by sending a LOAD QUERY message to the RNC. The message shall include a *Service Areas List* IE. Upon reception of the LOAD QUERY message the RNC shall respond with a LOAD QUERY COMPLETE message.

8.4.3 Unsuccessful Operation

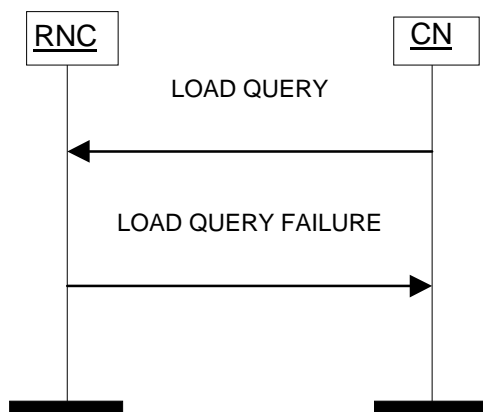


Figure 6: Load Status Enquiry Procedure: Un-Successful Operation

If the RNC contains Service Area(s) for which the RNC was not able to respond to, it shall respond with a LOAD QUERY FAILURE message which includes the *Failure List* IE.

The LOAD QUERY FAILURE response message may – if applicable - also contain a *Radio Resource Loading List* IE for which the LOAD STATUS QUERY reporting was successful.

8.4.4 Abnormal Conditions

8.5 Message Status Query

8.5.1 General

The Message Status Query procedure is used by the CN to obtain the message status of a broadcast message.

8.5.2 Successful Operation

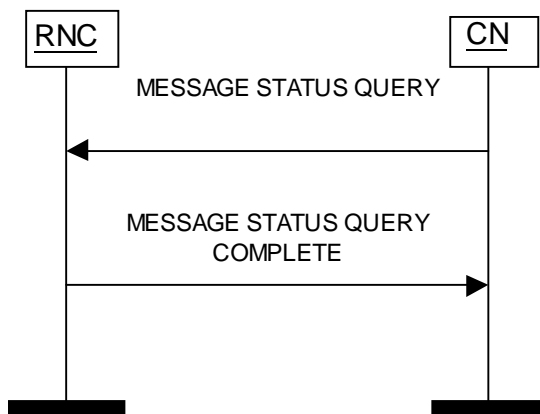


Figure 7: Message Status Enquiry Procedure: Successful Operation

The CN shall initiate the procedure by sending a MESSAGE STATUS QUERY message to the RNC. The message will contain the *Old Serial Number* IE along with the appropriate *Service Areas List* IE.

Upon receipt of the MESSAGE STATUS QUERY message the RNC shall respond using the MESSAGE STATUS QUERY COMPLETE message.

Within this message the *No of Broadcasts Completed List* IE contains each Service Area which successfully performed the requested operation and for each of these Service Area(s), the number of times this broadcast message has been sent to this particular Service Area(s) for broadcast.

8.5.3 Unsuccessful Operation

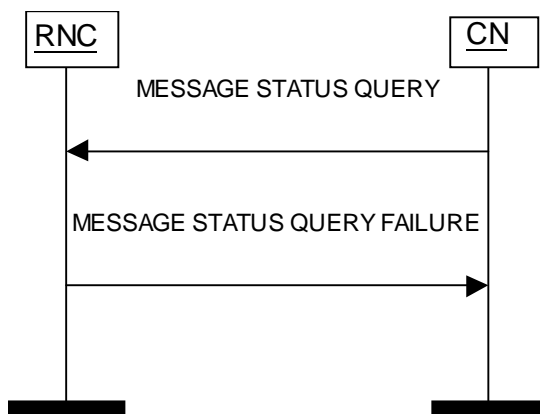


Figure 8: Message Status Enquiry Procedure: Un-Successful Operation

If the requested operation fails (e.g. because the Message Identifier is unknown, or when the RNC cannot send the status for a known Message Identifier) the RNC shall send a MESSAGE STATUS QUERY FAILURE message to the CN containing a *Failure List* IE for Service Area(s) for which the requested operation failed.

The MESSAGE STATUS QUERY FAILURE message may – if applicable - also include the *No of Broadcasts Completed List* IE indicating those Service Area(s) for which the MESSAGE STATUS QUERY message was successful.

8.5.4 Abnormal Conditions

8.6 Reset

8.6.1 General

The purpose of the Reset procedure is to end broadcasting in one or more Service Areas in the RNC.

8.6.2 Successful Operation

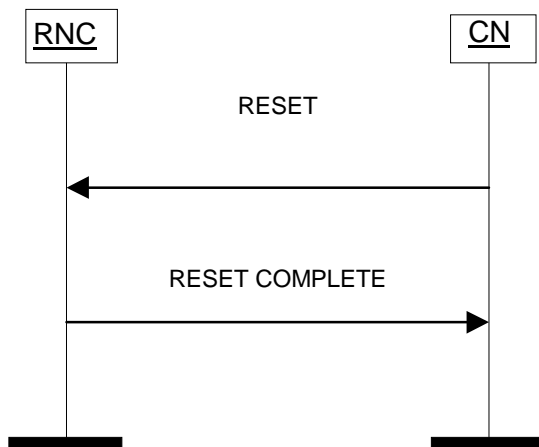


Figure 9: Reset Procedure: Successful Operation

The CN shall initiate the procedure by sending a RESET message to the RNC, in order to end broadcasting in one or more Service Areas of the RNC.

It may also be used by the CN to inquire about the Service Area broadcasting operational state of Service Area(s) who had earlier indicated as having failed.

Upon receipt of this message the RNC shall end broadcasting in the indicated Service Area(s) and shall respond using a RESET COMPLETE message.

8.6.3 Unsuccessful Operation

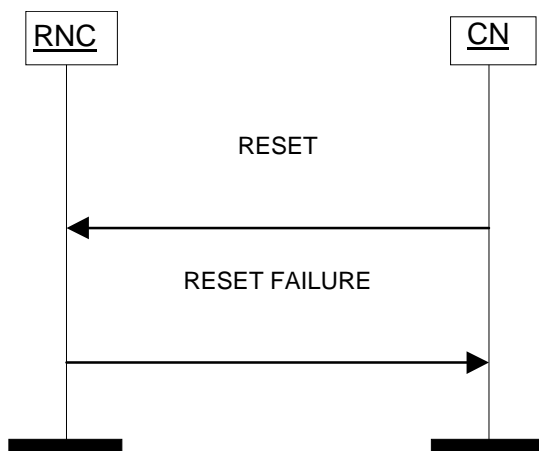


Figure 10: Reset Procedure: Un-Successful Operation

If upon receipt of this message the RNC can not end broadcasting in the indicated Service Area(s), it shall respond using a RESET FAILURE message containing the *Service Areas List* IE indicating the relevant Service Area(s).

The RESET FAILURE message may – if applicable - also include those Service Area(s) for which the RESET message was successful.

8.6.4 Abnormal Conditions

8.7 Restart Indication

8.7.1 General

The purpose of the Restart Indication procedure is for the RNC to indicate to the CN that a Service Area broadcasting related restart situation has occurred in one or more of its Service Areas e.g. when a Service Area becomes operational or when the RNC is initialised.

8.7.2 Successful Operation

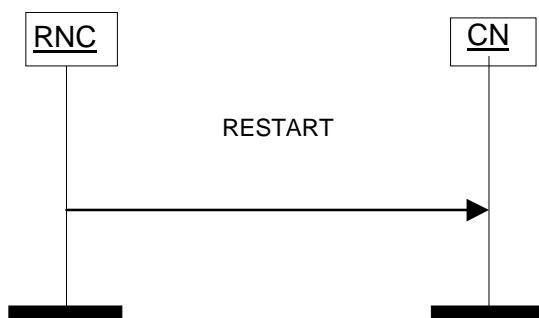


Figure 11: Restart Indication Procedure: Successful Operation

The RNC shall initiate the procedure by sending a RESTART message to the CN. This message shall contain a *Service Areas List* IE for reference and may also include an indication as to whether the previously sent information needs to be re Loaded.

8.7.3 Abnormal Conditions

8.8 Failure Indication

8.8.1 General

The purpose of the Failure Indication procedure is to indicate to the CN from the RNC that a Service Area broadcasting related problem is occurring in one or more of its Service Areas.

8.8.2 Successful Operation

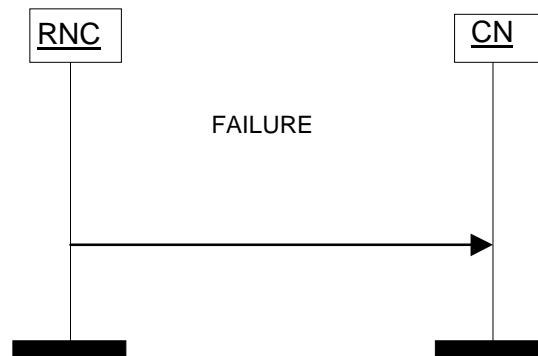


Figure 12: Failure Indication Procedure: Successful Operation

The RNC shall initiate the procedure by sending a FAILURE message to the CN.

Upon receipt of this FAILURE message, the CN will not generate further WRITE or REPLACE messages for these Service Area(s) until the CN is informed by a RESTART message that the Service Area can resume normal Service Area broadcasting operation.

8.8.3 Abnormal Conditions

8.9 Error Indication

8.9.1 General

The Error Indication procedure is used by the RNC to indicate to the CN that a message is not understood, provided it cannot be reported by an appropriate failure message.

8.9.2 Successful Operation

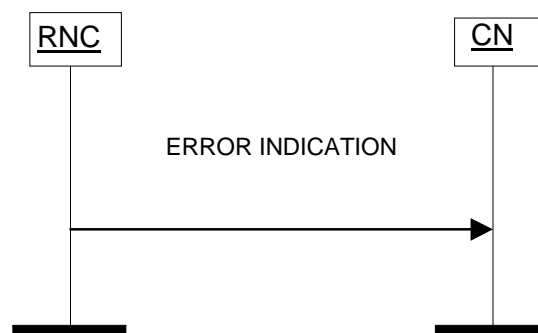


Figure 13: Error Indication Procedure: Successful Operation

The RNC shall initiate the procedure by sending an ERROR INDICATION message to the CN in response to any message that is not understood e.g. invalid parameter or parameter value. This message shall contain information

necessary for the CN to be able to identify which initial message this is in response to, and additional information e.g. Cause Value.

8.9.3 Abnormal Conditions

9 Elements for SABP Communication

9.1 Message Functional Definition and Content

9.1.1 General

NOTE: The messages have been defined in accordance to the guidelines specified in [10].

For each message there is, a table listing the signalling elements in their order of appearance in the transmitted message.

9.1.2 Message Contents

9.1.2.1 Presence

All information elements in the message descriptions below are marked mandatory, optional or conditional according to the following table:

Table 3: Meaning of abbreviations used in SABP messages

Abbreviation	Meaning
M	IE's marked as Mandatory (M) will always be included in the message.
O	IE's marked as Optional (O) may or may not be included in the message.
C	IE's marked as Conditional (C) will be included in a message only if the condition is satisfied. Otherwise the IE is not included.

9.1.2.2 Criticality

Each Information Element or Group of Information Elements may have a criticality information applied to it. Following cases are possible:

Table 4: Meaning of content within "Criticality" column

Abbreviation	Meaning
–	No criticality information is applied explicitly.
YES	Criticality information is applied. This is usable only for non-repeatable IEs
GLOBAL	The IE and all its repetitions together have one common criticality information. This is usable only for repeatable IEs.
EACH	Each repetition of the IE has its own criticality information. It is not allowed to assign different criticality values to the repetitions. This is usable only for repeatable IEs.

9.1.2.3 Range

The Range column indicates the allowed number of copies of repetitive IEs/IE groups.

9.1.2.4 Assigned Criticality

This column provides the actual criticality information as defined in chapter 10.3.2, if applicable.

9.1.3 WRITE-REPLACE

This message is sent by the CN to the RNC.

Direction: CN → RNC

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
New Serial Number	M		9.2.5		yes	ignore
Old Serial Number	O		9.2.4		yes	ignore
Service Areas List	M		9.2.6		yes	ignore
Category	O		9.2.7		yes	ignore
Repetition Period	O		9.2.8		yes	ignore
No of Broadcasts Requested	M		9.2.9		yes	ignore
Data Coding Scheme	M		9.2.15		yes	ignore
Broadcast Message Content	M		9.2.2		yes	ignore

9.1.4 WRITE-REPLACE COMPLETE

This message will be sent by the RNC to the CN in a successful response to a WRITE-REPLACE message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
New Serial Number	M		9.2.5		yes	ignore
No of Broadcasts Completed List	M		9.2.9		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.5 WRITE-REPLACE FAILURE

This message will be sent by the RNC to the CN as an unsuccessful response to a WRITE-REPLACE message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
New Serial Number	M		9.2.5		yes	ignore
Failure List	M		9.2.12		yes	ignore
No of Broadcasts Completed List	O		9.2.10		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.6 KILL

This message is sent by the CN to the RNC to stop broadcasting of a specific message.

Direction: CN → RNC

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
Service Areas List	M		9.2.6		yes	ignore

9.1.7 KILL COMPLETE

This message is sent by the RNC to the CN as a successful response to a KILL message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
No of Broadcasts Completed List	M		9.2.9		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.8 KILL FAILURE

This message is sent by the RNC to the CN as unsuccessful response to a KILL message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
Failure List	M		9.2.12		yes	ignore
No of Broadcasts Completed List	O		9.2.10		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.9 LOAD QUERY

This message is sent by the CN to the RNC to gain an indication of broadcast resources available.

Direction: CN → RNC

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Service Areas List	M		9.2.6		yes	ignore

9.1.10 LOAD QUERY COMPLETE

This message will be sent by the RNC as a successful response to the LOAD QUERY message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Radio Resource Loading List	M		9.2.13		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.11 LOAD QUERY FAILURE

This message is sent by the RNC to the CN as an unsuccessful response to a LOAD QUERY message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Failure List	M		9.2.12		yes	ignore
Radio Resource Loading List	O		9.2.13		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.12 MESSAGE STATUS QUERY

This message is sent by the CN to the RNC to obtain the current status of a Service Area broadcasting message.

Direction: CN → RNC

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
Service Areas List	M		9.2.6		yes	ignore

9.1.13 MESSAGE STATUS QUERY COMPLETE

This message is sent by the RNC to the CN as a successful response to a MESSAGE QUERY message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
No of Broadcasts Completed List	M		9.2.10		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.14 MESSAGE STATUS QUERY FAILURE

This message is sent by the RNC to the CN in an unsuccessful response to a MESSAGE QUERY message.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Failure List	M		9.2.12		yes	ignore
Old Serial Number	M		9.2.4		yes	ignore
No of Broadcasts Completed List	O		9.2.10		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.15 RESET

The message is sent by the CN to the RNC to request that the RNC end broadcasting in one or more Service Areas.

Direction: CN → RNC

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Service Areas List	M		9.2.6		yes	ignore

9.1.16 RESET COMPLETE

This message is sent from the RNC to the CN as a successful response to a RESET message where indicated Service-Area(s) are now not broadcasting any messages.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Service Areas List	M		9.2.6		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.17 RESET FAILURE

This message is sent from the RNC to the CN as an unsuccessful response to a RESET message to indicate that a Service Area broadcasting related problem exists in one or more of its Service Areas.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Failure List	M		9.2.12		yes	ignore
Service Areas List	O		9.2.6		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.1.18 RESTART

This message is sent from the RNC to the CN to indicate a Service Area broadcasting related restart situation in one or more of its Service-Areas.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Service Areas List	M		9.2.6		yes	ignore
Recovery Indication	O		9.2.16		yes	ignore

9.1.19 FAILURE

This message is sent from the RNC to the CN to indicate that a Service Area broadcasting related problem exists in one or more of its Service-Areas.

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Service Areas List	M		9.2.6		yes	ignore

9.1.20 ERROR INDICATION

This message is sent by the RNC to the CN in response to any message which is not understood (e.g. invalid parameter or parameter value).

Direction: RNC → CN

PARAMETER	PRESENCE	RANGE	IE Type and Reference	Semantics Description	Criticality	Assigned Criticality
Message Type	M		9.2.1		yes	ignore
Message Identifier	M		9.2.19		yes	ignore
Serial Number	O		9.2.3		yes	ignore
Cause	O		9.2.14		yes	ignore
Criticality Diagnostics	O		9.2.17		yes	ignore

9.2 Information Element Definitions

9.2.1 MessageType

Message Type IE uniquely identifies the message being sent. It is mandatory for all messages.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Message Type				
>Procedure Code	M		ENUMERATED (Write-Replace, Kill, Load Query, Message Status Query, Reset, Restart, Failure, Error Indication,)	
>Type of Message	M		ENUMERATED (Initiating Message, Successful Outcome, Unsuccessful Outcome, Outcome)	

9.2.2 Broadcast Message Content

Broadcast Message Content IE is sent from the CN to the RNC containing user information i.e. the message, and will be broadcast over the radio interface.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Broadcast Message Content	M		OCTET STRING (1246)	

9.2.3 Serial Number

Serial Number IE is a 16-bit integer which identifies a particular message from the source and type indicated by the Message Identifier and is altered every time the message with a given Message Identifier is changed.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Serial Number	O		INTEGER (16)	

9.2.4 Old Serial Number

Old Serial Number IE enables identification of an existing message to be identified. The format of this IE is defined in 9.2.3.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Old Serial Number	M		9.2.3	

9.2.5 New Serial Number

New Serial Number IE enables identification of a new message for broadcast to be identified, and is altered every time the message is changes. The format of this IE is defined in 9.2.3.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
New Serial Number	O		9.2.3	

9.2.6 Service Areas List

Service Areas List IE is sent from the CN to the RNC. It indicates the group of Service Area(s) that the message will be broadcast to. The *Service Areas List* IE must include at least one Service Area.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Service Areas List		1 to <maxno of SAI>		
>Service Area Identifier	M		9.2.11	

Range bound	Explanation
MaxnoofSAI	Maximum no. of SAI in Service Areas List. Value is 65535

9.2.7 Category

Category IE is sent from the CN to the RNC, and is used to indicate the priority of the message.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Category	O		Enumerated (High Priority, Background, Normal, or Default)	This IE contains the broadcast priority of the message.

9.2.8 Repetition Period

Repetition Period IE is sent from the CN to the RNC and indicates the periodicity of message broadcasts.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Repetition Period	M		INTEGER 1...4096	Range is 1 to 4096 where each unit will represent a repetition of one second to a maximum of once per ~1 hour

9.2.9 No of Broadcasts Requested

No of Broadcasts Requested IE is sent from the CN to the RNC and indicates the number of times a message is to be broadcast.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
No of Broadcasts Requested	M	0 to 65535	INTEGER 0.... 65535	This specifies the number of times the message is to be broadcast.

9.2.10 No of Broadcasts Completed List

No of Broadcasts Completed List IE is sent from the RNC to the CN, and indicates the number of times that a CN message (all pages) has been sent to each Service Area specified in the *Service Areas List* IE for broadcast over the radio interface.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
No of-Broadcasts Completed List		1 to <maxnoof SAI>		
>Service Area Identifier	M		OCTET STRING (7)	
>No of Broadcasts Completed	M		INTEGER (0.. 65535)	
>No of Broadcasts Compl Info	O		ENUMERATED (overflow, unknown)	

Range bound	Explanation
MaxnoofSAI	Maximum no. of SAI in Service Areas List Value is 65535

9.2.11 Service Area Identifier

Service Area Identifier IE is used to identify an area consisting of one or more cells belonging to the same Location Area. Such an area is called a Service Area. For this protocol, only a Service Area that is defined to be applicable to the BC domain shall be used.

IE/Group Name	Presence	Range	IE type and Reference	Semantics description
SAI				
>PLMN ID	M		OCTET STRING (SIZE (3))	- digits 0 to 9, two digits per octet, - each digit encoded 0000 to 1001, - 1111 used as filler - bit 4 to 1 of octet n encoding digit 2n-1 - bit 8 to 5 of octet n encoding digit 2n -The PLMN-ID consists of 3 digits from MCC followed by either -a filler plus 2 digits from MNC (in case of 2 digit MNC) or -3 digits from MNC (in case of a 3 digit MNC).
>LAC	M		OCTET STRING (2)	0000 and FFFE not allowed.
>SAC	M		OCTET STRING (2)	

9.2.12 Failure List

Failure List IE identifies the list of Service-Area(s) for which the RNC could not complete as requested.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Failure List		1 to <maxnoof SAI>		
>Service Area Identifier	M		9.2.11	
>Cause	M		9.2.14	

Range bound	Explanation
MaxnoofSAI	Maximum no. of SAI in Service-Area-List. Value is 65535

9.2.13 Radio Resource Loading List

Radio Resource Loading List IE presents the available bandwidth available for Broadcast purposes of a specific Service Area.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Radio Resource Loading List		1 to <maxnoof SAI>		
>Service Area Identifier	M		9.2.11	
>Available Bandwidth	M		9.2.18	

Range bound	Explanation
MaxnoofSAI	Maximum no. of SAI in Service Area List. Value is 65535

9.2.14 Cause

Cause IE indicates the reason for a particular error event for the SABP protocol.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and Reference	Semantics Description
>Cause	M		INTEGER (Parameter-not-recognised(0), Parameter-value invalid(1), Valid-CN-message-not-identified(2), Service-Area-identity-not-valid(3), Unrecognised-message(4), Missing-mandatory-element(5), RNC-capacity-exceeded(6),	Range is 0-255 Sent when the recipient (CN or RNC) was unable to act upon the message received due to an unrecognised parameter. A message should not be rejected only because a parameter is not recognised as this would prevent extensions to the service Sent when a failure occurred due to the value of a parameter being invalid, e.g. out of range, or in Write-Replace, the parameter "no of pages" does not equal the number of pages received Sent when the RNC does not recognise the CN message reference Sent when the RNC does not recognise a Service-Area Identity Sent when the RNC did not recognise the message at all Sent when a mandatory element is missing from the message Sent when a write-replace fails because the RNC cannot meet the requested repetition period because of the cell loading

IE/GROUP NAME	PRESENCE	RANGE	IE Type and Reference	Semantics Description
			RNC-memory-exceeded(7),	Sent when the RNC is unable to store a CBS message as the RNC memory has been exceeded.
			Service-Area-broadcast-not-supported(8),	Sent when the SABCH/CN related Radio Resource is not configured for a Service-Area
			Service-Area-broadcast-not-operational(9),	Sent when the SABCH/CN related radio resource is not available because of error conditions or due to maintenance activities
			Message-reference already-used(10),	Sent when the recipient was unable to act upon the Write-Replace message received due to a previous Write-Replace received with the same message reference.
			Unspecified-error(11),	Sent when none of the above cause values apply.
			(Transfer Syntax Error(12),	Sent to indicate transfer syntax error in any message
			Semantic Error (12),	Sent to indicate semantic error any message
			Message not compatible with receiver state (14),	Sent to indicate that received message is not compatible with the receiver state
			Abstract Syntax Error (Reject)	Sent to indicate rejection due to

IE/GROUP NAME	PRESENCE	RANGE	IE Type and Reference	Semantics Description
			(15),	Abstract Syntax Error
			Abstract Syntax Error (Ignore and Notify) (16),	Sent to indicate Abstract Syntax Error in some IE that has been ignored
			Abstract Syntax Error (Falsely Constructed Message) (17), ...)	Sent to indicate Abstract Syntax Error due to false message construction

9.2.15 Data Coding Scheme

Data Coding Scheme IE is sent from the RNC to the CN and identifies the alphabet or coding employed for the message characters and message handling at the UE (it is passed transparently from the CN to the UE).

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Data Coding Scheme	M		INTEGER (0..255)	

9.2.16 Recovery Indication

Recovery Indication IE is used to indicate whether the CN related data was lost or is still available.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Recovery Indication	O		ENUMERATED (Available, Lost)	

9.2.17 Criticality Diagnostics

IE/Group Name	Presence	Range	IE type and reference	Semantics description
Criticality Diagnostics				
>Procedure Code	O		INTEGER (0..255)	Procedure Code is to be used if Criticality Diagnostics is part of Error Indication procedure, and not within the response message of the same procedure that caused the error
>Triggering Message	O		ENUMERATED(initiating message, successful outcome, unsuccessful outcome, outcome)	The Triggering Message is used only if the Criticality Diagnostics is part of Error Indication procedure.
>Procedure Criticality	O		ENUMERATED(reject, ignore, notify)	This Procedure Criticality is used for reporting the Criticality of the Triggering message (Procedure). The value 'ignore' shall never be used.
Information Element Criticality Diagnostics		0 to <maxnoof errors>		
>IE Criticality	M		ENUMERATED(reject, ignore, notify)	The IE Criticality is used for reporting the criticality of the triggering IE. The value 'ignore' shall not be used.
>IE ID	M		INTEGER (0..65535)	The IE Id of the not understood or missing IE
>Repetition Number	O		INTEGER (1..256)	The repetition number of the not understood IE if applicable

Range bound	Explanation
Maxnooferrors	Maximum no. of IE errors allowed to be reported with a single message. The value for maxnooferrors is 256.

9.2.18 Available Bandwidth

Available Bandwidth IE is used to indicate the Bandwidth available for the broadcast of messages.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and	Semantics Description
Available Bandwidth	O		INTEGER (0...20480)	The unit is: bit/second

9.2.19 Message Identifier

Message Identifier IE is set by the CN, transfer to the UE by the RNC.

IE/GROUP NAME	PRESENCE	RANGE	IE Type and Reference	Semantics Description
Message Identifier	M		OCTET STRING (SIZE(2))	This IE is set by the CN, transfer to the UE by the RNC, the RNC needs not to understand what is the meaning of the value but shall treat it as a identifier of a message

9.3 Message and Information Element Abstract Syntax (with ASN.1)

9.3.0 General

SABP ASN.1 definition conforms with [7] and [8].

The ASN.1 definition specifies the structure and content of SABP messages. SABP messages can contain any IEs specified in the object set definitions for that message without the order or number of occurrence being restricted by ASN.1. However, for this version of the standard, a sending entity shall construct a SABP message according to the PDU definitions module and with the following additional rules (Note that in the following IE means an IE in the object set with an explicit id. If one IE needed to appear more than once in one object set, then the different occurrences have different IE ids):

- IEs shall be ordered (in an IE container) in the order they appear in object set definitions.
- Object set definitions specify how many times IEs may appear. An IE shall appear exactly once if the presence field in an object has value "mandatory". An IE may appear at most once if the presence field in an object has value "optional" or "conditional". If in a tabular format there is multiplicity specified for an IE (i.e. an IE list) then in the corresponding ASN.1 definition the list definition is separated into two parts. The first part defines an IE container list where the list elements reside. The second part defines list elements. The IE container list appears as an IE of its own. For this version of the standard an IE container list may contain only one kind of list elements.

If a SABP message that is not constructed as defined above is received, this shall be considered as Abstract Syntax Error, and the message shall be handled as defined for Abstract Syntax error in Chapter 103.6.

9.3.1 Usage of protocol extension mechanism for non-standard use

The protocol extension mechanism for non-standard use may be used:

- for special operator- (and/or vendor) specific features considered not to be part of the basic functionality, i.e. the functionality required for a complete and high-quality specification in order to guarantee multivendor interoperability.
- by vendors for research purposes, e.g. to implement and evaluate new algorithms/features before such features are proposed for standardisation.

The extension mechanism shall not be used for basic functionality. Such functionality shall be standardised.

9.3.2 Elementary Procedure Definitions

```

-- *****
--
-- Elementary Procedure definitions
--
-- *****

SABP-PDU-Descriptions {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-PDU-Descriptions (0)}

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *****
--
-- IE parameter types from other modules.
--
-- *****

IMPORTS
    Criticality,
    ProcedureCode
FROM SABP-CommonDataTypes

    Error-Indication,
    Failure,
    Kill,
    Kill-Complete,
    Kill-Failure,
    Load-Query,
    Load-Query-Complete,
    Load-Query-Failure,
    Reset,
    Reset-Complete,
    Reset-Failure,
    Restart,
    Message-Status-Query,
    Message-Status-Query-Complete,
    Message-Status-Query-Failure,
    Write-Replace,
    Write-Replace-Complete,
    Write-Replace-Failure
FROM SABP-PDU-Contents

    id-Error-Indication,
    id-Failure-Indication,
    id-Kill,
    id-Reset,
    id-Restart-Indication,

```

```
id-Status-Load-Enquiry,
id-Status-Message-Query,
id-Write-Replace
FROM SABP-Constants;

-- *****
--
-- Interface Elementary Procedure Class
--
-- *****

SABP-ELEMENTARY-PROCEDURE ::= CLASS {
    &InitiatingMessage
    ,
    &SuccessfulOutcome          OPTIONAL,
    &UnsuccessfulOutcome        OPTIONAL,
    &procedureCode              ProcedureCode UNIQUE,
    &criticality                 Criticality    DEFAULT ignore
}
WITH SYNTAX {
    INITIATING MESSAGE      &InitiatingMessage
    [SUCCESSFUL OUTCOME

```

```

&SuccessfulOutcome]
  [UNSUCCESSFUL OUTCOME      &UnsuccessfulOutcome]
  PROCEDURE CODE             &procedureCode
  [CRITICALITY               &criticality]
}

-- *****
--
-- Interface PDU Definition
--
-- *****

SABP-PDU ::= CHOICE {
  initiatingMessage  InitiatingMessage,
  successfulOutcome  SuccessfulOutcome,
  unsuccessfulOutcome UnsuccessfulOutcome,
  ...
}

InitiatingMessage ::= SEQUENCE {
  procedureCode  SABP-ELEMENTARY-PROCEDURE.&procedureCode    ( {SABP-ELEMENTARY-PROCEDURES} ),
  criticality    SABP-ELEMENTARY-PROCEDURE.&criticality        ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode ),
  value         SABP-ELEMENTARY-PROCEDURE.&InitiatingMessage  ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode )
}

SuccessfulOutcome ::= SEQUENCE {
  procedureCode  SABP-ELEMENTARY-PROCEDURE.&procedureCode    ( {SABP-ELEMENTARY-PROCEDURES} ),
  criticality    SABP-ELEMENTARY-PROCEDURE.&criticality        ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode ),
  value         SABP-ELEMENTARY-PROCEDURE.&SuccessfulOutcome  ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode )
}

UnsuccessfulOutcome ::= SEQUENCE {
  procedureCode  SABP-ELEMENTARY-PROCEDURE.&procedureCode    ( {SABP-ELEMENTARY-PROCEDURES} ),
  criticality    SABP-ELEMENTARY-PROCEDURE.&criticality        ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode ),
  value         SABP-ELEMENTARY-PROCEDURE.&UnsuccessfulOutcome ( {SABP-ELEMENTARY-PROCEDURES}@procedureCode )
}

-- *****
--
-- Interface Elementary Procedure List
--
-- *****

SABP-ELEMENTARY-PROCEDURES SABP-ELEMENTARY-PROCEDURE ::= {
  SABP-ELEMENTARY-PROCEDURES-CLASS-1 |
  SABP-ELEMENTARY-PROCEDURES-CLASS-2 ,
  ...
}

SABP-ELEMENTARY-PROCEDURES-CLASS-1 SABP-ELEMENTARY-PROCEDURE ::= {
  write-Replace |
  kill |
  status-Load-Enquiry |
  status-Message-Query |

```

```

    reset
    ...
}

SABP-ELEMENTARY-PROCEDURES-CLASS-2 SABP-ELEMENTARY-PROCEDURE ::= {
    restart-Indication |
    failure-Indication |
    error-Indication ,
    ...
}

write-Replace SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Write-Replace
    SUCCESSFUL OUTCOME Write-Replace-Complete
    UNSUCCESSFUL OUTCOME Write-Replace-Failure
    PROCEDURE CODE id-Write-Replace
}

kill SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Kill
    SUCCESSFUL OUTCOME Kill-Complete
    UNSUCCESSFUL OUTCOME Kill-Failure
    PROCEDURE CODE id-Kill
}

status-Load-Enquiry SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Load-Query
    SUCCESSFUL OUTCOME Load-Query-Complete
    UNSUCCESSFUL OUTCOME Load-Query-Failure
    PROCEDURE CODE id-Status-Load-Enquiry
}

status-Message-Query SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Message-Status-Query
    SUCCESSFUL OUTCOME Message-Status-Query-Complete
    UNSUCCESSFUL OUTCOME Message-Status-Query-Failure
    PROCEDURE CODE id-Status-Message-Query
}

reset SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Reset
    SUCCESSFUL OUTCOME Reset-Complete
    UNSUCCESSFUL OUTCOME Reset-Failure
    PROCEDURE CODE id-Reset
}

restart-Indication SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Restart
    PROCEDURE CODE id-Restart-Indication
}

failure-Indication SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE Failure
    PROCEDURE CODE id-Failure-Indication
}

```

```

error-Indication SABP-ELEMENTARY-PROCEDURE ::= {
    INITIATING MESSAGE  Error-Indication
    PROCEDURE CODE      id-Error-Indication
}

```

END

9.3.3 PDU Definitions

```

-- *****
--
-- PDU definitions for SABP.
--
-- *****

SABP-PDU-Contents {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-PDU-Contents (1) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *****
--
-- IE parameter types from other modules.
--
-- *****

IMPORTS
    Broadcast-Message-Content,
    Category,
    Cause,
    Criticality-Diagnostics,
    Data-Coding-Scheme,
    Failure-List,
    Message-Identifier,
    New-Serial-Number,
    No-of-Broadcasts-Completed-List,
    No-of-Broadcasts-Requested,
    Old-Serial-Number,
    Radio-Resource-Loading-List,
    Recovery-Indication,
    Repetition-Period,
    Serial-Number ,
    Service-Areas-List
FROM SABP-IEs

    ProtocolExtensionContainer{},
    ProtocolIE-Container{},
    SABP-PROTOCOL-EXTENSION,
    SABP-PROTOCOL-IES
FROM SABP-Containers

```

```

id-Broadcast-Message-Content,
id-Category,
id-Criticality-Diagnostics,
id-Cause,
id-Data-Coding-Scheme,
id-Failure-List,
id-Message-Identifier,
id-New-Serial-Number,
id-No-of-Broadcasts-Completed-List,
id-No-of-Broadcasts-Requested,
id-Old-Serial-Number,
id-Radio-Resource-Loading-List,
id-Recovery-Indication,
id-Repetition-Period,
id-Serial-Number,
id-Service-Areas-List
FROM SABP-Constants;

-- *****
--
-- Write-Replace
--
-- *****

Write-Replace ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      { {Write-Replace-IEs} },
    protocolExtensions   ProtocolExtensionContainer { {Write-Replace-Extensions} } OPTIONAL,
    ...
}

Write-Replace-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier      PRESENCE mandatory } |
    { ID id-New-Serial-Number   CRITICALITY ignore  TYPE New-Serial-Number        PRESENCE mandatory } |
    { ID id-Old-Serial-Number   CRITICALITY ignore  TYPE Old-Serial-Number       PRESENCE optional  } |
    { ID id-Service-Areas-List  CRITICALITY ignore  TYPE Service-Areas-List      PRESENCE mandatory } |
    { ID id-Category            CRITICALITY ignore  TYPE Category                 PRESENCE optional  } |
    { ID id-Repetition-Period   CRITICALITY ignore  TYPE Repetition-Period       PRESENCE optional  } |
    { ID id-No-of-Broadcasts-Requested
      CRITICALITY ignore  TYPE No-of-Broadcasts-Requested PRESENCE mandatory } |
    { ID id-Data-Coding-Scheme  CRITICALITY ignore  TYPE Data-Coding-Scheme      PRESENCE mandatory } |
    { ID id-Broadcast-Message-Content
      CRITICALITY ignore  TYPE Broadcast-Message-Content PRESENCE mandatory } ,
    ...
}

Write-Replace-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Write-Replace-Complete
--
-- *****

```



```

Write-Replace-Complete ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container          { {Write-Replace-Complete-IEs} },
    protocolExtensions   ProtocolExtensionContainer { {Write-Replace-Complete-Extensions} } OPTIONAL,
    ...
}

Write-Replace-Complete-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier    PRESENCE mandatory } |
    { ID id-New-Serial-Number   CRITICALITY ignore  TYPE New-Serial-Number      PRESENCE mandatory } |
    { ID id-No-of-Broadcasts-Completed-List
      CRITICALITY ignore  TYPE No-of-Broadcasts-Completed-List
      PRESENCE mandatory } |
    { ID id-Criticality-Diagnostics
      CRITICALITY ignore  TYPE Criticality-Diagnostics    PRESENCE optional },
    ...
}

Write-Replace-Complete-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Write-Replace-Failure
--
-- *****

Write-Replace-Failure ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container          { {Write-Replace-Failure-IEs} },
    protocolExtensions   ProtocolExtensionContainer { {Write-Replace-Failure-Extensions} } OPTIONAL,
    ...
}

Write-Replace-Failure-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier    PRESENCE mandatory } |
    { ID id-New-Serial-Number   CRITICALITY ignore  TYPE New-Serial-Number      PRESENCE mandatory } |
    { ID id-Failure-List        CRITICALITY ignore  TYPE Failure-List          PRESENCE mandatory } |
    { ID id-No-of-Broadcasts-Completed-List
      CRITICALITY ignore  TYPE No-of-Broadcasts-Completed-List
      PRESENCE optional } |
    { ID id-Criticality-Diagnostics
      CRITICALITY ignore  TYPE Criticality-Diagnostics    PRESENCE optional },
    ...
}

Write-Replace-Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Kill
--
-- *****

```

```

Kill ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container    {{Kill-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Kill-Extensions}}
    ...
}

Kill-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier  PRESENCE mandatory } |
    { ID id-Old-Serial-Number   CRITICALITY ignore  TYPE Old-Serial-Number   PRESENCE mandatory } |
    { ID id-Service-Areas-List  CRITICALITY ignore  TYPE Service-Areas-List  PRESENCE mandatory } ,
    ...
}

Kill-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Kill-Complete
--
-- *****

Kill-Complete ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container    {{Kill-Complete-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Kill-Complete-Extensions}}
    ...
}

Kill-Complete-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier  PRESENCE mandatory } |
    { ID id-Old-Serial-Number   CRITICALITY ignore  TYPE Old-Serial-Number   PRESENCE mandatory } |
    { ID id-No-of-Broadcasts-Completed-List
      CRITICALITY ignore      TYPE No-of-Broadcasts-Completed-List
      PRESENCE mandatory } |
    { ID id-Criticality-Diagnostics
      CRITICALITY ignore      TYPE Criticality-Diagnostics
      PRESENCE optional } ,
    ...
}

Kill-Complete-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Kill-Failure
--
-- *****

Kill-Failure ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container    {{Kill-Failure-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Kill-Failure-Extensions}}
    ...
}

```

```

Kill-Failure-IEs SABP-PROTOCOL-IES ::= {
  { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier    PRESENCE mandatory } |
  { ID id-Old-Serial-Number    CRITICALITY ignore  TYPE Old-Serial-Number      PRESENCE mandatory } |
  { ID id-Failure-List          CRITICALITY ignore  TYPE Failure-List                  PRESENCE mandatory } |
  { ID id-No-of-Broadcasts-Completed-List
                                CRITICALITY ignore  TYPE No-of-Broadcasts-Completed-List
                                                                PRESENCE optional } |
  { ID id-Criticality-Diagnostics
    CRITICALITY ignore  TYPE Criticality-Diagnostics      PRESENCE optional },
  ...
}

Kill-Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
  ...
}

-- *****
--
-- Load-Query
--
-- *****

Load-Query ::= SEQUENCE {
  protocolIEs      ProtocolIE-Container      {{Load-Query-IEs}},
  protocolExtensions  ProtocolExtensionContainer {{Load-Query-Extensions}}  OPTIONAL,
  ...
}

Load-Query-IEs SABP-PROTOCOL-IES ::= {
  { ID id-Service-Areas-List  CRITICALITY ignore  TYPE Service-Areas-List    PRESENCE mandatory } ,
  ...
}

Load-Query-Extensions SABP-PROTOCOL-EXTENSION ::= {
  ...
}

-- *****
--
-- Load-Query-Complete
--
-- *****

Load-Query-Complete ::= SEQUENCE {
  protocolIEs      ProtocolIE-Container      {{Load-Query-Complete-IEs}},
  protocolExtensions  ProtocolExtensionContainer {{Load-Query-Complete-Extensions}}  OPTIONAL,
  ...
}

Load-Query-Complete-IEs SABP-PROTOCOL-IES ::= {
  { ID id-Radio-Resource-Loading-List
                                CRITICALITY ignore  TYPE Radio-Resource-Loading-List
                                                                PRESENCE mandatory } |
  { ID id-Criticality-Diagnostics

```

```

CRITICALITY ignore TYPE Criticality-Diagnostics PRESENCE optional },
    ...
}

Load-Query-Complete-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Load-Query-Failure
--
-- *****

Load-Query-Failure ::= SEQUENCE {
    protocolIEs ProtocolIE-Container {{Load-Query-Failure-IEs}},
    protocolExtensions ProtocolExtensionContainer {{Load-Query-Failure-Extensions}} OPTIONAL,
    ...
}

Load-Query-Failure-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Service-Areas-List CRITICALITY ignore TYPE Service-Areas-List PRESENCE mandatory } |
    { ID id-Failure-List CRITICALITY ignore TYPE Failure-List PRESENCE mandatory } |
    { ID id-Radio-Resource-Loading-List
        CRITICALITY ignore TYPE Radio-Resource-Loading-List
        PRESENCE optional } |
    { ID id-Criticality-Diagnostics
        CRITICALITY ignore TYPE Criticality-Diagnostics PRESENCE optional },
    ...
}

Load-Query-Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Message-Status-Query
--
-- *****

Message-Status-Query ::= SEQUENCE {
    protocolIEs ProtocolIE-Container {{Message-Status-Query-IEs}},
    protocolExtensions ProtocolExtensionContainer {{Message-Status-Query-Extensions}} OPTIONAL,
    ...
}

Message-Status-Query-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier CRITICALITY ignore TYPE Message-Identifier PRESENCE mandatory } |
    { ID id-Old-Serial-Number CRITICALITY ignore TYPE Old-Serial-Number PRESENCE mandatory } |
    { ID id-Service-Areas-List CRITICALITY ignore TYPE Service-Areas-List PRESENCE mandatory } ,
    ...
}

```

```

Message-Status-Query-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Message-Status-Query-Complete
--
-- *****

Message-Status-Query-Complete ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      {{Message-Status-Query-Complete-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Message-Status-Query-Complete-Extensions}} OPTIONAL,
    ...
}

Message-Status-Query-Complete-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier    PRESENCE mandatory } |
    { ID id-Old-Serial-Number   CRITICALITY ignore  TYPE Old-Serial-Number      PRESENCE mandatory } |
    { ID id-No-of-Broadcasts-Completed-List
      CRITICALITY ignore  TYPE No-of-Broadcasts-Completed-List
      PRESENCE mandatory } |
    { ID id-Criticality-Diagnostics
      CRITICALITY ignore  TYPE Criticality-Diagnostics      PRESENCE optional },
    ...
}

Message-Status-Query-Complete-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Message-Status-Query-Failure
--
-- *****

Message-Status-Query-Failure ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      {{Message-Status-Query-Failure-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Message-Status-Query-Failure-Extensions}} OPTIONAL,
    ...
}

Message-Status-Query-Failure-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Message-Identifier  CRITICALITY ignore  TYPE Message-Identifier    PRESENCE mandatory } |
    { ID id-Failure-List        CRITICALITY ignore  TYPE Failure-List          PRESENCE mandatory } |
    { ID id-Old-Serial-Number   CRITICALITY ignore  TYPE Old-Serial-Number      PRESENCE mandatory } |

    { ID id-No-of-Broadcasts-Completed-List
      CRITICALITY ignore  TYPE No-of-Broadcasts-Completed-List
      PRESENCE optional } |
    { ID id-Criticality-Diagnostics
      CRITICALITY ignore  TYPE Criticality-Diagnostics      PRESENCE optional },
    ...
}

```

```

Message-Status-Query-Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Reset
--
-- *****

Reset ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container          {{Reset-IEs}},
    protocolExtensions   ProtocolExtensionContainer  {{Reset-Extensions}}  OPTIONAL,
    ...
}

Reset-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Service-Areas-List  CRITICALITY ignore  TYPE Service-Areas-List          PRESENCE mandatory } ,
    ...
}

Reset-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Reset-Complete
--
-- *****

Reset-Complete ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container          {{Reset-Complete-IEs}},
    protocolExtensions   ProtocolExtensionContainer  {{Reset-Complete-Extensions}}  OPTIONAL,
    ...
}

Reset-Complete-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Service-Areas-List  CRITICALITY ignore  TYPE Service-Areas-List          PRESENCE mandatory } |
    { ID id-Criticality-Diagnostics  CRITICALITY ignore  TYPE Criticality-Diagnostics  PRESENCE optional } ,
    ...
}

Reset-Complete-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Reset-Failure
--
-- *****

```

```

Reset-Failure ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      {{Reset-Failure-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Reset-Failure-Extensions}} OPTIONAL,
    ...
}

Reset-Failure-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Failure-List          CRITICALITY ignore TYPE Failure-List          PRESENCE mandatory } |
    { ID id-Service-Areas-List    CRITICALITY ignore TYPE Service-Areas-List    PRESENCE optional } |
    { ID id-Criticality-Diagnostics CRITICALITY ignore TYPE Criticality-Diagnostics PRESENCE optional } ,
    ...
}

Reset-Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Restart
--
-- *****

Restart ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      {{Restart-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Restart-Extensions}} OPTIONAL,
    ...
}

Restart-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Service-Areas-List    CRITICALITY ignore TYPE Service-Areas-List    PRESENCE mandatory } |
    { ID id-Recovery-Indication   CRITICALITY ignore TYPE Recovery-Indication   PRESENCE optional } ,
    ...
}

Restart-Extensions SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- *****
--
-- Failure
--
-- *****

Failure ::= SEQUENCE {
    protocolIEs          ProtocolIE-Container      {{Failure-IEs}},
    protocolExtensions   ProtocolExtensionContainer {{Failure-Extensions}} OPTIONAL,
    ...
}

Failure-IEs SABP-PROTOCOL-IES ::= {
    { ID id-Service-Areas-List    CRITICALITY ignore TYPE Service-Areas-List    PRESENCE mandatory } ,
    ...
}

```

```

}

Failure-Extensions SABP-PROTOCOL-EXTENSION ::= {
  ...
}

-- *****
--
-- Error-Indication
--
-- *****

Error-Indication ::= SEQUENCE {
  protocolIEs          ProtocolIE-Container    {{Error-Indication-IEs}},
  protocolExtensions  ProtocolExtensionContainer {{Error-Indication-Extensions}} OPTIONAL,
  ...
}

Error-Indication-IEs SABP-PROTOCOL-IES ::= {
  { ID id-Message-Identifier CRITICALITY ignore TYPE Message-Identifier PRESENCE mandatory } |
  { ID id-Serial-Number      CRITICALITY ignore TYPE Serial-Number      PRESENCE optional } |
  { ID id-Cause              CRITICALITY ignore TYPE Cause              PRESENCE optional } |
  { ID id-Criticality-Diagnostics
  CRITICALITY ignore TYPE Criticality-Diagnostics PRESENCE optional },
  ...
}

Error-Indication-Extensions SABP-PROTOCOL-EXTENSION ::= {
  ...
}

END

```

9.3.4 Information Element Definitions

```

-- *****
--
-- Information Element Definitions
--
-- *****

SABP-IEs {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-IEs (2) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS
  maxRadio-Resource-Loading-List,

```



```
maxFailure-List,
maxNo-of-Broadcasts-Completed-List,
maxNrOfErrors,
maxService-Areas-List
FROM SABP-Constants

Criticality,
ProcedureCode,
TriggeringMessage,
ProtocolIE-ID
FROM SABP-CommonDataTypes

ProtocolExtensionContainer{}

SABP-PROTOCOL-EXTENSION
FROM SABP-Containers;

-- A

Available-Bandwidth ::= INTEGER (0..20480)
-- bits/sec

-- B

Broadcast-Message-Content ::= OCTET STRING (SIZE (1246))
-- This IE is sent from the CN to the RNC containing user information i.e.
-- the message.

-- C

Category ::= ENUMERATED {
    high-priority,
    background-priority,
    normal-priority,
    default-priority,
    ...
}

Cause ::= INTEGER {
    parameter-not-recognised (0),
    parameter-value-invalid (1),
    valid-CN-message-not-identified (2),
    service-area-identity-not-valid (3),
    unrecognised-message (4),
    missing-mandatory-element (5),
    rNC-capacity-exceeded (6),
    rNC-memory-exceeded (7),
    service-area-broadcast-not-supported (8),
    service-area-broadcast-not-operational (9),
    message-reference-already-used (10),
    unspecified-error (11),
    transfer-syntax-error (12),
    semantic-error (13),
    message-not-compatible-with-receiver-state (14),
```

```

    abstract-syntax-error-reject                (15),
    abstract-syntax-error-ignore-and-notify     (16),
    abstract-syntax-error-falsely-constructed-message (17)
} (0..255)

Criticality-Diagnostics ::= SEQUENCE {
    procedureCode          ProcedureCode          OPTIONAL,
    triggeringMessage      TriggeringMessage      OPTIONAL,
    procedureCriticality   Criticality            OPTIONAL,
    iEsCriticalityDiagnostics CriticalityDiagnostics-IE-List OPTIONAL,
    iE-Extensions          ProtocolExtensionContainer { {CriticalityDiagnostics-ExtIEs} } OPTIONAL,
    ...
}

CriticalityDiagnostics-ExtIEs SABP-PROTOCOL-EXTENSION ::= {
    ...
}

CriticalityDiagnostics-IE-List ::= SEQUENCE (SIZE (1..maxNrOfErrors)) OF
SEQUENCE {
    iECriticality          Criticality,
    iE-ID                  ProtocolIE-ID,
    repetitionNumber       RepetitionNumber      OPTIONAL,
    iE-Extensions          ProtocolExtensionContainer { {CriticalityDiagnostics-IE-List-ExtIEs} } OPTIONAL,
    ...
}

CriticalityDiagnostics-IE-List-ExtIEs SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- D

Data-Coding-Scheme ::= INTEGER (0..255)

-- E

-- F

Failure-List ::= SEQUENCE (SIZE (1..maxFailure-List)) OF Failure-List-Item

Failure-List-Item ::= SEQUENCE {
    service-area-identifier Service-Area-Identifier,
    cause                    Cause,
    iE-Extensions            ProtocolExtensionContainer { {FailureListItemIE-ExtIEs} } OPTIONAL,
    ...
}

FailureListItemIE-ExtIEs SABP-PROTOCOL-EXTENSION ::= {
    ...
}

-- G

```

```

-- H
-- I
-- J
-- K
-- L
-- M

Message-Identifier ::= OCTET STRING (SIZE (2))

-- N

New-Serial-Number          ::= Serial-Number

No-of-Broadcasts-Completed-List ::= SEQUENCE (SIZE (1..maxNo-of-Broadcasts-Completed-List)) OF
    No-of-Broadcasts-Completed-List-Item

No-of-Broadcasts-Completed-List-Item ::= SEQUENCE {
    service-area-identifier      Service-Area-Identifier,
    no-of-broadcasts-compl       INTEGER (0..65535),
    no-of-broadcasts-compl-info  No-Of-Broadcasts-Compl-Info          OPTIONAL,
    iE-Extensions                ProtocolExtensionContainer { {NoOfBroadcastsCompletedListItemIE-ExtIEs} } OPTIONAL,
    ...
}

NoOfBroadcastsCompletedListItemIE-ExtIEs SABP-PROTOCOL-EXTENSION ::= {
    ...
}

No-Of-Broadcasts-Compl-Info      ::= ENUMERATED {
    overflow,
    unknown,
    ...
}

No-of-Broadcasts-Requested      ::= INTEGER {
    broadcast-indefinitely (0)
} (0..65535)

-- O

Old-Serial-Number              ::= Serial-Number

-- P
-- Q
-- R

Radio-Resource-Loading-List ::= SEQUENCE (SIZE (1..maxRadio-Resource-Loading-List)) OF
    Radio-Resource-Loading-List-Item

```

```

Radio-Resource-Loading-List-Item ::= SEQUENCE {
    service-area-identifier      Service-Area-Identifier,
    available-bandwidth         Available-Bandwidth,
    iE-Extensions                ProtocolExtensionContainer { {RadioResourceLoadingListItemIE-ExtIEs} } OPTIONAL,
    ...
}

RadioResourceLoadingListItemIE-ExtIEs SABP-PROTOCOL-EXTENSION ::= {
    ...
}

Recovery-Indication ::= ENUMERATED {
    data-lost,
    data-available
}

RepetitionNumber          ::= INTEGER(1..256)

Repetition-Period         ::= INTEGER (1..4096)
-- Each unit represents a repetition of one second to a maximum of
-- once per 4096 seconds (~1 hour).

-- S

Serial-Number             ::= INTEGER (0..65535)

Service-Area-Identifier ::= SEQUENCE {
    plmn-id                OCTET STRING (SIZE (3))
                            -- Digits 0 to 9, two digits per octet.      --
                            -- Each octet encoded 0000 to 1001.          --
                            -- 1111 used as filler                       --
                            -- Bit 4 to 1 of octet n encoding digit 2n-1. --
                            -- Bit 8 to 5 of octet n encoding digit 2n.  --
                            -- The PLMN-ID consists of 3 digits from MCC  --
                            -- followed by either a filler plus 2 digits  --
                            -- from MNC (in case of 2 digit MNC) or 3 digits --
                            -- from MNC (in case of 3 digit MNC).        -- ,
    lac                    OCTET STRING (SIZE (2))
                            -- 0000 and FFFE not allowed                -- ,
    sac                    OCTET STRING (SIZE (2))
}

-- **TODO** The IE type for these parameters is not known as yet
Service-Areas-List ::= SEQUENCE (SIZE (1..maxService-Areas-List)) OF Service-Area-Identifier

-- T

-- U

-- V

```

```
-- W
-- X
-- Y
END
```

9.3.5 Common Definitions

```
-- *****
--
-- Common definitions
--
-- *****

SABP-CommonDataTypes {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-CommonDataTypes (3) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

Criticality      ::= ENUMERATED { reject, ignore, notify }

Presence        ::= ENUMERATED { optional, conditional, mandatory }

ProcedureCode    ::= INTEGER (0..255)

ProtocolExtensionID ::= INTEGER (0..65535)

ProtocolIE-ID    ::= INTEGER (0..65535)

TriggeringMessage ::= ENUMERATED { initiating-message, successful-outcome, unsuccessful-outcome, outcome}

END
```

9.3.6 Constant Definitions

```
-- *****
--
-- Constant definitions
--
-- *****

SABP-Constants {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-Constants (4) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN
```

```
-- *****
--
-- Elementary Procedures
--
-- *****

id-Write-Replace          INTEGER ::= 0
id-Kill                   INTEGER ::= 1
id-Status-Load-Enquiry   INTEGER ::= 2
id-Status-Message-Query  INTEGER ::= 3
id-Restart-Indication     INTEGER ::= 4
id-Reset                  INTEGER ::= 5
id-Failure-Indication     INTEGER ::= 6
id-Error-Indication      INTEGER ::= 7

-- *****
--
-- IEs
--
-- *****

id-Broadcast-Message-Content  INTEGER ::= 0
id-Category                   INTEGER ::= 1
id-Cause                       INTEGER ::= 2
id-Criticality-Diagnostics    INTEGER ::= 3
id-Data-Coding-Scheme         INTEGER ::= 4
id-Failure-List               INTEGER ::= 5
id-Message-Identifier         INTEGER ::= 6
id-New-Serial-Number          INTEGER ::= 7
id-No-of-Broadcasts-Completed-List  INTEGER ::= 8
id-No-of-Broadcasts-Requested  INTEGER ::= 9
id-Old-Serial-Number          INTEGER ::= 10
id-Radio-Resource-Loading-List  INTEGER ::= 11
id-Recovery-Indication        INTEGER ::= 12
id-Repetition-Period          INTEGER ::= 13
id-Serial-Number              INTEGER ::= 14
id-Service-Areas-List        INTEGER ::= 15

-- *****
--
-- Extension constants
--
-- *****

-- *****
--
-- Lists
--
-- *****

maxRadio-Resource-Loading-List  INTEGER ::= 65535
maxFailure-List                 INTEGER ::= 65535
maxNo-of-Broadcasts-Completed-List  INTEGER ::= 65535
maxNrOfErrors                   INTEGER ::= 256
```

```

maxService-Areas-List          INTEGER ::= 65535

maxProtocolExtensions          INTEGER ::= 65535
maxProtocolIEs                 INTEGER ::= 65535

```

END

9.3.7 Container Definitions

```

-- *****
--
-- Container definitions
--
-- *****

SABP-Containers {
itu-t (0) identified-organization (4) etsi (0) mobileDomain (0)
umts-Access (20) modules (3) sabp (3) version1 (1) sabp-Containers (5) }

DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- *****
--
-- IE parameter types from other modules.
--
-- *****

IMPORTS
    Criticality,
    Presence,
    ProtocolExtensionID,
    ProtocolIE-ID
FROM SABP-CommonDataTypes

    maxProtocolExtensions,
    maxProtocolIEs
FROM SABP-Constants;

-- *****
--
-- Class Definition for Protocol IEs
--
-- *****

SABP-PROTOCOL-IES ::= CLASS {
    &id          ProtocolIE-ID          UNIQUE,
    &criticality Criticality          DEFAULT ignore,
    &Value,
    &presence    Presence
}
WITH SYNTAX {
    ID          &id

```

```

    CRITICALITY      &criticality
    TYPE              &Value
    PRESENCE          &presence
}

-- *****
--
-- Class Definition for Protocol Extensions
--
-- *****

SABP-PROTOCOL-EXTENSION ::= CLASS {
    &id                ProtocolExtensionID    UNIQUE,
    &criticality        Criticality            DEFAULT ignore,
    &Extension,
    &presence           Presence
}
WITH SYNTAX {
    ID                 &id
    CRITICALITY        &criticality
    EXTENSION           &Extension
    PRESENCE            &presence
}

-- *****
--
-- Container for Protocol IEs
--
-- *****

ProtocolIE-Container {SABP-PROTOCOL-IES : IEsSetParam} ::=
    SEQUENCE (SIZE (0..maxProtocolIEs)) OF
    ProtocolIE-Field {{IEsSetParam}}

ProtocolIE-Field {SABP-PROTOCOL-IES : IEsSetParam} ::= SEQUENCE {
    id                 SABP-PROTOCOL-IES.&id          ({{IEsSetParam}}),
    criticality         SABP-PROTOCOL-IES.&criticality  ({{IEsSetParam}}{@id}),
    value               SABP-PROTOCOL-IES.&Value       ({{IEsSetParam}}{@id})
}

-- *****
--
-- Container Lists for Protocol IE Containers
--
-- *****

ProtocolIE-ContainerList {INTEGER : lowerBound, INTEGER : upperBound, SABP-PROTOCOL-IES : IEsSetParam} ::=
    SEQUENCE (SIZE (lowerBound..upperBound)) OF
    ProtocolIE-Container {{IEsSetParam}}

-- *****
--
-- Container for Protocol Extensions
--
-- *****

```



```
ProtocolExtensionContainer {SABP-PROTOCOL-EXTENSION : ExtensionSetParam} ::=
    SEQUENCE (SIZE (1..maxProtocolExtensions)) OF
    ProtocolExtensionField {{ExtensionSetParam}}

ProtocolExtensionField {SABP-PROTOCOL-EXTENSION : ExtensionSetParam} ::= SEQUENCE {
    id                SABP-PROTOCOL-EXTENSION.&id                ({ExtensionSetParam}),
    criticality       SABP-PROTOCOL-EXTENSION.&criticality       ({ExtensionSetParam}{@id}),
    extensionValue    SABP-PROTOCOL-EXTENSION.&Extension        ({ExtensionSetParam}{@id})
}
```

END

9.4 Message Transfer Syntax

SABP shall use the ASN.1 Basic Packed Encoding Rules (BASIC-PER) Aligned Variant as transfer syntax as specified in ref. [9].

10 Handling of Unknown, Unforeseen or Erroneous Protocol Data

10.1 General

Protocol Error cases can be divided into three classes:

- Transfer Syntax Error;
- Abstract Syntax Error;
- Logical Error.

10.2 Transfer Syntax Error

A Transfer Syntax Error occurs when the receiver is not able to decode the received physical message Transfer syntax errors are always detected in the process of ASN.1 decoding. If a Transfer Syntax Error occurs, the receiver should initiate Error Indication procedure with appropriate cause value for the Transfer Syntax protocol error.

10.3 Abstract Syntax Error

10.3.1 General

An Abstract Syntax Error occurs when the receiving functional SABP entity:

1. receives IEs or IE groups that cannot be understood (unknown IE id);
2. receives IEs for which the logical range is violated (e.g.: ASN.1 definition: 0 to 15, the logical range is 0 to 10 (values 11 to 15 are undefined), and 12 will be received; this case will be handled as an abstract syntax error using criticality information sent by the originator of the message);
3. does not receive IEs or IE groups but according to the specified presence of the concerning object, the IEs or IE groups should have been present in the received message.
4. receives IEs or IE groups that are defined to be part of that message in wrong order or with too many occurrences of the same IE or IE group

Cases 1 and 2 (not comprehended IE/IE group) are handled based on received Criticality information. Case 3 (missing IE/IE group) is handled based on Criticality information and Presence information for the missing IE/IE group specified in the version of the specification used by the receiver. Case 4 (IEs or IE groups in wrong order or with too many occurrences) results in rejecting the procedure.

If an Abstract Syntax Error occurs, the receiver shall read the remaining message and shall then for each detected Abstract Syntax Error act according to the Criticality Information and Presence Information for the IE/IE group due to which Abstract Syntax Error occurred in accordance with subclauses 10.3.4 and 10.3.5. The handling of case 4 is specified in subclause 10.3.6.

10.3.2 Criticality Information

In the SABP messages there is criticality information set for individual IEs and/or IE groups. This criticality information instructs the receiver how to act when receiving an IE or an IE group that is not comprehended i.e. the entire item (IE or IE group) which is not (fully or partially) comprehended shall be treated in accordance with its own criticality information as specified in subclause 10.3.4.

In addition, the criticality information is used in case of the missing IE/IE group abstract syntax error (see subclause 10.3.5).

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:

- Reject IE;
- Ignore IE and Notify Sender;
- Ignore IE.

The following rules restrict when a receiving entity may consider an IE, an IE group or an EP not comprehended (not implemented), and when action based on criticality information is applicable:

1. IE or IE group: When one new or modified IE or IE group is implemented for one EP from a standard version, then other new or modified IEs or IE groups specified for that EP in that standard version shall be considered comprehended by the receiving entity (some may still remain unsupported).

Note that this restriction is applicable to a sending entity for constructing messages.

2. EP: The comprehension of different EPs within a standard version or between different standard versions is not mandated. Any EP that is not supported may be considered not comprehended, even if another EP from that standard version is comprehended, and action based on criticality shall be applied.

10.3.3 Presence Information

For many IEs/IE groups which are optional according to the ASN.1 transfer syntax, RANAP specifies separately if the presence of these IEs/IE groups is optional or mandatory with respect to RNS application by means of the presence field of the concerning object of class RANAP-PROTOCOL-IES, RANAP-PROTOCOL-IES-PAIR, RANAP-PROTOCOL-EXTENSION or RANAP-PRIVATE-IES.

The presence field of the indicated classes supports three values:

1. Optional;
2. Conditional;
3. Mandatory.

If an IE/IE group is not included in a received message and the presence of the IE/IE group is mandatory or the presence is conditional and the condition is true according to the version of the specification used by the receiver, an abstract syntax error occurs due to a missing IE/IE group.

10.3.4 Not comprehended IE/IE group

10.3.4.1 Procedure Code

The receiving node shall treat the different types of received criticality information of the *Procedure Code* according to the following:

Reject IE:

- If a message is received with a *Procedure Code* marked with "*Reject IE*" which the receiving node does not comprehend, the receiving node shall reject the procedure using the Error Indication procedure.

Ignore IE and Notify Sender:

- If a message is received with a *Procedure Code* marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the procedure and initiate the Error Indication procedure.

Ignore IE:

- If a message is received with a *Procedure Code* marked with "*Ignore IE*" which the receiving node does not comprehend, the receiving node shall ignore the procedure.

When using the Error Indication procedure to reject a procedure or to report an ignored procedure it shall include the *Procedure Code IE*, the *Triggering Message IE*, and the *Procedure Criticality IE* in the *Criticality Diagnostics IE*.

10.3.4.2 IEs other than the Procedure Code

The receiving node shall treat the different types of received criticality information of an IE/IE group other than the *Procedure Code* according to the following:

Reject IE:

- If a message *initiating* a procedure is received containing one or more Ies/IE groups marked with "*Reject IE*" which the receiving node does not comprehend; none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the rejection of one or more Ies/IE groups using the message normally used to report unsuccessful outcome of the procedure.
- If a message *initiating* a procedure that does not have a message to report unsuccessful outcome is received containing one or more Ies/IE groups marked with "*Reject IE*" which the receiving node does not comprehend, the receiving node shall initiate the Error Indication procedure.
- If a *response* message is received containing one or more IEs marked with "*Reject IE*" which the receiving node does not comprehend, the receiving node shall initiate local error handling.

Ignore IE and Notify Sender:

- If a message *initiating* a procedure is received containing one or more Ies/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups, continue with the procedure as if the not comprehended IEs/IE groups were not received (except for the reporting) using the understood IEs/IE groups, and report in the response message of the procedure that one or more IEs/IE groups have been ignored.
- if a message *initiating* a procedure that does not have a message to report the outcome of the procedure is received containing one or more IEs/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups, continue with the procedure as if the not comprehended IEs/IE groups were not received (except for the reporting) using the understood IEs/IE groups, and initiate the Error Indication procedure to report that one or more IEs/IE groups have been ignored.
- If a *response* message is received containing one or more IEs/IE groups marked with "*Ignore IE and Notify Sender*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IE/IE groups and initiate the Error Indication procedure.

Ignore IE:

- If a message *initiating* a procedure is received containing one or more IEs/IE groups marked with "*Ignore IE*" which the receiving node does not comprehend, the receiving node shall ignore the content of the not comprehended IEs/IE groups and continue with the procedure as if the not comprehended IEs/IE groups were not received using only the understood IEs/IE groups.

When reporting not comprehended IEs/IE groups marked with "*Reject IE*" or "*Ignore IE and Notify Sender*" using a response message defined for the procedure, the *Information Element Criticality Diagnostics IE* shall be included in the *Criticality Diagnostics IE* for each reported IE/IE group. The *Repetition Number IE* shall be included in the *Information Element Criticality Diagnostics IE* if the reported IE/IE group was part of a "SEQUENCE OF" definition.

When reporting not comprehended IEs/IE groups marked with "*Reject IE*" or "*Ignore IE and Notify Sender*" using the Error Indication procedure, the *Procedure Code IE*, the *Triggering Message IE*, *Procedure Criticality IE*, and the *Information Element Criticality Diagnostics IE* shall be included in the *Criticality Diagnostics IE* for each reported IE/IE group. The *Repetition Number IE* shall be included in the *Information Element Criticality Diagnostics IE* if the reported IE/IE group was part of a "SEQUENCE OF" definition.

10.3.5 Missing IE or IE group

The receiving node shall treat the missing IE/IE group according to the criticality information for the missing IE/IE group in the received message specified in the version of this specification used by the receiver:

Reject IE:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Reject IE*"; none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the missing IEs/IE groups using the message normally used to report unsuccessful outcome of the procedure.
- if a received message *initiating* a procedure that does not have a message to report unsuccessful outcome is missing one or more IEs/IE groups with specified criticality "*Reject IE*", the receiving node shall initiate the Error Indication procedure.
- if a received *response* message is missing one or more IEs/IE groups with specified criticality "*Reject IE*", the receiving node shall initiate local error handling.

Ignore IE and Notify Sender:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall continue with the procedure based on the other IEs/IE groups present in the message and report in the response message of the procedure that one or more IEs/IE groups were missing.
- if a received message *initiating* a procedure that does not have a message to report the outcome of the procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall continue with the procedure based on the other IEs/IE groups present in the message and initiate the Error Indication procedure to report that one or more IEs/IE groups were missing.
- if a received *response* message is missing one or more IEs/IE groups with specified criticality "*Ignore IE and Notify Sender*", the receiving node shall initiate the Error Indication procedure.

Ignore IE:

- if a received message *initiating* a procedure is missing one or more IEs/IE groups with specified criticality "*Ignore IE*", the receiving node shall continue with the procedure based on the other IEs/IE groups present in the message.

When reporting missing IEs/IE groups with specified criticality "*Reject IE*" or "*Ignore IE and Notify Sender*" using a response message defined for the procedure, the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

When reporting missing IEs/IE groups with specified criticality "*Reject IE*" or "*Ignore IE and Notify Sender*" using the Error Indication procedure, the *Procedure Code* IE, the *Triggering Message* IE, *Procedure Criticality* IE, and the *Information Element Criticality Diagnostics* IE shall be included in the *Criticality Diagnostics* IE for each reported IE/IE group.

10.3.6 IEs or IE groups received in wrong order or with too many occurrences

If a message with IEs or IE groups in wrong order or with too many occurrences is received, the receiving node shall behave according to the following:

- If a message *initiating* a procedure is received containing IEs or IE groups in wrong order or with too many occurrences, none of the functional requests of the message shall be executed. The receiving node shall reject the procedure and report the cause value "Abstract Syntax Error (Falsely Constructed Message)" using the message normally used to report unsuccessful outcome of the procedure.
- If a message *initiating* a procedure that does not have a message to report unsuccessful outcome is received containing IEs or IE groups in wrong order or with too many occurrences, the receiving node shall initiate the Error Indication procedure, and use cause value "Abstract Syntax Error (Falsely Constructed Message)".
- If a *response* message is received containing IEs or IE groups in wrong order or with too many occurrences, the receiving node shall initiate local error handling.

10.4 Logical Error

Logical error situations occur when a message is comprehended correctly, but the information contained within the message is not valid (i.e. semantic error), or describes a procedure which is not compatible with the state of the receiver. In these conditions, the following behaviour shall be performed (unless otherwise specified) as defined by the class of the elementary procedure, irrespective of the criticality information of the IE's/IE groups containing the erroneous values.

Class 1:

Where the logical error occurs in a request message of a class 1 procedure, and the procedure has a failure message, the failure message shall be sent with an appropriate cause value. Typical cause values are:

- Semantic Error;
- Message not compatible with receiver state.

Where the logical error is contained in a request message of a class 1 procedure, and the procedure does not have a failure message, the Error Indication procedure shall be initiated with an appropriate cause value.

Where the logical error exists in a response message of a class 1 procedure, local error handling shall be initiated.

Class 2:

Where the logical error occurs in a message of a class 2 procedure, the Error Indication procedure shall be initiated with an appropriate cause value.

Annex A (informative): Change history

Change history					
TSG RAN#	Version	CR	Tdoc RAN	New Version	Subject/Comment
RAN_07	-	-	-	3.0.0	Approved at TSG RAN #7 and placed under Change Control
RAN_08	3.0.0	-	RP-000239	3.1.0	Approved at TSG RAN #8
RAN_09	3.1.0	011 012 014 015 016 017 018 019 020 021 022 023	RP-000377	3.2.0	Approved at TSG RAN #9
RAN_10	3.2.0	024 025 026 027 028 029	RP-000616	3.3.0	Approved at TSG RAN #10

History

Document history		
V3.0.0	March 2000	Publication
V3.1.0	June 2000	Publication
V3.2.0	September 2000	Publication
V3.3.0	December 2000	Publication