3GPP TSG SA WG3 Security — S3#16 **S3-000648**

28-30 November, 2000

Sophia Antipolis, France

TSG-RAN Working Group 2 (Radio L2 and Radio L3) ***R2-001286***
Oahu, HI, USA, 22 - 26 May 2000

Source: **TSG-RAN WG2**

To: **TSG-SA WG3**

Cc: **TSG-SA**

Subject: **LS on Security issues**

Contact: Ainkaran Krishnarajah, Ericsson
Email: Ainkaran.Krishnarajah@era.ericsson.se
Tel.: +46 8 757 2565

RAN WG2 would like to inform SA WG3 that RAN WG2 has been reviewing the specification of security in RAN WG2. The outcome has been to reorganise the relevant RAN WG2 specifications in order to have security clearly specified.

RAN WG2 has noticed that there are inconsistencies with RAN WG2 specifications on security and TS 33.102. RAN WG2 would like to work with SA WG3 to ensure that security is clearly specified in both working groups.

1. The review started with a list of questions and this is found in R2-001172, from which CRs were written to clarify some of these issues. This discussion paper is attached in this LS.
2. RAN WG2 has started to remove sections that are covered in TS 33.102. New chapters will be created in TS 25.331 and references to the Security Architecture specification will be made, as necessary. RAN WG2 would like SA WG3 to look at these deleted sections and determine whether information from these chapters is required in TS 33.102.

   For example R2-001185 has removed Chapter 8 Ciphering from TS 25.301. Definitions of the ciphering unit are now required in TS 33.102. Also the CFN is 8 bits whereas it is specified as 7 bits in TS 33.102.
3. The HFN has been clarified by R2-001275. Please indicate if our assumptions are correct.
4. The RRC messages that shall not be integrity protected are now specified in TS 25.331 (R2-001276). RAN WG2 would therefore like to ask SA WG3 to remove the similar list in TS 33.102.
5. A reset mechanism for HFN was proposed in R2-001277 to handle cases where the HFN may become out-of-sync between the UE and UTRAN.
6. Clarification on ciphering parameters and integrity protection procedure in case of SRNS relocation is addressed in R2-001278. Please indicate if our assumptions are correct.
7. RAN WG2 would like to ask SA WG3 whether they are planning to specify UTRAN-UTRAN handover in TS 33.102.

3GPP TSG RAN WG2 #13 **Tdoc RAN WG2 1172/00**
22 - 26th May 2000
Oahu, Hawaii, USA

Source : Nortel Networks

Object : List of questions and issues on security

1. Introduction

This contribution addresses a list of questions on the security principles and the associated description in the RAN WG2 specifications. It also identifies some potential corrections needed. The analysis is based on the March version of the 33.102 specification which specifies the security architecture.
Last, the document also addresses some organisation aspects on how to complete security aspects between RAN2 and SA3.

2. Integrity protection

Question/issue 1:
The messages on which integrity protection should be applied is currently indicated in 33.102, specified witha shall that does not give any dependancy on the security procedure. Also, the list of messages is specified as "all but …", and therefore covers messages that should not be intergrity protected e.g. TFC control in Transparent mode RLC (the message being on a few bits only…). Also, one can wonder whether the allocation of DSCH/USCH capacity in TDD should be integrity protected.
Proposed way forward:

- describe for every message in RRC whether IP applies
- propose SA3 the removal of the list from 33.102
- request SA3 to clarify requirement for some messages where need for IP is dubious or very costly

Question/issue 2:
It is not clear whether the RRC sequence number is part of the message on which XMAC-I is calculated
Proposed way forward:

- Precise in RRC

Question/issue 3:
There is a need to have the rules for RRC SN incrementation clearly specified
Proposed way forward:

- Create a new section on security counters in RRC, and precise when the SN is incremented

Question/issue 4:
Does the RRC SN work in an unambiguous way
Proposed way forward:

- Create a new section on security counters in RRC, and precise when the SN is incremented. Needs probably a study of the issue for all RRC procedures…

Question/issue 5:
In case of SRNS relocation, the RRC SN is exchanged between peer RNCs. How to ensure that the value is still aligned during the procedure?
Proposed way forward:

- CR of correction is needed. A new value may be sent from target RNC to UE in case of hard handover. How to cover soft handover?

Question/issue 6:
There is no diagram showing integrity protection in 25.301.
Proposed way forward:

- Align with ciphering. Add in 25.301, add in RRC, remove and reference 33.102?

3. Ciphering

Question/issue 1:
There is a diagram showing integrity protection in 25.301, duplicated in 33.102
Proposed way forward:

- Remove from 25.301? add in RRC? Refer to 33.102?

Question/issue 2:
In case of SRNS relocation with SHO, for Transparent mode (MAC case) current HFN is sent in RRC initialisation information, but may be invalid when received, leading to a loss of synchronisation
Proposed way forward:

- Send the CFN value when message was sent, and send the information on Iur? Or send the SFN value when message was sent, and send the information on Iu.

Question/issue 3:
In case of SRNS relocation with HHO, for Transparent mode (MAC case), current HFN is sent in RRC initialisation information, but may be invalid when received, leading to a loss of synchronisation. Also CFN seems missing.
Proposed way forward:

- New HFN+CFN is sent from target RNC, same as initial start of ciphering

Question/issue 4:
TDD ciphering is always synchronised on cell SFN, and therefore mechanism is different from FDD.
Proposed way forward:

- Align with FDD? Keep as it is?

Question/issue 5:
33.102 describes CFN for TM on 7 bits, 25.301 on 8 bits.
Proposed way forward:

- Align with FDD? Keep as it is?

Question/issue 6:
How is the HFN managed at RRC disconnection should be specified i.e. what is stored in the USIM.
Proposed way forward:

- In new RRC section

Question/issue 7:
Vocabulary on security, with definitions, need to be specified for the protocol.
Proposed way forward:

- CR on RRC

4. Conclusion

Depending on result of dicussions.

Also, the split between SA3 and R2 needs to be clearly identified so as to ensure that nothing is missing from the specifications.

CRs and suggestions should be provided to SA3 for consideration.

3GPP TSP RAN WG 2 Meeting #13
Oahu, Hawaii, USA, 22-26 May 2000

***Document* R2-001185**
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.301	CR	40	Current Version:	3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	RAN#8	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: Siemens **Date:** 24 May 2000

Subject: Replacement of duplicated information on ciphering description by references

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	X

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change:
Removal of duplicated information between diefferent specifications.
- Section 8 on ciphering is removed.
- References to the Security Architecture specification are included in the MAC, RLC, and RRC functions related to ciphering.

Clauses affected: 5.3.1.2, 5.3.2.2, 5.4.2, 8 (removed)

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

5.3.1.2 MAC functions

The functions of MAC include:

- **Mapping between logical channels and transport channels.** The MAC is responsible for mapping of logical channel(s) onto the appropriate transport channel(s).

- **Selection of appropriate Transport Format for each Transport Channel depending on instantaneous source rate**. Given the Transport Format Combination Set assigned by RRC, MAC selects the appropriate transport format within an assigned transport format set for each active transport channel depending on source rate**.** The control of transport formats ensures efficient use of transport channels.

- **Priority handling between data flows of one UE**. When selecting between the Transport Format Combinations in the given Transport Format Combination Set, priorities of the data flows to be mapped onto the corresponding Transport Channels can be taken into account. Priorities are e.g. given by attributes of Radio Bearer services and RLC buffer status. The priority handling is achieved by selecting a Transport Format Combination for which high priority data is mapped onto L1 with a "high bit rate" Transport Format, at the same time letting lower priority data be mapped with a "low bit rate" (could be zero bit rate) Transport Format. Transport format selection may also take into account transmit power indication from Layer 1.

- **Priority handling between UEs by means of dynamic scheduling**. In order to utilise the spectrum resources efficiently for bursty transfer, a dynamic scheduling function may be applied. MAC realises priority handling on common and shared transport channels. Note that for dedicated transport channels, the equivalent of the dynamic scheduling function is implicitly included as part of the reconfiguration function of the RRC sublayer.

NOTE: In the TDD mode the data to be transported are represented in terms of sets of resource units.

- **Identification of UEs on common transport channels**. When a particular UE is addressed on a common downlink channel, or when a UE is using the RACH, there is a need for inband identification of the UE. Since the MAC layer handles the access to, and multiplexing onto, the transport channels, the identification functionality is naturally also placed in MAC.

- **Multiplexing/demultiplexing of higher layer PDUs into/from transport blocks delivered to/from the physical layer on common transport channels**. MAC should support service multiplexing for common transport channels, since the physical layer does not support multiplexing of these channels.

- **Multiplexing/demultiplexing of higher layer PDUs into/from transport block sets delivered to/from the physical layer on dedicated transport channels**. The MAC allows service multiplexing for dedicated transport channels. This function can be utilised when several upper layer services (e.g. RLC instances) can be mapped efficiently on the same transport channel. In this case the identification of multiplexing is contained in the MAC protocol control information.

- **Traffic volume monitoring**. Measurement of traffic volume on logical channels and reporting to RRC. Based on the reported traffic volume information, RRC performs transport channel switching decisions.

- **Dynamic Transport Channel type switching.** Execution of the switching between common and dedicated transport channels based on a switching decision derived by RRC**.**

- **Ciphering**. This function prevents unauthorised acquisition of data. Ciphering is performed in the MAC layer for transparent RLC mode. Details of the security architecture are specified in [15].

- **Access Service Class selection for RACH transmission**. The RACH resources (i.e. access slots and preamble signatures for FDD, timeslot and channelisation code for TDD) may be divided between different Access Service Classes in order to provide different priorities of RACH usage. In addition it is possible for more than one ASC or for all ASCs to be assigned to the same access slot/signature space. Each access service class will also have a set of back-off parameters associated with it, some or all of which may be broadcast by the network. The MAC function applies the appropriate back-off and indicates to the PHY layer the RACH partition associated to a given MAC PDU transfer.

Next modifed Section

5.3.2.2 RLC Functions

- **Segmentation and reassembly.** This function performs segmentation/reassembly of variable-length higher layer PDUs into/from smaller RLC Payload Units (PUs). The RLC PDU size is adjustable to the actual set of transport formats.

NOTE: Multiple PUs in a RLC PDU is not supported in Release 99. For Release 99 an RLC PDU will include only a single RLC PU.

- **Concatenation.** If the contents of an RLC SDU do not fill an integer number of RLC PUs, the first segment of the next RLC SDU may be put into the RLC PU in concatenation with the last segment of the previous RLC SDU.

- **Padding.** When concatenation is not applicable and the remaining data to be transmitted does not fill an entire RLC PDU of given size, the remainder of the data field shall be filled with padding bits.

- **Transfer of user data.** This function is used for conveyance of data between users of RLC services. RLC supports acknowledged, unacknowledged and transparent data transfer. QoS setting controls transfer of user data.

- **Error correction.** This function provides error correction by retransmission (e.g. Selective Repeat, Go Back N, or a Stop-and-Wait ARQ) in acknowledged data transfer mode.

- **In-sequence delivery of higher layer PDUs**. This function preserves the order of higher layer PDUs that were submitted for transfer by RLC using the acknowledged data transfer service. If this function is not used, out-of-sequence delivery is provided.

- **Duplicate Detection.** This function detects duplicated received RLC PDUs and ensures that the resultant higher Layer PDU is delivered only once to the upper layer.

- **Flow control**. This function allows an RLC receiver to control the rate at which the peer RLC transmitting entity may send information.

- **Sequence number check (Unacknowledged data transfer mode).** This function guarantees the integrity of reassembled PDUs and provides a mechanism for the detection of corrupted RLC SDUs through checking sequence number in RLC PDUs when they are reassembled into a RLC SDU. A corrupted RLC SDU will be discarded.

- **Protocol error detection and recovery**. This function detects and recovers from errors in the operation of the RLC protocol.

- **Ciphering**. This function prevents unauthorised acquisition of data. Ciphering is performed in RLC layer for non-transparent RLC mode. Details of the security architecture are specified in [15].

- **Suspend/resume function**. Suspension and resumption of data transfer as in e.g. LAPDm (cf. GSM 04.05).

Next modifed Section

5.4.2 RRC functions

The Radio Resource Control (RRC) layer handles the control plane signalling of Layer 3 between the UEs and UTRAN. The RRC performs the following functions:

- **Broadcast of information provided by the non-access stratum (Core Network).** The RRC layer performs system information broadcasting from the network to all UEs. The system information is normally repeated on a regular basis. The RRC layer performs the scheduling, segmentation and repetition. This function supports broadcast of higher layer (above RRC) information. This information may be cell specific or not. As an example RRC may broadcast Core Network location service area information related to some specific cells.

- **Broadcast of information related to the access stratum.** The RRC layer performs system information broadcasting from the network to all UEs. The system information is normally repeated on a regular basis. The RRC layer performs the scheduling, segmentation and repetition. This function supports broadcast of typically cell-specific information.

- **Broadcast of ODMA relay node neighbour information.** The RRC layer performs probe information broadcasting to allow ODMA routeing information to be collected.

- **Establishment, re-establishment, maintenance and release of an RRC connection between the UE and UTRAN.** The establishment of an RRC connection is initiated by a request from higher layers at the UE side to establish the first Signalling Connection for the UE. The establishment of an RRC connection includes an optional cell re-selection, an admission control, and a layer 2 signalling link establishment. The release of an RRC connection can be initiated by a request from higher layers to release the last Signalling Connection for the UE or by the RRC layer itself in case of RRC connection failure. In case of connection loss, the UE requests re-establishment of the RRC connection. In case of RRC connection failure, RRC releases resources associated with the RRC connection.

- **Collating ODMA neighbour list and gradient information.** The ODMA relay node neighbour lists and their respective gradient information will be maintaining by the RRC.

- **Maintenance of number of ODMA relay node neighbours.** The RRC will adjust the broadcast powers used for probing messages to maintain the desired number of neighbours.

- **Establishment, maintenance and release of a route between ODMA relay nodes.** The establishment of an ODMA route and RRC connection based upon the routeing algorithm.

- **Interworking between the Gateway ODMA relay node and the UTRAN.** The RRC layer will control the interworking with the standard TDD or FDD communication link between the Gateway ODMA relay node and the UTRAN.

- **Establishment, reconfiguration and release of Radio Bearers.** The RRC layer can, on request from higher layers, perform the establishment, reconfiguration and release of Radio Bearers in the user plane. A number of Radio Bearers can be established to an UE at the same time. At establishment and reconfiguration, the RRC layer performs admission control and selects parameters describing the Radio Bearer processing in layer 2 and layer 1, based on information from higher layers.

- **Assignment, reconfiguration and release of radio resources for the RRC connection.** The RRC layer handles the assignment of radio resources (e.g. codes, CPCH channels) needed for the RRC connection including needs from both the control and user plane. The RRC layer may reconfigure radio resources during an established RRC connection. This function includes coordination of the radio resource allocation between multiple radio bearers related to the same RRC connection. RRC controls the radio resources in the uplink and downlink such that UE and UTRAN can communicate using unbalanced radio resources (asymmetric uplink and downlink). RRC signals to the UE to indicate resource allocations for purposes of handover to GSM or other radio systems.

- **RRC connection mobility functions.** The RRC layer performs evaluation, decision and execution related to RRC connection mobility during an established RRC connection, such as handover, preparation of handover to GSM or other systems, cell re-selection and cell/paging area update procedures, based on e.g. measurements done by the UE**.**

- **Paging/notification.** The RRC layer can broadcast paging information from the network to selected UEs. Higher layers on the network side can request paging and notification. The RRC layer can also initiate paging during an established RRC connection.

- **Routing of higher layer PDUs.** This function performs at the UE side routing of higher layer PDUs to the correct higher layer entity, at the UTRAN side to the correct RANAP entity.

- **Control of requested QoS**. This function shall ensure that the QoS requested for the Radio Bearers can be met. This includes the allocation of a sufficient number of radio resources.

- **UE measurement reporting and control of the reporting.** The measurements performed by the UE are controlled by the RRC layer, in terms of what to measure, when to measure and how to report, including both UMTS air interface and other systems. The RRC layer also performs the reporting of the measurements from the UE to the network.

- **Outer loop power control.** The RRC layer controls setting of the target of the closed loop power control.

- **Control of ciphering.** The RRC layer provides procedures for setting of ciphering (on/off) between the UE and UTRAN. Details of the security architecture are specified in [15].

- **Slow DCA**. Allocation of preferred radio resources based on long-term decision criteria. It is applicable only in TDD mode.

- **Arbitration of radio resources on uplink DCH**. This function controls the allocation of radio resources on uplink DCH on a fast basis, using a broadcast channel to send control information to all involved users.

NOTE: This function is implemented in the CRNC.

- **Initial cell selection and re-selection in idle mode.** Selection of the most suitable cell based on idle mode measurements and cell selection criteria.

- **Integrity protection.** This function adds a Message Authentication Code (MAC-I) to those RRC messages that are considered sensitive and/or contain sensitive information. The mechanism how the MAC-I is calculated is described in TS 33.105 [14].

- **Initial Configuration for CBS**
  This function performs the initial configuration of the BMC sublayer.

- **Allocation of radio resources for CBS**
  This function allocates radio resources for CBS based on traffic volume requirements indicated by BMC. The radio resource allocation set by RRC (i.e. the schedule for mapping of CTCH onto FACH/S-CCPCH) is indicated to BMC to enable generation of schedule messages. The resource allocation for CBS shall be broadcast as system information.

- **Configuration for CBS discontinuous reception**
  This function configures the lower layers (L1, L2) of the UE when it shall listen to the resources allocated for CBS based on scheduling information received from BMC.

- **Timing advance control.** The RRC controls the operation of timing advance. It is applicable only in TDD mode.

8 Ciphering

The ciphering architecture is specified in TS 33.102 [15].

8.1 Location of ciphering function in the UTRAN protocol architecture

The ciphering function is performed either in the RLC sub-layer or in the MAC sub-layer, according to the following rules:

- If a logical channel is expected to be supported on common transport channel and has to be ciphered, it can not use the transparent mode of RLC (it should use the UM RLC mode instead).

- If a logical channel is using a non-transparent RLC mode (AM or UM), ciphering is performed in the RLC sub-layer.

- If a logical channel is using the transparent RLC mode, ciphering is performed in the MAC sub-layer (MAC-d entity).

According to this model, ciphering when applied is performed in the SRNC and the UE, and the context needed for ciphering (CK, HFN, etc.) is only known in SRNC and the UE.

8.2 Input parameters to the ciphering algorithm

8.2.1 Overview

When ciphering is performed in the RLC sub-layer, it performs the encryption/decryption of the ciphering unit of an RLC PDU, based on XOR combining with a mask obtained as an output of the ciphering algorithm. For UM RLC, the

ciphering unit is defined as the UMD PDU minus the first octet. The first octet comprises the sequence number used as LSB of the COUNT parameter. For AM RLC, the ciphering unit is defined as the AMD PDU minus the two first octets. These two octets comprise the sequence number used as LSB of the COUNT parameter.

When ciphering is performed in the MAC sub-layer, it performs the encryption/decryption of a MAC SDU (RLC PDU), based on XOR operation with a mask obtained as an output of the ciphering algorithm.

Requirements and interfaces to the generic algorithm are specified in TS 33.105 and described in the following figure.

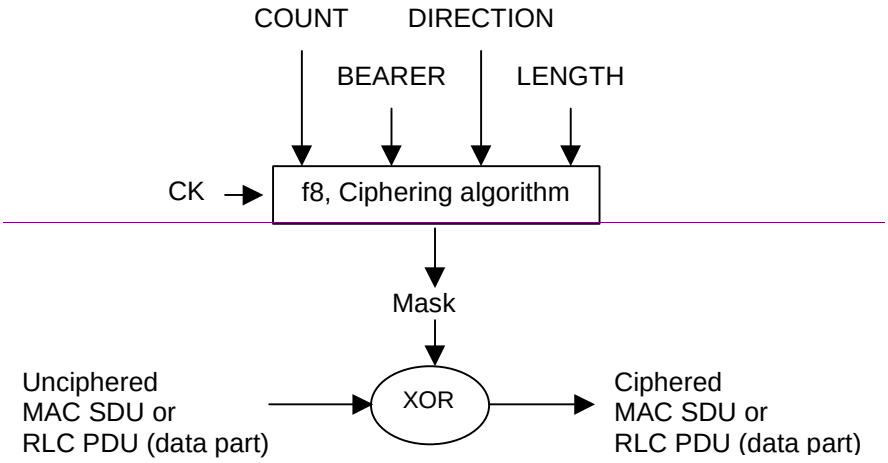

Figure 28: Ciphering algorithm and parameters

8.2.2 Ciphering algorithms parameters

8.2.2.1 COUNT

COUNT shall be at least 32 bits long. It is composed of a 'long' sequence number called Hyper Frame Number HFN, and a 'short' sequence number, which depends on the ciphering mode, as described below. There is one ciphering sequence per logical channel using AM or UM mode plus one for all logical channels using the transparent mode (and mapped onto DCH).

The Hyper Frame Number (HFN) is initialised by the UE and signalled to the SRNC before ciphering is started. It is used as initial value for each ciphering sequence, and it is then incremented independently in each ciphering sequence, at each cycle of the 'short' sequence number. When a new RAB / logical channel is created during a RRC connection, the highest HFN value currently in use is incremented, and used as initial value for the ciphering sequence of this new logical channel. The highest HFN value used during a RRC connection (by any ciphering sequence) is stored in the USIM, and the UE initialises the new HFN for the next session with a higher number than the stored one. If no HFN value is available in USIM, the UE randomly selects a HFN value.

Depending on the requirements (e.g. how many successive RRC Connections can use the same ciphering key), it may be sufficient to use only the most significant bits of HFN in the re-initialisation (and set LSBs implicitly to zero). This may be necessary at least if the HFN value needs to be included in the RRC Connection Request message.

The 'short' sequence number is:

- For RLC TM on DCH, the CFN of the UEFN is used and is independently maintained in UE MAC and SRNC MAC-d. The ciphering sequence number is identical to the UEFN.
- For RLC UM and AM modes, the RLC sequence number is used, and is directly available in each RLC PDU at the receiver side (it is not ciphered). The HFN is incremented at each RLC SN cycle.

The figure below presents some examples of the different COUNT parameters, assuming various sizes for the 'short' sequence numbers. This proposal permits to exchange a unique HFN and also to use a unique CSN size, which should permit to reduce the implementation complexity of the ciphering function. In this example, the HFN is 25 bits long, and only the 24 or 20 MSB are used for the CSN in the RLC modes TM or AM, respectively.

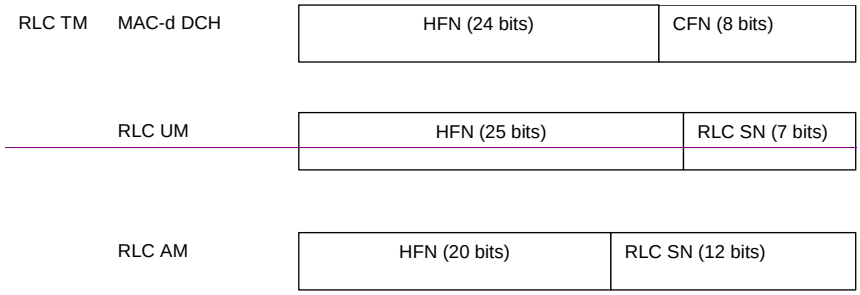

Figure 29: Example of ciphering sequence number for all possible configurations

8.2.2.2 Ciphering key, CK

CK is established between the UE and SRNC during the authentication phase. In the two-key solution, the CS-domain bearers are ciphered with the most recent cipher key agreed between the user and the 3G-MSC (CK-CS). The PS-domain bearers are ciphered with the most recent cipher key agreed between the user and the3G-SGSN (CK-PS). The signalling link is ciphered with the most recent cipher key established between the user and the network, i.e., the youngest of CK-CS and CK-PS.

To ensure performing the right ciphering function at the RLC and MAC layers, three conditions must be met:

- Each logical traffic channel can only transfer the information either from CS-domain or PS-domain, but not from both.
- RRC maps a given Radio Bearer to a given domain in order to derive the correct key to utilise for each RB.
- The RLC and MAC layers receive the Radio Bearer IDs and CKs they should use from RRC.

8.2.2.3 BEARER

This parameter indicates the logical channel identity, which shall be unique within a RRC connection. It is used as input parameter of the ciphering algorithm to ensure that the same ciphering mask is not applied to two or more parallel logical channels having the same CK and same COUNT. Each logical channel is ciphered independently.

8.2.2.4 Direction

This parameter indicates the transmission direction (uplink/downlink).

8.2.2.5 Length

This parameter indicates the length of the keystream block (mask) to be generated by the algorithm. It is not an input to the keystream generation function.

3GPP TSG RAN WG2#13
Oahu, HI, USA, 22nd-26th May, 2000

Document R2-001275
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.331	CR	358r2	Current Version:	3.4.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	RAN #8	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from:* ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: Ericsson **Date:** 2000-05-22

Subject: Clarification of HFN

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction	X	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

1) A new section has been added 8.5.10a Hyper Frame Number where the HFN and the initialisation of COUNT-C and COUNT-I are clearly described.

2) The IE "Hyper frame number" has been clarified.

Clauses affected: 8.5.10a (new), 10.3.3.13

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

8.5.10a Hyper Frame Number

~~There is one hyper frame number (HFN) for each CN Domain.~~ The hyper frame number (HFN) in the IE "Hyper frame number" is used to initialise both the ciphering sequence number (COUNT-C) and the integrity sequence number (COUNT-I) for the ciphering and integrity protection algorithms, respectively~~, for the corresponding service domain~~. There is a COUNT-C per radio bearer (uplink/downlink) and a COUNT-I per signalling radio bearer (uplink/downlink). COUNT-C and COUNT-I are defined in Security Architecture, TS 33.102.

~~For ciphering, HFN forms the:~~

~~24 MSB of COUNT-C, for a RB using transparent mode RLC~~

~~25 MSB of COUNT-C, for a RB using unacknowledged mode RLC~~

~~20 MSB of COUNT-C, for a RB using acknowledged mode RLC~~

~~For integrity protection, HFN forms the 28 MSB of COUNT-I..~~

~~For each CN Domain:~~

COUNT-C is initialised: COUNT-C = HFN (the LSB not part of the HFN in COUNT-C are set to zero).

COUNT-I is initialised: COUNT-I = HFN (the LSB not part of the HFN in COUNT-I are set to zero).

8.5.11 Integrity protection

Integrity protection shall be performed independently on the RRC messages sent on each signalling radio bearer.

For each signalling radio bearer, the UE shall use two integrity protection hyper frame numbers,

- "Uplink HFN";
- "Downlink HFN".

and two message sequence numbers,

- "Uplink RRC Message sequence number";
- "Downlink RRC Message sequence number".

The above information is stored in the variable INTEGRITY_PROTECTION_INFO per signalling radio bearer (0-3).

8.5.11.1 Integrity protection in downlink

If the UE receives an RRC message on signalling radio bearer with RB identity n, the "Status" in the variable INTEGRITY_ PROTECTION_INFO has the value "Started" and the IE 'Integrity check info' is present the UE shall:

- check the value of the IE "RRC message sequence number" included in the IE "Integrity check info". If the RRC message sequence number is lower than or equal to the "Downlink RRC Message sequence number" for RB#n in the variable INTEGRITY_PROTECTION_INFO, the UE shall increment "Downlink HFN" for RB#n in the variable INTEGRITY_PROTECTION_INFO with one.
- calculate an expected message authentication code in accordance with 8.5.11.3.
- compare the expected message authentication code with the value of the received IE "message authentication code" contained in the IE 'Integrity check info'.
    - If the expected message authentication code and the received message authentication code are the same, the integrity check is successful.
    - If the calculated expected message authentication code and the received message authentication code differ, the message shall be discarded.

If the UE receives an RRC message on signalling radio bearer with identity n, the "Status" in the variable INTEGRITY_PROTECTION_INFO has the value "Started" and the IE 'Integrity check info' is not present the UE shall discard the message.

10.3.3.13 Hyper Frame Number

The hyper frame number (HFN) is used to initialise both ~~the~~ COUNT-C and COUNT-I for the ciphering ~~algorithm and the COUNT-I~~ and integrity protection algorithms, respectively.

~~For ciphering, HFN forms the most significant bits of COUNT. When the COUNT is initialised: COUNT = HFN (the LSB part of COUNT is set to zero).For integrity protection, the HFN forms the most significant bits of COUNT-I. When the COUNT-I is initialised: COUNT-I = HFN (the LSB part of COUNT-I is set to zero).~~

Information Element/Group name	Need	Multi	Type and Reference	Semantics description
HFN	MP		Bit string (20)	Start value for uplink and downlink COUNT-C and COUNT-I. For RBs using RLC transparent mode, zeros shall be added, as LSB, to form a HFN of 24 bits. For ~~or~~ RLC unacknowledged mode, zeros shall be added, as LSB, to form a HFN of 25 bits. For integrity protection function, zeros shall be added, as LSB, to form a HFN of 28 bits.

3GPP TSG RAN WG2#13
Oahu, HI, USA, 22nd-26th May, 2000

Document R2-001276
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.331	CR	359r3	Current Version:	3.2.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	RAN #8	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from:* ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: *(at least one should be marked with an X)*	(U)SIM		ME	X	UTRAN / Radio	X	Core Network	

Source: Ericsson **Date:** 2000-05-22

Subject: Clarification of Integrity Protection

Work item:

Category: *(only one category Shall be marked With an X)*

F	Correction	X	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

1) This contribution clarifies the messages that are never integrity protected.

2) Integrity check info has been removed from the tabular description for the following messages:

PHYSICAL SHARED CHANNEL ALLOCATION
PUSCH CAPACITY REQUEST

No changes were required in the ASN.1 as the IE “Integrity check info” was not included.

3) When MEASUREMENT REPORT is sent on UM and if the UE is out of coverage then there could be problems with the synchronisation of the RRC SN if this is sent more than once, as there is no reponse message.
A note has been added that this needs to be studied.

4) Clarification of the calculation of the message authentication code. RRC padding is added after encoding and is therefore part of the message authentication code calculation.

NOTE:
a) The message SIGNALLING CONNECTION RELEASE REQUEST was included in CR315 to 25.331 but the tabular format is missing the IE “Integrity check info”.

Clauses affected: 8.5.11, 8.5.11.2, 8.5.11.3, 10.2.21, 10.2.22, 10.3.3.16

Other specs Affected:	Other 3G core specifications		→ List of CRs:	
	Other GSM core specifications		→ List of CRs:	

MS test specifications		→ List of CRs:	
BSS test specifications		→ List of CRs:	
O&M specifications		→ List of CRs:	

Other comments:

help.doc

<--------- double-click here for help and instructions on how to create a CR.

8.5.11 Integrity protection

Integrity protection shall be performed ~~independently~~ on all ~~the~~ RRC messages ~~sent on each signalling radio bearer.~~ , with the following exceptions ~~(as stated in TS 33.102)~~:

HANDOVER TO UTRAN COMPLETE

PAGING TYPE 1

PUSCH CAPACITY REQUEST

PHYSICAL SHARED CHANNEL ALLOCATION

RRC CONNECTION REQUEST

RRC CONNECTION SETUP

RRC CONNECTION SETUP COMPLETE

RRC CONNECTION REJECT

SYSTEM INFORMATION (BROADCAST~~ED~~ INFORMATION)

SYSTEM INFORMATION CHANGE INDICATION

TRANSPORT FORMAT CONTROL

Note: MEASUREMENT REPORT needs to be studied when used on UM as in some cases there could be synchronisation problems with the RRC SN.

For each signalling radio bearer, the UE shall use two integrity protection hyper frame numbers,

- "Uplink HFN";
- "Downlink HFN".

and two message sequence numbers,

- "Uplink RRC Message sequence number";
- "Downlink RRC Message sequence number".

The above information is stored in the variable INTEGRITY_PROTECTION_INFO per signalling radio bearer (0-3).

8.5.11.2 Integrity protection in uplink

Upon transmitting an RRC message using the signalling radio bearer with radio bearer identity n, and the "Status" in the variable INTEGRITY_ PROTECTION_INFO has the value "Started" the UE shall:

- increment "Uplink RRC Message sequence number" for RB#n in the variable INTEGRITY_PROTECTION_INFO with 1. When "Uplink RRC Message sequence number" for RB#n in the variable INTEGRITY_PROTECTION_INFO becomes 0, the UE shall increment "Uplink HFN" for RB#n in the variable INTEGRITY_PROTECTION_INFO with 1~~;~~
- calculate ~~a~~ the message authentication code in accordance with 8.5.11.3~~;~~

~~- include the IE "Integrity check info" in the message with contents set to the new value of the "Uplink RRC Message sequence number" for RB#n in the variable INTEGRITY_PROTECTION_INFO and~~

- replace the “Message authentication code“ in the IE "Integrity check info" in the message with the calculated message authentication code.

- replace the “RRC Message sequence number” in the IE "Integrity check info" in the message with contents set to the new value of the "Uplink RRC Message sequence number" for RB#n in the variable INTEGRITY_PROTECTION_INFO

8.5.11.3 Calculation of message authentication code

The UE shall calculate the message authentication code in accordance with 3G TS 33.102. The input parameter MESSAGE (TS 33.102) for the integrity algorithm shall be constructed by:

- setting the “Message authentication code“ in the IE "Integrity check info" in the message to the signalling radio bearer identity
- setting the “RRC Message sequence number” in the IE "Integrity check info" in the message to zero
- encoding the message
- appending RRC padding (if any) as a bitstring to the encoded bitstring as the least significant bits

~~The UE shall apply, after encoding, all the information elements in the message except the IE "Integrity check info", together with the signalling radio bearer identity as a bitstring, which is appended to the encoded bitstring as the most significant bits, to form the input parameter MESSAGE (TS 33.102) for the integrity algorithm. Note that the bitstring (radio bearer identity) is not part of the PDU to be transmitted.~~

10.2.21 PHYSICAL SHARED CHANNEL ALLOCATION

NOTE: Only for TDD.

This message is used by UTRAN to assign physical resources to USCH/DSCH transport channels in TDD, for temporary usage by the UE.

RLC-SAP: TM or UM

Logical channel: SHCCH

Direction: UTRAN → UE

Information Element	Need	Multi	Type and reference	Semantics description
Message Type	MP		Message type	
~~Integrity check info~~	~~CH~~		~~Integrity check info 10.3.3.16~~	
C-RNTI	MP		C-RNTI 10.3.3.7	
Uplink timing advance	MD		Uplink Timing Advance 10.3.6.69	Default value is the existing value for uplink timing advance
Allocation period info	OP		Allocation period info 10.3.6.4	
PUSCH info	OP		PUSCH info 10.3.6.46	
PDSCH info	OP		PDSCH info 10.3.6.30	
Timeslot list	OP	1 .. 14		
>Timeslot number	MP		Integer(0 .. 14)	Timeslot numbers, for which the UE shall report the timeslot ISCP in PUSCH CAPACITY REQUEST message.

10.2.22 PUSCH CAPACITY REQUEST

NOTE: Only for TDD.

This message is used by the UE for request of PUSCH resources to the UTRAN.

RLC-SAP: TM

Logical channel: SHCCH

Direction: UE → UTRAN

Information Element	Need	Multi	Type and reference	Semantics description
Message Type	MP		Message Type	
~~Integrity check info~~	~~CH~~		~~Integrity check info 10.3.3.16~~	
C-RNTI	MP		C-RNTI 10.3.3.7	
Traffic Volume	MP		Traffic Volume, measured results list 10.3.7.93	
Timeslot list	OP	1 .. 14		
>Timeslot number	MP		Integer(0 .. 14)	
>Timeslot ISCP	MP			
Primary CCPCH RSCP	OP			

10.3.3.16 Integrity check info

The Integrity check info contains the RRC message sequence number needed in the calculation of XMAC-I [TS 33.102] and the calculated MAC-I.

Information Element/Group name	Need	Multi	Type and reference	Semantics description
Message authentication code	MP		bit string(32)	MAC-I [TS 33.102] ~~The 27 MSB of the IE shall be set to zero and the 5 LSB of the IE shall be set to the used signalling radio bearer identity when the encoded RRC message is used as the MESSAGE parameter in the integrity protection algorithm.~~
RRC Message sequence number	MP		Integer (0..15)	The ~~local~~ RRC hyper frame number (HFN) is concatenated with the RRC message sequence number to form the input parameter COUNT-I for the integrity protection algorithm. ~~The IE value shall be set to zero when the encoded RRC message is used as the MESSAGE parameter in the integrity protection algorithm.~~

```
-- ************************************************
--
-- PHYSICAL SHARED CHANNEL ALLOCATION (TDD only)
--
```

```
-- **************************************************

PhysicalSharedChannelAllocation ::= SEQUENCE {
    -- User equipment IEs
        c-RNTI                          C-RNTI,
    -- Physical channel IEs
        ul-TimingAdvance                UL-TimingAdvance                    OPTIONAL,
        allocationPeriodInfo            AllocationPeriodInfo                OPTIONAL,
        pusch-Info                      PUSCH-Info                          OPTIONAL,
        pdsch-Info                      PDSCH-Info                          OPTIONAL,
        timeslotList                    TimeslotList                        OPTIONAL,
    -- Extension mechanism
        non-Release99-Information       SEQUENCE {}                         OPTIONAL
}

-- **************************************************
--
-- PUSCH CAPACITY REQUEST (TDD only)
--
-- **************************************************

PUSCHCapacityRequest ::= SEQUENCE {
    -- User equipment IEs
        c-RNTI                          C-RNTI,
    -- Measurement IEs
        trafficVolumeMeasuredResultsList
                                        TrafficVolumeMeasuredResultsList,
        timeslotListWithISCP            TimeslotListWithISCP                OPTIONAL,
        primaryCCPCH-RSCP               PrimaryCCPCH-RSCP                   OPTIONAL,
    -- Extension mechanism
        non-Release99-Information       SEQUENCE {}                         OPTIONAL
}
```

3GPP-RAN-WG2 Meeting #13
Oahu, Hawaii, USA, 22-26 May 2000

Document R2-001277

e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

TS25.331 **CR** **408r1** Current Version: 3.2.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: TSG RAN#8 *list expected approval meeting # here ↑*

for approval [X]
for information [ ]

strategic [ ]
non-strategic [ ]
(for SMG use only)

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [X] UTRAN / Radio [X] Core Network [ ]

Source: NTT DoCoMo **Date:** 22nd May, 2000

Subject: HFN Reset

Work item:

Category: *(only one category Shall be marked With an X)*

F	Correction	[ ]
A	Corresponds to a correction in an earlier release	[ ]
B	Addition of feature	[ ]
C	Functional modification of feature	[X]
D	Editorial modification	[ ]

Release:

Phase 2	[ ]
Release 96	[ ]
Release 97	[ ]
Release 98	[ ]
Release 99	[X]
Release 00	[ ]

Reason for change:

This CR proposes a reset mechanism of HFN for ciphering.

TM RLC

If UE have a capability of incrementing CFN precisely in itself in out-of-sync state, there will be no HFN inconsistency problem. However, if the UE doesn't support this function and the out-of-sync state continues, there will be an HFN inconsistency.

AM RLC

Since the Window_size is maximum 4096, there will be no case that HFN inconsistency occurs except for the case that AMD with SN=0 has not reached to the receiving side.

UM RLC

Since there is no restriction in Window_size, there is a possibility that HFN inconsistency occurs due to out-of-sync.

As shown above, there are possibilities that the HFN inconsistency between UE and NW occurs. This CR proposes to reset HFN by UE sending latest HFN+1, and the NW use it for new HFN.

This mechanism is proposed in RRC CONNECTION RE-ESATABLISHMENT REQUEST messages since HFN inconsistency may be caused by "out-of-sync in CELL_DCH state".

This mechanism is also proposed in CELL UPDATE messages since HFN inconsistency may be caused by "out of service area in CELL_FACH state".

Rev1

Highlighted part is the revised part.

UE shall include one latest HFN+1 in above 2 messages. The value to set is "the maximum

value in the currently used HFNs among CS and PS domains ” + “1”.

Clauses affected: 8.3.1.2, 8.1.5.2, 10.2.4, 10.2.37, 11.2

Other specs	Other 3G core specifications		→	List of CRs:
Affected:	Other GSM core specifications		→	List of CRs:
	MS test specifications		→	List of CRs:
	BSS test specifications		→	List of CRs:
	O&M specifications		→	List of CRs:

Other comments:


help.doc

<--------- double-click here for help and instructions on how to create a CR.

8.3.1 Cell update

8.3.1.1 General

8.3.1.2 Initiation

A UE in CELL_FACH, CELL_PCH or URA_PCH state may apply the cell update procedure for a number of purposes. The specific requirements the UE shall take into account for each case are specified in the following:

- Upon initiation of the procedure, the UE shall set the variable PROTOCOL_ERROR_INDICATOR to FALSE.
- In CELL_FACH or CELL_PCH state, the UE shall perform the cell update procedure when selecting another cell (cell reselection).
- In CELL_FACH and CELL_PCH state, the UE shall perform the cell update procedure upon expiry of T305 while the UE is in the service area. The UE shall only perform this periodic cell updating if configured by means of the IE "Information for periodical cell and URA update" in System Information Block Type 2. The UE shall initially start timer T305 upon entering CELL_FACH or CELL_PCH state.
- In CELL_PCH state and URA_PCH state, the UE shall initiate the cell update procedure if it wants to transmit UL data.
- In CELL_PCH and URA_PCH state, the UE shall perform the cell update procedure when receiving a PAGING TYPE 1 message as in subclause 8.1.2.3.
- moving to CELL_FACH state, if not already in that state.
- delete any C-RNTI and suspend data transmission on RB 2 and upward, if RLC-AM or RLC-UM is used on those radio bearers.
- sending a CELL UPDATE message on the uplink CCCH.
- starting timer T302 and resetting counter V302.

The IE "cell update cause" shall be used as follows:

- In case of cell reselection: "cell reselection";
- In case of periodic cell updating: "periodic cell update";
- In case of UL data transmission: "UL data transmission";
- In case of paging response: "paging response".

If the value of the variable PROTOCOL_ERROR_INDICATOR is TRUE, the UE shall set the IE "Protocol error indicator" to TRUE and include the IE "Protocol error information" set to the value of the variable PROTOCOL_ERROR_INFORMATION.

If the value of the variable PROTOCOL_ERROR_INDICATOR is FALSE, the UE shall set the IE "Protocol error indicator" to FALSE.

The IE "AM_RLC error indication" shall be set when the UE detects unrecoverable error in an AM RLC entity for the signalling link.

UE shall include “the maximum value in the currently used HFNs among CS and PS domains ” + “1” in IE “HFN” in CELL UPDATE message.

The UE shall include an intra-frequency measurement report in the CELL UPDATE message, as specified in the IE "Intra-frequency reporting quantity for RACH reporting" and the IE "Maximum number of reported cells on RACH" in system information block type 12.

8.1.5 RRC connection re-establishment

8.1.5.1 General

8.1.5.2 Initiation

When a UE loses the radio connection due to e.g. radio link failure (see 8.5.6) in CELL_DCH state, the UE may initiate a new cell selection by transiting to CELL_FACH state.

If timer T314=0 and timer T315=0 the UE shall:

- Enter idle mode. The procedure ends and a connection failure may be indicated to the non-access stratum. Other actions the UE shall perform when entering idle mode from connected mode are specified in subclause 8.5.2

If timer T314=0 the UE shall:

- Release locally all radio bearers (except Signalling Radio Bearers) using Tr or UM RLC. An indication may be sent to the non-access stratum.

If timer T315=0 the UE shall:

- Release locally all radio bearers (except Signalling Radio Bearers) using AM RLC. An indication may be sent to the non-access stratum.

If T314>0, the UE shall start timer T314.

If T315>0, the UE shall start timer T315.

Upon initiation of the procedure, the UE shall set the variable PROTOCOL_ERROR_INDICATOR to FALSE.

UE shall include “the maximum value in the currently used HFNs among CS and PS domains” plus “1” in IE “HFN” in RRC CONNECTION RE-ESTABLISHMENT REQUEST message.

10.2.4 CELL UPDATE

This message is used by the UE to initiate a cell update procedure.

RLC-SAP: TM

Logical channel: CCCH

Direction: UE→UTRAN

Information Element	Need	Multi	Type and reference	Semantics description
Message Type	MP		Message Type	
UE information elements				
U-RNTI	MP		U-RNTI 10.3.3.45	
Integrity check info	CH		Integrity check info 10.3.3.16	
Hyper frame number	MP		Hyper frame number 10.3.3.13	
AM_RLC error indication	MP		Boolean	TRUE indicates AM_RLC unrecoverable error occurred on c-plane in the UE
Cell update cause	MP		Cell update cause 10.3.3.3	
Protocol error indicator	MD		Protocol error indicator 10.3.3.29	Default value is FALSE
Measurement information elements				
Measured results on RACH	OP		Measured results on RACH 10.3.7.70	
Other information elements				
Protocol error information	CV-*ProtErr*		Protocol error information 10.3.8.9	

10.2.37 RRC CONNECTION RE-ESTABLISHMENT REQUEST

NOTE: Functional description of this message to be included here.

RLC-SAP: TM

Logical channel: CCCH

Direction: UE → UTRAN

Information Element	Need	Multi	Type and reference	Semantics description
Message Type	MP		Message Type	
UE information elements				
U-RNTI	MP		U-RNTI 10.3.3.45	
Integrity check info	CH		Integrity check info 10.3.3.16	
Hyper frame number	MP		Hyper frame number 10.3.3.13	
Protocol error indicator	MD		Protocol error indicator 10.3.3.29	Default value is FALSE
Measurement information elements				
Measured results on RACH	OP		Measured results on RACH 10.3.7.70	
Other information elements				
Protocol error information	CV-*ProtErr*		Protocol error information 10.3.8.9	

11.2 PDU definitions

```
CellUpdate ::= SEQUENCE {
    -- User equipment IEs
        u-RNTI                          U-RNTI,
        hyperFrameNumber                HyperFrameNumber,
        am-RLC-ErrorIndication          BOOLEAN,
        cellUpdateCause                 CellUpdateCause,
        protocolErrorIndicator          ProtocolErrorIndicatorWithInfo,
        -- TABULAR: Protocol error information is nested in
        -- ProtocolErrorIndicatorWithInfo.
    -- Measurement IEs
        measuredResultsOnRACH           MeasuredResultsOnRACH               OPTIONAL,
    -- Extension mechanism
        non-Release99-Information       SEQUENCE {}                         OPTIONAL
}


RRCConnectionReEstablishmentRequest ::= SEQUENCE {
    -- User equipment IEs
        u-RNTI                          U-RNTI,
        hyperFrameNumber                HyperFrameNumber,
        protocolErrorIndicator          ProtocolErrorIndicatorWithInfo,
        -- TABULAR: The IE above is MD in tabular, but making a 2-way choice
        -- optional wastes one bit (using PER) and produces no additional
        -- information.
    -- Measurement IEs
```

```
        measuredResultsOnRACH           MeasuredResultsOnRACH               OPTIONAL,
    -- Extension mechanism
        non-Release99-Information       SEQUENCE {}                         OPTIONAL
}
```

3GPP TSG RAN WG2#13
Oahu, HI, USA, 22-26 May, 2000

Document R2-001278
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.331	CR	409r1	Current Version:	3.2.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	RAN #8	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from:* ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: *(at least one should be marked with an X)*	(U)SIM		ME	X	UTRAN / Radio	X	Core Network	

Source: Nortel Networks **Date:** 2000-05-22

Subject: Clarification on ciphering parameters and integrity protection procedure in case of SRNS relocation

Work item:

Category: *(only one category shall be marked with an X)*

F	Correction		**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature	X		Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

1. Addition to the RRC Initialisation Information between network nodes in order to enable the ciphering procedure in case of SRNS relocation without hard handover. The change refers to RLC TM case.

2. Clarification on the ciphering/integrity procedure in case of SRNS relocation.

3. Minor editorial modifications

Clauses affected: 8.2.1, 8.2.2, 8.2.4, 8.2.6, 10.3.7.6, 11.X, 11.3.7, 14.10.1

Other specs Affected:

Other 3G core specifications		→ List of CRs:
Other GSM core specifications		→ List of CRs:
MS test specifications		→ List of CRs:
BSS test specifications		→ List of CRs:
O&M specifications		→ List of CRs:

Other comments: The changes on chapter 11.X took in account the ASN.1 description proposed in CR 397r1 to 25.331 (Tdoc R2-001253)

help.doc

<--------- double-click here for help and instructions on how to create a CR.

8.2 Radio Bearer control procedures

8.2.1 Radio bearer establishment

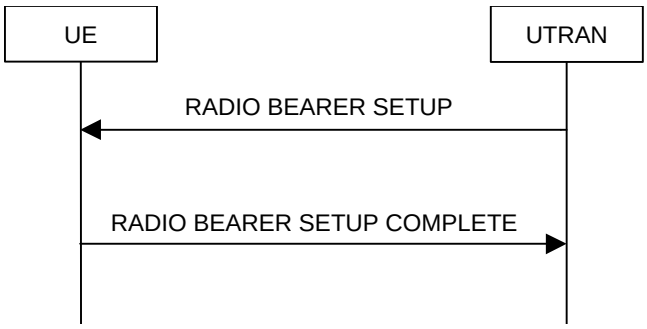

Figure 18: Radio Bearer Establishment, normal case

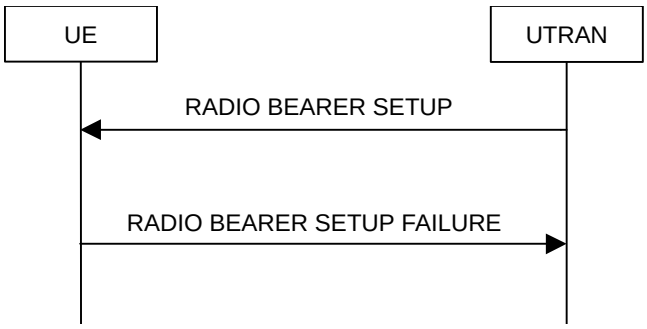

Figure 19: Radio Bearer Establishment, UE reverts to old configuration

8.2.1.1 General

The purpose with this procedure is to establish new radio bearer(s). Each radio bearer established by the procedure belongs to one of the following categories:

- a signalling radio bearer, i.e. used for control plane signalling;
- a radio bearer that implements a radio access bearer (RAB) or RAB subflow(s) in the user plane.

While establishing radio bearers, the procedure may perform a hard handover, see 8.3.5. The procedure may also be used to establish a transport channel for the transparent transfer of signalling.

8.2.1.2 Initiation

The upper layer in the network may request an establishment of radio bearer(s).

To initiate the procedure, UTRAN should:

- configures new radio links in any new physical channel configuration and start transmission and reception on the new radio links;
- transmits a RADIO BEARER SETUP message on the downlink DCCH using AM or UM RLC.

If the Radio Bearer Establishment procedure is simultaneous with SRNS relocation procedure, and ciphering and/or integrity protection are activated, transmit new ciphering and/or integrity protection information to be used after reconfiguration.

If transport channels are added, reconfigured or deleted in uplink and/or downlink, UTRAN shall:

- set TFCS according to the new transport channel(s).

If the IE "Activation Time" is included, UTRAN should set it to a value taking the UE performance requirements into account.

UTRAN should take the UE capabilities into account when setting the new configuration.

8.2.2 Radio bearer reconfiguration

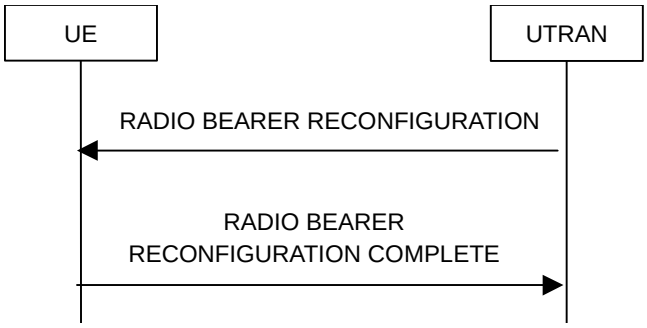

Figure 20: Radio bearer reconfiguration, normal flow

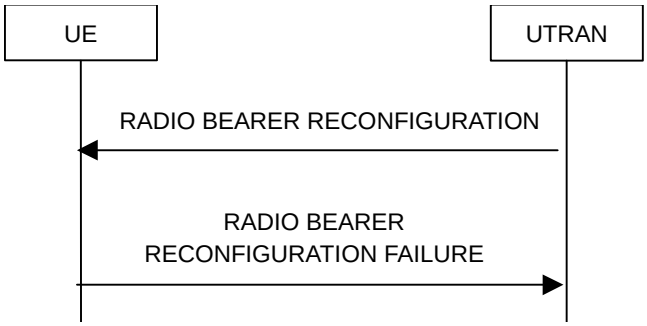

Figure 21: Radio bearer reconfiguration, failure case

8.2.2.1 General

The radio bearer reconfiguration procedure is used to reconfigure parameters for a radio bearer or the signalling link to reflect a change in QoS. While doing so, the procedure may perform a hard handover, see 8.3.5.

8.2.2.2 Initiation

To initiate the procedure, UTRAN should:~~The UTRAN initiates the procedure by~~:

- configur~~e~~ing new radio links in any new physical channel configuration and start transmission and reception on the new radio links;
- t~~T~~ransmit~~ting~~ a RADIO BEARER RECONFIGURATION message on the downlink DCCH using AM or UM RLC

If the Radio Bearer Reconfiguration procedure is simultaneous with SRNS relocation procedure, and ciphering and/or integrity protection are activated, transmit new ciphering and/or integrity protection information to be used after reconfiguration~~.~~

If transport channels are added, reconfigured or deleted in uplink and/or downlink, the UTRAN shall:

- Set TFCS according to the new transport channel(s).

UTRAN should indicate that uplink transmission shall be suspended on certain bearers. Uplink transmission on a radio bearer used by the RRC signalling should not be suspended.

If the IE "Activation Time" is included, UTRAN should set it to a value taking the UE performance requirements into account.

UTRAN should take the UE capabilities into account when setting the new configuration.

If the message is used to initiate a transition from CELL_DCH to CELL_FACH state, the UTRAN may assign a common channel configuration of a given cell and C-RNTI to be used in that cell to the UE.

8.2.4 Transport channel reconfiguration

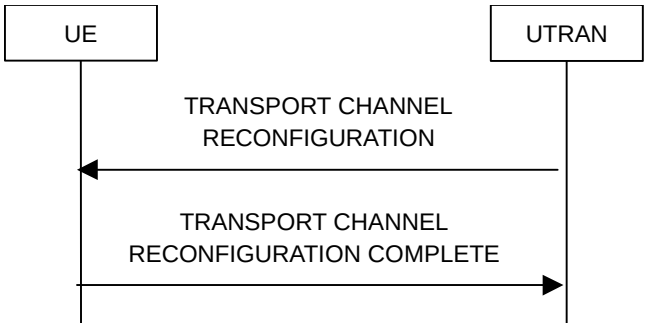

Figure 24: Transport channel reconfiguration, normal flow

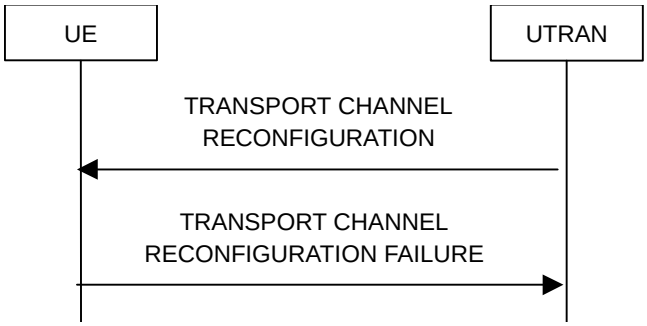

Figure 25: Transport channel reconfiguration, failure case

8.2.4.1 General

The transport channel reconfiguration procedure is used to reconfigure transport channel parameters. While doing so, the procedure may perform a hard handover, see 8.3.5.

8.2.4.2 Initiation

To initiate the procedure, UTRAN should:~~The UTRAN shall:~~

- Configure new radio links in any new physical channel configuration and start transmission and reception on the new radio links.
- transmit a TRANSPORT CHANNEL RECONFIGURATION message on the downlink DCCH using AM or UM RLC.

If the Transport Channel Reconfiguration procedure is simultaneous with SRNS relocation procedure, and ciphering and/or integrity protection are activated, transmit new ciphering and/or integrity protection information to be used after reconfiguration.

If transport channels are added, reconfigured or deleted in uplink and/or downlink, the UTRAN shall:

- Set TFCS according to the new transport channel(s).

If the IE "Activation Time" is included, UTRAN should set it to a value taking the UE performance requirements into account.

UTRAN should take the UE capabilities into account when setting the new configuration.

8.2.6 Physical channel reconfiguration

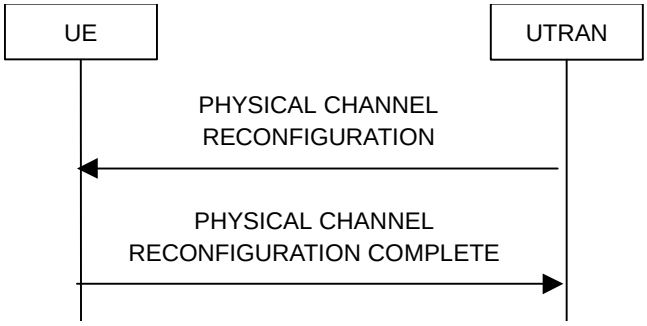

Figure 27: Physical channel reconfiguration, normal flow

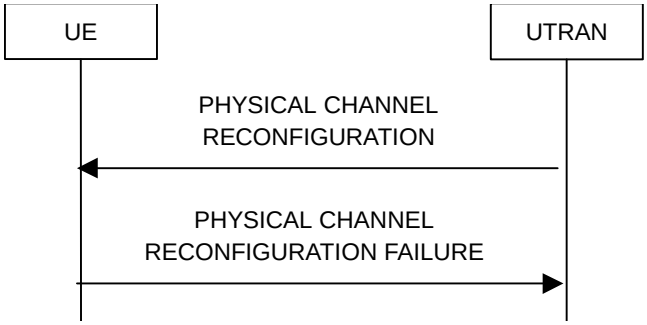

Figure 28: Physical channel reconfiguration, failure case

8.2.6.1 General

The physical channel reconfiguration procedure is used to establish, reconfigure and release physical channels. While doing so, the procedure may perform a hard handover, see 8.3.5.

8.2.6.2 Initiation

To initiate the procedure, the UTRAN should:

- Configure new radio links in any new physical channel configuration and start transmission and reception on the new radio links.
- transmit a PHYSICAL CHANNEL RECONFIGURATION message on the downlink DCCH using AM or UM RLC.

If the Physical Channel Reconfiguration procedure is simultaneous with SRNS relocation procedure, and ciphering and/or integrity protection are activated, transmit new ciphering and/or integrity protection information to be used after reconfiguration.

UTRAN should take the UE capabilities into account when setting the new configuration.

If the message is used to initiate a transition from CELL_DCH to CELL_FACH state, the UTRAN may assign a common channel configuration of a given cell and C-RNTI to be used in that cell to the UE.

14.10 Provision and reception of RRC information between network nodes

14.10.1 RRC Initialisation Information, source RNC to target RNC

When relocation of SRNS is decided to be executed, the RRC shall build the state information, which contains the RRC, RLC and MAC related RRC message information elements, which currently specify the state of the RRC including the radio bearer and transport channel configuration. This "RRC initialisation information, source RNC to target RNC" shall be sent by the source RNC to the target RNC to enable transparent relocation of the RRC and lower layer protocols. Correspondingly, the RRC in the target RNC shall receive the "RRC initialisation information, source RNC to target RNC" and update its state parameters accordingly to facilitate a transparent relocation of SRNS for the UE.

Information Element	Need	Multi	Type and reference	Semantics description
Non RRC IEs				
State of RRC	M		Enumerated (CELL_DCH, CELL_FACH,CELL_PCH, URA_PCH)	
State of RRC procedure	M		Enumerated (await no RRC message, await RRC Connection Re-establishment Complete, await RB Setup Complete, await RB Reconfiguration Complete, await RB Release Complete, await Transport CH Reconfiguration Complete, await Physical CH Reconfiguration Complete, await Active Set Update Complete, await Handover Complete, others)	
Variable RLC parameters	M		?????	
Ciphering related information				
Ciphering status	M		Enumerated(Not started, Started)	
Calculation time for ciphering related information	CV *Ciphering*			Time when the ciphering information of the message were calculated, relative to a cell of the target RNC
>Cell Identity	MP		Cell Identity 10.3.2.2	Identity of one of the cells under the target RNC and included in the active set of the current call
>SFN	MP		Integer(0..4095)	
Ciphering info per radio bearer		0 to < numberOfRadioBearers>		
>RB identity	M		RB identity	
>Downlink HFN	M		Ciphering hyperframe number	
>Uplink HFN	M		Ciphering hyperframe number	
>Downlink RLC sequence Number	O		Integer(0..4095)	RLC SN [TS 25.322]
>Uplink RLC sequence number	O		Integer(0..4095)	RLC SN [TS 25.322]
Integrity protection related information				
Integrity protection status	M		Enumerated(Not started, Started)	
Integrity protection failure count	M		Integer(0..N316)	
Signalling radio bearer specific integrity protection information	CV *IP*	3 to <maxSRBcount>		Status information for RB#0-3 in that order
> Uplink HFN	M		Integrity protection hyper frame number	

Information Element	Need	Multi	Type and reference	Semantics description
> Downlink HFN	M		Integrity protection hyper frame number	
> Uplink RRC Message sequence number	M		Integer (0.. 15)	
> Downlink RRC Message sequence number	M		Integer (0.. 15)	
Implementation specific parameters	O		Bitstring (1..512)	
RRC IEs				
UE Information elements				
U-RNTI	M			
C-RNTI	O			
UE radio access Capability	M			
Other Information elements				
Inter System message (inter system classmark)	O			
UTRAN Mobility Information elements				
URA Identifier	O			
CN Information Elements				
CN common GSM-MAP NAS system information	M		GSM-MAP NAS system information	
CN domain related information		0 to <MaxNo CNdomains>		CN related information to be provided for each CN domain
>CN domain identity	O			
>CN domain specific GSM-MAP NAS system info	O		GSM-MAP NAS system information	
Measurement Related Information elements				
For each ongoing measurement reporting		0 to <maxNo OfMeas>		
Measurement Identity Number	M			
Measurement Command	M			
Measurement Type	C Setup			
Measurement Reporting Mode	O			
Additional Measurement Identity number				
CHOICE Measurement				
Intra-frequency				
Intra-frequency cell info		0 to <MaxIntr aCells>		
Intra-frequency measurement quantity	O			
Intra-frequency reporting quantity	O			
Reporting cell status	O			
Measurement validity	O			
CHOICE report criteria	O			
Intra-frequency measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
Inter-frequency				
Inter-frequency cell info		0 to <MaxInte rCells>		
Inter-frequency measurement quantity	O			
Inter-frequency reporting quantity	O			
Reporting cell status	O			

Information Element	Need	Multi	Type and reference	Semantics description
Measurement validity	O			
CHOICE report criteria	O			
Inter-frequency measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
Inter-system				
Inter-system cell info		0 to <MaxInterSysCells>		
Inter-system measurement quantity	O			
Inter-system reporting quantity	O			
Reporting cell status	O			
Measurement validity				
CHOICE report criteria				
Inter-system measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
Traffic Volume				
Traffic volume measurement Object	O			
Traffic volume measurement quantity	O			
Traffic volume reporting quantity	O			
CHOICE report criteria	O			
Traffic volume measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
Quality				
Quality measurement Object	O			
Quality measurement quantity	O			
Quality reporting quantity	O			
CHOICE report criteria	O			
Quality measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
UE internal				
UE internal measurement quantity	O			
UE internal reporting quantity	O			
CHOICE report criteria	O			
UE internal measurement reporting criteria				
Periodical reporting				
No reporting			NULL	
Radio Bearer Information Elements				
Signalling radio bearer information		3 to <maxSRBcount>		For each signalling radio bearer
>RB identity	M			
>RLC info	M			
>RB mapping info	M			
RAB information		0 to <maxRABcount>		Information for each RAB

Information Element	Need	Multi	Type and reference	Semantics description
>RAB info	M			
>For each Radio Bearer		0 to <maxRB count>		Information for each radio bearer belonging to this RAB
>>RB Identity	M			
>>RLC Info	M			
>>PDCP Info	O			Absent ifPDCP is not configured for RB
>>PDCP SN Info	C PDCP			
>>RB mapping info	M			
Transport Channel Information Elements				
TFCS (UL DCHs)	O			
TFCS (DL DCHs)	O			
TFC subset (UL DCHs)	O			
TFCS (USCHs)	O			
TFCS (DSCHs)	O			
TFC subset (USCHs)	O			
Uplink transport channels				
For each uplink transport channel		0 to <MaxTrCH>		
>Transport channel identity	M			
>TFS	M			
Downlink transport channels				
For each downlink transport channel		0 to <MaxTrCH>		
>Transport channel identity	M			
>TFS	M			
Measurement report	O			MEASUREMENT REPORT 10.1.15

Condition	Explanation
Ciphering	The IE is mandatory when the IE Ciphering Status has the value “started” and the ciphering counters need not be reinitialised, otherwise the IE is not needed.
IP	The IE is mandatory when the IE Integrity protection status has the value “started” and the ciphering counters need not be reinitialised, otherwise the IE is not needed.
PDCP	The IE is only present when PDCP Info IE is present

10.3.7.6 CFN-SFN observed time difference

NOTE: Only for FDD.

The measured time difference to cell indicates the time difference that is measured by UE between ~~CFN~~ RLC Transparent Mode COUNT-C in the UE and the SFN of the target neighbouring cell. It is notified to SRNC by Measurement Report message or Measurement Information Element in other RRC messages. This measurement is for FDD only.

Information Element/Group name	Need	Multi	Type and reference	Semantics description
CFN-SFN observed time difference	MP		Enumerated(0..~~9830399~~ 157286399)	Number of chip

11.4 Constant definitions

```
maxNoOfMeas                    INTEGER ::= 16
```

11.x RRC information between network nodes

```
Internode-definitions DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS

    HandoverToUTRANCommand,
    MeasurementReport,
    PhysicalChannelReconfiguration,
    RadioBearerReconfiguration,
    RadioBearerRelease,
    RadioBearerSetup,
    TransportChannelReconfiguration,
    UECapabilityInformation
FROM PDU-definitions

    CN-DomainInformationList,
    NAS-SystemInformationGSM-MAP
FROM CoreNetwork-IEs

    URA-Identity
    CellIdentity
FROM UTRANMobility-IEs

    C-RNTI,
    HyperFrameNumber,
    RRC-MessageSequenceNumber,
    U-RNTI,
    UE-RadioAccessCapability
FROM UserEquipment-IEs

    PDCP-InfoReconfig,
    RAB-Info,
    RB-Identity,
    RB-MappingInfo,
    RLC-Info,
    RLC-SequenceNumber,
    SRB-InformationSetup
FROM RadioBearer-IEs

    TFC-Subset,
    TFCS,
    TransportChannelIdentity,
    TransportFormatSet
FROM TransportChannel-IEs

    MeasurementIdentityNumber,
    MeasurementReportingMode,
    MeasurementType,
    AdditionalMeasurementID-List
FROM Measurement-IEs

    InterSystemMessage
FROM Other-IEs

    maxNoOfMeas,
    maxRABcount,
    maxRBcount,
    maxSRBcount,
    maxTrCH
FROM Constant-definitions;

CalculationTimeForCiphering ::=     SEQUENCE {
    cell-Id                             CellIdentity,
    sfn                                 Integer (0..4095)
}
CipheringInfoPerRB ::=              SEQUENCE {
    dl-HFN                              HyperFrameNumber,
    ul-HFN                              HyperFrameNumber,
```

```
    dl-RLC-SequenceNumber                RLC-SequenceNumber,
    ul-RLC-SequenceNumber                RLC-SequenceNumber
}

-- TABULAR: Multiplicity value numberOfRadioBearers has been replaced
-- with maxRBcount.
CipheringInfoPerRB-List ::=         SEQUENCE (SIZE (1..maxRBcount)) OF
                                        CipheringInfoPerRB

CipheringStatus ::=                 ENUMERATED {
                                        started, notStarted }

ImplementationSpecificParams ::=    BIT STRING (SIZE (1..512))

-- **TODO** Upper limit N316 is undefined! An arbitrary upper limit of
-- 7 has been used here instead.
IntegrityProtectionFailureCount ::= INTEGER (0..7)

IntegrityProtectionStatus ::=       ENUMERATED {
                                        started, notStarted }

MeasurementCommandWithType ::=      CHOICE {
    setup                               MeasurementType,
    modify                              NULL,
    release                             NULL
}

OngoingMeasRep ::=                  SEQUENCE {
    measurementIdentityNumber           MeasurementIdentityNumber,
    measurementCommandWithType          MeasurementCommandWithType,
    -- TABULAR: The CHOICE Measurement in the tabular description is included
    -- in the IE above.
    measurementReportingMode            MeasurementReportingMode            OPTIONAL,
    additionalMeasurementID-List        AdditionalMeasurementID-List        OPTIONAL
}

OngoingMeasRepList ::=              SEQUENCE (SIZE (1..maxNoOfMeas)) OF
                                        OngoingMeasRep

RAB-Information ::=                 SEQUENCE {
    rab-Info                            RAB-Info,
    rb-InformationList                  RB-InformationList                  OPTIONAL
}

RAB-InformationList ::=             SEQUENCE (SIZE (1..maxRABcount)) OF
                                        RAB-Information

RB-Information ::=                  SEQUENCE {
    rb-Identity                         RB-Identity,
    rlc-Info                            RLC-Info,
    pdcp-Info                           PDCP-InfoReconfig                   OPTIONAL,
    rb-MappingInfo                      RB-MappingInfo
}

RB-InformationList ::=              SEQUENCE (SIZE (1..maxRBcount)) OF
                                        RB-Information

-- ************************************************
--
-- Source RNC to target RNC
--
-- ************************************************

SourceRNCToTargetRNC ::=            SEQUENCE {
    -- Non-RRC IEs
        stateOfRRC                      StateOfRRC,
        stateOfRRC-Procedure            StateOfRRC-Procedure,
        cipheringStatus                 CipheringStatus,
        calculationTimeForCiphering     CalculationTimeForCiphering         OPTIONAL,
        cipheringInfoPerRB-List         CipheringInfoPerRB-List             OPTIONAL,
        integrityProtectionStatus       IntegrityProtectionStatus,
        integrityProtectionFailureCount IntegrityProtectionFailureCount,
        srb-SpecificIntegrityProtInfo   SRB-SpecificIntegrityProtInfoList,
        implementationSpecificParams    ImplementationSpecificParams        OPTIONAL,
    -- User equipment IEs
        u-RNTI                          U-RNTI,
        c-RNTI                          C-RNTI                              OPTIONAL,
        ue-RadioAccessCapability        UE-RadioAccessCapability,
```

```
    -- Other IEs
        interSystemMessage              InterSystemMessage                  OPTIONAL,
    -- UTRAN mobility IEs
        ura-Identity                    URA-Identity                        OPTIONAL,
    -- Core network IEs
        cn-CommonGSM-MAP-NAS-SysInfo    NAS-SystemInformationGSM-MAP,
        cn-DomainInformationList        CN-DomainInformationList            OPTIONAL,
    -- Measurement IEs
        ongoingMeasRepList              OngoingMeasRepList                  OPTIONAL,
    -- Radio bearer IEs
        srb-InformationList             SRB-InformationList,
        rab-InformationList             RAB-InformationList                 OPTIONAL,
    -- Transport channel IEs
        ul-DCH-TFCS                     TFCS                                OPTIONAL,
        dl-DCH-TFCS                     TFCS                                OPTIONAL,
        ul-DCH-TFC-Subset               TFC-Subset                          OPTIONAL,
        usch-TFCS                       TFCS                                OPTIONAL,
        dsch-TFCS                       TFCS                                OPTIONAL,
        usch-TFC-Subset                 TFC-Subset                          OPTIONAL,
        ul-TransChInfoList              TransChInfoList                     OPTIONAL,
        dl-TransChInfoList              TransChInfoList                     OPTIONAL,
    -- Measurement report
        measurementReport               MeasurementReport                   OPTIONAL
}

-- ************************************************
--
-- Source system to target RNC
--
-- ************************************************

SourceSystemToTargetRNC ::=         CHOICE {
    ueCapabilityInformation             UECapabilityInformation,
    spare                               NULL
}

SRB-InformationList ::=             SEQUENCE (SIZE (3..maxSRBcount)) OF
                                        SRB-InformationSetup

SRB-SpecificIntegrityProtInfo ::=   SEQUENCE {
    ul-HFN                              HyperFrameNumber,
    dl-HFN                              HyperFrameNumber,
    ul-RRC-SequenceNumber               RRC-MessageSequenceNumber,
    dl-RRC-SequenceNumber               RRC-MessageSequenceNumber
}

SRB-SpecificIntegrityProtInfoList ::= SEQUENCE (SIZE (3..maxSRBcount)) OF
                                        SRB-SpecificIntegrityProtInfo

StateOfRRC ::=                      ENUMERATED {
                                        cell-DCH, cell-FACH,
                                        cell-PCH, ura-PCH }

StateOfRRC-Procedure ::=            ENUMERATED {
                                        awaitNoRRC-Message,
                                        awaitRRC-ConnectionRe-establishmentComplete,
                                        awaitRB-SetupComplete,
                                        awaitRB-ReconfigurationComplete,
                                        awaitTransportCH-ReconfigurationComplete,
                                        awaitPhysicalCH-ReconfigurationComplete,
                                        awaitActiveSetUpdateComplete,
                                        awaitHandoverComplete,
                                        otherStates }

-- ************************************************
--
-- Target system to source RNC
--
-- ************************************************

TargetSystemToSourceRNC ::=         CHOICE {
    radioBearerSetup                    RadioBearerSetup,
    radioBearerReconfiguration          RadioBearerReconfiguration,
    radioBearerRelease                  RadioBearerRelease,
    transportChannelReconfiguration     TransportChannelReconfiguration,
    physicalChannelReconfiguration      PhysicalChannelReconfiguration,
    handoverToUTRANCommand              HandoverToUTRANCommand
}
```

```
TransChInfo ::=                     SEQUENCE {
    transportChannelIdentity            TransportChannelIdentity,
    transportFormatSet                  TransportFormatSet
}

TransChInfoList ::=                 SEQUENCE (SIZE (1..maxTrCH)) OF
                                        TransChInfo

END
```

11.3.7 Measurement information elements

```
Measurement-IEs DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS

    CellIdentity
FROM UTRANMobility-IEs

    DRX-CycleLengthCoefficient
FROM UserEquipment-IEs

    RB-Identity
FROM RadioBearer-IEs

    TransportChannelIdentity
FROM TransportChannel-IEs

    FrequencyInfo,
    MaxAllowedUL-TX-Power,
    PrimaryCCPCH-Info,
    PrimaryCCPCH-TX-Power,
    PrimaryCPICH-Info,
    PrimaryCPICH-TX-Power,
    Timeslot
FROM PhysicalChannel-IEs

    BSIC
FROM Other-IEs

    maxAdditionalMeas,
    maxAddRLcount,
    maxBLER,
    maxCCTrCHcount,
    maxCellCount,
    maxCellsForbidden,
    maxDelRLcount,
    maxEventCount,
    maxFreqCount,
    maxInterCells,
    maxInterRAT,
    maxInterSys,
    maxInterSysCells,
    maxIntraCells,
    maxN-BadSAT,
    maxN-SAT,
    maxNoCells,
    maxNonUsedFrequency,
    maxNumFreq,
    maxTraf,
    maxTrCHcount,
    maxTSperCCTrCHcount,
    maxTStoMeasureCount,
    maxUsedRLcount,
    maxUsedUplTScount
FROM Constant-definitions;

AcquisitionSatInfo ::=              SEQUENCE {
    satID                               INTEGER (0..63),
    doppler0thOrder                     INTEGER (-2048..2047),
    extraDopplerInfo                    ExtraDopplerInfo                OPTIONAL,
    codePhase                           INTEGER (0..1022),
    integerCodePhase                    INTEGER (0..19),
    gps-BitNumber                       INTEGER (0..3),
    codePhaseSearchWindow               CodePhaseSearchWindow,
    azimuthAndElevation                 AzimuthAndElevation             OPTIONAL
}

AcquisitionSatInfoList ::=          SEQUENCE (SIZE (1..maxN-SAT)) OF
                                        AcquisitionSatInfo

ActiveSetCellReport ::=             ENUMERATED {
                                        includeAll,
                                        excludeAll,
                                        other }
```

```
-- **TODO**, definition to be checked from TS 09.31
AdditionalAssistanceData ::=        SEQUENCE {
}

AdditionalMeasurementID-List ::=    SEQUENCE (SIZE (1..maxAdditionalMeas)) OF
                                        MeasurementIdentityNumber

AlmanacSatInfo ::=                  SEQUENCE {
    satID                               INTEGER (0..63),
    deltaI                              BIT STRING (SIZE (16)),
    e                                   BIT STRING (SIZE (16)),
    m0                                  BIT STRING (SIZE (24)),
    a-Sqrt                              BIT STRING (SIZE (24)),
    omega0                              BIT STRING (SIZE (24)),
    omegaDot                            BIT STRING (SIZE (16)),
    omega                               BIT STRING (SIZE (24)),
    af0                                 BIT STRING (SIZE (11)),
    af1                                 BIT STRING (SIZE (11))
}

AlmanacSatInfoList ::=              SEQUENCE (SIZE (1..maxN-SAT)) OF
                                        AlmanacSatInfo

AverageRLC-BufferPayload ::=        ENUMERATED {
                                        pla0, pla4, pla8, pla16, pla32,
                                        pla64, pla128, pla256, pla512,
                                        pla1024, pla2k, pla4k, pla8k, pla16k }

AzimuthAndElevation ::=             SEQUENCE {
    azimuth                             INTEGER (0..31),
    elevation                           INTEGER (0..7)
}

BadSatList ::=                      SEQUENCE (SIZE (1..maxN-BadSAT)) OF
                                        INTEGER (0..63)

BCCH-ARFCN ::=                      INTEGER (0..1023)

BLER-MeasurementResults ::=         SEQUENCE {
    transportChannelIdentity            TransportChannelIdentity,
    dl-TransportChannelBLER             DL-TransportChannelBLER             OPTIONAL
}

BLER-MeasurementResultsList ::=     SEQUENCE (SIZE(1..maxBLER)) OF
                                        BLER-MeasurementResults

BLER-TransChIdList ::=              SEQUENCE (SIZE (1..maxBLER)) OF
                                        TransportChannelIdentity

-- IE value 0 = true value -0.05, IE value 16 = true value -0.003125,
-- IE value 17 = true value 0.003125, IE value 32 = true value 0.05
BTS-ClockDrift ::=                  INTEGER (0..31)

BurstModeParameters ::=             SEQUENCE {
    burstStart                          INTEGER (0..15),
    burstLength                         INTEGER (10..25),
    burstFreq                           INTEGER (1..16)
}

CCTrCH-Timeslot ::=                 SEQUENCE {
    iscp                                DL-TimeslotISCP                    OPTIONAL,
    rscp                                RSCP                               OPTIONAL
}

CCTrCH-TimeslotList ::=             SEQUENCE (SIZE(1..maxTSperCCTrCHcount)) OF
                                        CCTrCH-Timeslot

CellDCH-ReportCriteria ::=          CHOICE {
    intraFreqReportingCriteria          IntraFreqReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria
}

-- Actual value = IE value * 0.5
CellIndividualOffset ::=            INTEGER (-20..20)

CellInfo ::=                        SEQUENCE {
    cellIndividualOffset                CellIndividualOffset               DEFAULT 1,
```

```
    referenceTimeDifferenceToCell       ReferenceTimeDifferenceToCell       OPTIONAL,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            primaryCPICH-Info                   PrimaryCPICH-Info           OPTIONAL,
            primaryCPICH-TX-Power               PrimaryCPICH-TX-Power       OPTIONAL,
            readSFN-Indicator                   BOOLEAN,
            tx-DiversityIndicator               BOOLEAN
        },
        tdd                                 SEQUENCE {
            primaryCCPCH-Info                   PrimaryCCPCH-Info,
            primaryCCPCH-TX-Power               PrimaryCCPCH-TX-Power,
            dl-CCTrCH-Info                      DL-CCTrCH-Info              OPTIONAL,
            dl-TimeslotInfo                     DL-TimeslotInfo             OPTIONAL
        }
    }
}

CellInfoSI ::=                      SEQUENCE {
    cellIndividualOffset                CellIndividualOffset                DEFAULT 1,
    referenceTimeDifferenceToCell       ReferenceTimeDifferenceToCell       OPTIONAL,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            primaryCPICH-Info                   PrimaryCPICH-Info           OPTIONAL,
            primaryCPICH-TX-Power               PrimaryCPICH-TX-Power       OPTIONAL,
            readSFN-Indicator                   BOOLEAN,
            tx-DiversityIndicator               BOOLEAN
        },
        tdd                                 SEQUENCE {
            primaryCCPCH-Info                   PrimaryCCPCH-Info,
            primaryCCPCH-TX-Power               PrimaryCCPCH-TX-Power,
            dl-CCTrCH-Info                      DL-CCTrCH-Info              OPTIONAL,
            dl-TimeslotInfo                     DL-TimeslotInfo             OPTIONAL
        }
    },
    cellSelectionReselectionInfo        CellSelectionReselectionInfo,
    signallingOption                    SignallingOption
}

CellMeasuredResults ::=             SEQUENCE {
    cellIdentity                        CellIdentity                        OPTIONAL,
    sfn-SFN-ObsTimeDifference           SFN-SFN-ObsTimeDifference           OPTIONAL,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            primaryCPICH-Info                   PrimaryCPICH-Info,
            cpich-Ec-N0                         CPICH-Ec-N0                 OPTIONAL,
            cpich-RSCP                          CPICH-RSCP                  OPTIONAL,
            cpich-SIR                           CPICH-SIR                   OPTIONAL,
            pathloss                            Pathloss                    OPTIONAL,
            cfn-SFN-ObsTimeDifference           CFN-SFN-ObsTimeDifference   OPTIONAL
        },
        tdd                                 SEQUENCE {
            primaryCCPCH-Info                   PrimaryCCPCH-Info,
            dl-CCTrCH-SIR-List                  DL-CCTrCH-SIR-List          OPTIONAL,
            dl-TimeslotISCP-List                DL-TimeslotISCP-List        OPTIONAL
        }
    }
}

CellMeasurementEventResults ::=     CHOICE {
    fdd                                 SEQUENCE (SIZE (1..maxCellCount)) OF
                                            PrimaryCPICH-Info,
    tdd                                 SEQUENCE (SIZE (1..maxCellCount)) OF
                                            PrimaryCCPCH-Info
}

CellPosition ::=                    SEQUENCE {
    relativeNorth                       INTEGER (-32767..32767),
    relativeEast                        INTEGER (-32767..32767),
    relativeAltitude                    INTEGER (-4095..4095)
}

CellReportingQuantities ::=         SEQUENCE {
    sfn-SFN-OTD-Type                    SFN-SFN-OTD-Type,
    cellIdentity                        CellIdentity,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            cpich-Ec-N0                         BOOLEAN,
            cpich-RSCP                          BOOLEAN,
```

```
            cpich-SIR                           BOOLEAN,
            pathloss                            BOOLEAN,
            cfn-SFN-ObsTimeDifference           BOOLEAN
        },
        tdd                                 SEQUENCE {
            dl-CCTrCH-SIR                       BOOLEAN,
            timeslotISCP                        BOOLEAN,
            primaryCCPCH-RSCP                   BOOLEAN,
            pathloss                            BOOLEAN
        }
    }
}

CellSelectionReselectionInfo ::=    SEQUENCE {
    modeSpecificInfo                    CHOICE {
        fdd                                 Qmin-FDD,
        tdd                                 Qmin-TDD
    }                                                               OPTIONAL,
    maxAllowedUL-TX-Power               MaxAllowedUL-TX-Power       OPTIONAL,
    signallingOption                    SignallingOption
}

CellToMeasure ::=                   SEQUENCE {
    sfn-sfn-Drift                       INTEGER (0..30)             OPTIONAL,
    primaryCPICH-Info                   PrimaryCPICH-Info,
    frequencyInfo                       FrequencyInfo               OPTIONAL,
    sfn-SFN-ObservedTimeDifference      SFN-SFN-ObsTimeDifference1,
    fineSFN-SFN                         FineSFN-SFN,
    cellPosition                        CellPosition                OPTIONAL
}

CellToMeasureInfoList ::=           SEQUENCE (SIZE (1..maxNoCells)) OF
                                        CellToMeasure

CellToReport ::=                    SEQUENCE {
    frequency                           Frequency,
    bsic                                BSIC
}

CellToReportList ::=                SEQUENCE (SIZE (1..maxCellCount)) OF
                                        CellToReport

CFN-SFN-ObsTimeDifference ::=       INTEGER (0..98303991572863999)

CodePhaseSearchWindow ::=           ENUMERATED {
                                        w1023, w1, w2, w3, w4, w6, w8,
                                        w12, w16, w24, w32, w48, w64,
                                        w96, w128, w192 }

CompressedNavModel ::=              SEQUENCE {
    iode                                BIT STRING (SIZE (4)),
    t-oe                                BIT STRING (SIZE (7)),
    c-rc                                BIT STRING (SIZE (12)),
    c-rs                                BIT STRING (SIZE (12)),
    c-ic                                BIT STRING (SIZE (9)),
    c-is                                BIT STRING (SIZE (9)),
    c-uc                                BIT STRING (SIZE (11)),
    c-us                                BIT STRING (SIZE (11)),
    e                                   BIT STRING (SIZE (16)),
    m0                                  BIT STRING (SIZE (22)),
    a-Sqrt                              BIT STRING (SIZE (13)),
    delta-n                             BIT STRING (SIZE (11)),
    omega0                              BIT STRING (SIZE (14)),
    omegaDot                            BIT STRING (SIZE (12)),
    i0                                  BIT STRING (SIZE (15)),
    iDot                                BIT STRING (SIZE (11)),
    omega                               BIT STRING (SIZE (21)),
    t-oc                                BIT STRING (SIZE (7)),
    af0                                 BIT STRING (SIZE (7)),
    af1                                 BIT STRING (SIZE (3)),
    af2                                 BIT STRING (SIZE (1))
}

CPICH-Ec-N0 ::=                     INTEGER (-20..0)

-- IE value 0 = <-24 dB, 1 = between -24 and -23 and so on
CPICH-Ec-N0-OTDOA ::=               INTEGER (0..26)
```

```
CPICH-RSCP ::=                      INTEGER (-115..-40)

CPICH-SIR ::=                       INTEGER (-10..20)

DGPS-CorrectionSatInfo ::=          SEQUENCE {
    satID                               INTEGER (0..63),
    iode                                BIT STRING (SIZE (8)),
    udre                                UDRE,
    prc                                 INTEGER (-2048..2048),
    rrc                                 INTEGER (-125..125),
    deltaPRC2                           INTEGER (-127..127),
    deltaRRC2                           INTEGER (-7..7),
    deltaPRC3                           INTEGER (-127..127),
    deltaRRC3                           INTEGER (-7..7)
}

DGPS-CorrectionSatInfoList ::=      SEQUENCE (SIZE (1..maxN-SAT)) OF
                                        DGPS-CorrectionSatInfo

DGPS-Information ::=                SEQUENCE {
    satID                               SatID,
    iode                                IODE,
    udre                                UDRE,
    scaleFactor                         ScaleFactor,
    prc                                 PRC,
    rrc                                 RRC
}

DGPS-InformationList ::=            SEQUENCE (SIZE (1..maxN-SAT)) OF
                                        DGPS-Information

DiffCorrectionStatus ::=            ENUMERATED {
                                        udre-1-0, udre-0-75, udre-0-5, udre-0-3,
                                        udre-0-2, udre-0-1, noData, invalidData }

-- **TODO**, not defined yet
DL-CCTrCH-Info ::=                  SEQUENCE {
}

DL-CCTrCH-SIR ::=                   SEQUENCE {
    ccTrCH-TimeslotList                 CCTrCH-TimeslotList
}

DL-CCTrCH-SIR-List ::=              SEQUENCE (SIZE(1..maxCCTrCHcount)) OF
                                        DL-CCTrCH-SIR

-- Actual value = IE value * 0.02
DL-PhysicalChannelBER ::=           INTEGER (0..255)

-- **TODO**, not defined yet
DL-TimeslotInfo ::=                 SEQUENCE {
}

-- **TODO**, not defined yet
DL-TimeslotISCP ::=                 SEQUENCE {
}

DL-TimeslotISCP-List ::=            SEQUENCE (SIZE(1..maxTStoMeasureCount)) OF
                                        DL-TimeslotISCP

-- Actual value = IE value * 0.02
DL-TransportChannelBLER ::=         INTEGER (0..255)

DopplerUncertainty ::=              ENUMERATED {
                                        hz12-5, hz25, hz50, hz100, hz200 }

EnvironmentCharacterization ::=     ENUMERATED {
                                        possibleHeavyMultipathNLOS,
                                        lightMultipathLOS,
                                        notDefined }

Event1a ::=                         SEQUENCE {
    triggeringCondition                 TriggeringCondition,
    reportingRange                      ReportingRange,
    forbiddenAffectCellList             ForbiddenAffectCellList,
    w                                   W,
    hysteresis                          Hysteresis                          OPTIONAL,
    reportDeactivationThreshold         ReportDeactivationThreshold
```

```
}

Event1b ::=                         SEQUENCE {
    triggeringCondition                 TriggeringCondition,
    reportingRange                      ReportingRange,
    forbiddenAffectCellList             ForbiddenAffectCellList,
    w                                   W,
    hysteresis                          Hysteresis                          OPTIONAL
}

Event1c ::=                         SEQUENCE {
    hysteresis                          Hysteresis                          OPTIONAL,
    replacementActivationThreshold      ReplacementActivationThreshold
}

Event2a ::=                         SEQUENCE {
    usedFreqThreshold                   Threshold,
    usedFreqW                           W,
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval,
    nonUsedFreqParameterList            NonUsedFreqParameterList            OPTIONAL
}

Event2b ::=                         SEQUENCE {
    usedFreqThreshold                   Threshold,
    usedFreqW                           W,
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval,
    nonUsedFreqParameterList            NonUsedFreqParameterList            OPTIONAL
}

Event2c ::=                         SEQUENCE {
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval,
    nonUsedFreqParameterList            NonUsedFreqParameterList            OPTIONAL
}

Event2d ::=                         SEQUENCE {
    usedFreqThreshold                   Threshold,
    usedFreqW                           W,
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

Event2e ::=                         SEQUENCE {
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval,
    nonUsedFreqParameterList            NonUsedFreqParameterList            OPTIONAL
}

Event2f ::=                         SEQUENCE {
    usedFreqThreshold                   Threshold,
    usedFreqW                           W,
    hysteresis                          HysteresisInterFreq,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

Event3a ::=                         SEQUENCE {
    thresholdOwnSystem                  Threshold,
    w                                   W,
    thresholdOtherSystem                Threshold,
    hysteresis                          Hysteresis,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}
```

```
Event3b ::=                         SEQUENCE {
    thresholdOtherSystem                Threshold,
    hysteresis                          Hysteresis,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

Event3c ::=                         SEQUENCE {
    thresholdOtherSystem                Threshold,
    hysteresis                          Hysteresis,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

Event3d ::=                         SEQUENCE {
    hysteresis                          Hysteresis,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

EventIDInterFreq ::=                ENUMERATED {
                                        e2a, e2b, e2c, e2d, e2e, e2f }

EventIDInterSystem ::=              ENUMERATED {
                                        e3a, e3b, e3c, e3d }

EventIDIntraFreq ::=                ENUMERATED {
                                        e1a, e1b, e1c, e1d, e1e,
                                        e1f, e1g, e1h, e1i, e1j }

EventIDTrafficVolume ::=            ENUMERATED {
                                        e4a, e4b }

EventResults ::=                    CHOICE {
    intraFreqEventResults               IntraFreqEventResults,
    interFreqEventResults               InterFreqEventResults,
    interSystemEventResults             InterSystemEventResults,
    trafficVolumeEventResults           TrafficVolumeEventResults,
    qualityEventResults                 QualityEventResults,
    ue-InternalEventResults             UE-InternalEventResults,
    lcs-MeasurementEventResults         LCS-MeasurementEventResults
}

ExtraDopplerInfo ::=                SEQUENCE {
    doppler1stOrder                     INTEGER (-42..21),
    dopplerUncertainty                  DopplerUncertainty
}

FACH-MeasurementOccasionInfo ::=    SEQUENCE {
    k-UTRA                              DRX-CycleLengthCoefficient,
    otherRAT-InSysInfoList              OtherRAT-InSysInfoList
}

FilterCoefficient ::=               ENUMERATED {
                                        fc1, fc2, fc3, fc4, fc6, fc8,
                                        fc12, fc16, fc24, fc32, fc64,
                                        fc128, fc256, fc512, fc1024,
                                        spare1 }

FineSFN-SFN ::=                     ENUMERATED {
                                        fs0, fs0-25, fs0-5, fs0-75 }

ForbiddenAffectCell ::=             SEQUENCE {
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            primaryCPICH-Info                   PrimaryCPICH-Info
        },
        tdd                                 SEQUENCE {
            primaryCCPCH-Info                   PrimaryCCPCH-Info
        }
    }
}

ForbiddenAffectCellList ::=         SEQUENCE (SIZE(1..maxCellsForbidden)) OF
```

```
                                    ForbiddenAffectCell

FreqQualityEstimateQuantity-FDD ::= ENUMERATED {
                                    cpich-Ec-N0,
                                    cpich-RSCP }

FreqQualityEstimateQuantity-TDD ::= ENUMERATED {
                                    primaryCCPCH-RSCP }

-- **TODO**, not defined yet
Frequency ::=                       SEQUENCE {
}

GPS-MeasurementParam ::=            SEQUENCE {
    satelliteID                         INTEGER (0..63),
    c-N0                                INTEGER (0..63),
    doppler                             INTEGER (-32768..32768),
    wholeGPS-Chips                      INTEGER (0..1023),
    fractionalGPS-Chips                 INTEGER (0..1023),
    multipathIndicator                  MultipathIndicator,
    pseudorangeRMS-Error                INTEGER (0..63)
}

GPS-MeasurementParamList ::=        SEQUENCE (SIZE (1..maxN-SAT)) OF
                                    GPS-MeasurementParam

GPS-TOW-1msec ::=                   INTEGER (0..604700000)

GPS-TOW-Assist ::=                  SEQUENCE {
    satID                               INTEGER (0..63),
    tlm-Message                         BIT STRING (SIZE (14)),
    antiSpoof                           BOOLEAN,
    alert                               BOOLEAN,
    tlm-Reserved                        BIT STRING (SIZE (2))
}

GPS-TOW-AssistList ::=              SEQUENCE (SIZE (1..maxN-SAT)) OF
                                    GPS-TOW-Assist

GPS-TOW-HighResolution ::=          INTEGER (0..999)

GSM-CarrierRSSI ::=                 BIT STRING (SIZE (6))

-- **TODO**, not defined yet
GSM-OutputPower ::=                 SEQUENCE {
}

HCS-CellReselectInformation ::=     SEQUENCE {
    penaltyTime                         PenaltyTime
}

HCS-NeighbouringCellInformation ::= SEQUENCE {
    hcs-PRIO                            HCS-PRIO                            OPTIONAL,
    q-HCS                               Q-HCS                               OPTIONAL,
    hcs-CellReselectInformation         HCS-CellReselectInformation         OPTIONAL
}

HCS-PRIO ::=                        INTEGER (0..7)

-- Actual value = IE value * 0.5
Hysteresis ::=                      INTEGER (0..15)

-- Actual value = IE value * 0.5
HysteresisInterFreq ::=             INTEGER (0..29)

InterFreqCell ::=                   SEQUENCE {
    frequencyInfo                       FrequencyInfo,
    nonFreqRelatedEventResults          CellMeasurementEventResults
}

InterFreqCellID ::=                 INTEGER (0..maxInterCells)

InterFreqCellInfoList ::=           SEQUENCE {
    removedInterFreqCellList            RemovedInterFreqCellList            OPTIONAL,
    newInterFreqCellList                NewInterFreqCellList                OPTIONAL
}

InterFreqCellInfoSI-List ::=        SEQUENCE {
```

```
    removedInterFreqCellList            RemovedInterFreqCellList            OPTIONAL,
    newInterFreqCellList                NewInterFreqCellSI-List             OPTIONAL
}

InterFreqCellList ::=               SEQUENCE (SIZE (1..maxFreqCount)) OF
                                        InterFreqCell

InterFreqCellMeasuredResultsList ::= SEQUENCE (SIZE (1..maxInterCells)) OF
                                        CellMeasuredResults

InterFreqEvent ::=                  CHOICE {
    event2a                             Event2a,
    event2b                             Event2b,
    event2c                             Event2c,
    event2d                             Event2d,
    event2e                             Event2e,
    event2f                             Event2f
}

InterFreqEventList ::=              SEQUENCE (SIZE(1..maxEventCount)) OF
                                        InterFreqEvent

InterFreqEventResults ::=           SEQUENCE {
    eventID                             EventIDInterFreq,
    interFreqCellList                   InterFreqCellList
}

InterFreqMeasQuantity ::=           SEQUENCE {
    reportingCriteria                   CHOICE {
        intraFreqReportingCriteria          SEQUENCE {
            intraFreqMeasQuantity               IntraFreqMeasQuantity,
        },
        interFreqReportingCriteria          SEQUENCE {
            filterCoefficient                   FilterCoefficient,
            modeSpecificInfo                    CHOICE {
                fdd                                 SEQUENCE {
                    freqQualityEstimateQuantity-FDD     FreqQualityEstimateQuantity-FDD
                },
                tdd                                 SEQUENCE {
                    freqQualityEstimateQuantity-TDD     FreqQualityEstimateQuantity-TDD
                }
            }
        }
    }
}

InterFreqMeasuredResults ::=        SEQUENCE {
    frequencyInfo                       FrequencyInfo                       OPTIONAL,
    utra-CarrierRSSI                    UTRA-CarrierRSSI                    OPTIONAL,
    interFreqCellMeasuredResultsList    InterFreqCellMeasuredResultsList    OPTIONAL
}

InterFreqMeasuredResultsList ::=    SEQUENCE (SIZE (1..maxNumFreq)) OF
                                        InterFreqMeasuredResults

InterFreqMeasurementSysInfo ::=     SEQUENCE {
    interFreqMeasurementID              MeasurementIdentityNumber           OPTIONAL,
    interFreqCellInfoSI-List            InterFreqCellInfoSI-List            OPTIONAL,
    interFreqMeasQuantity               InterFreqMeasQuantity               OPTIONAL
}

InterFreqReportCriteria ::=         CHOICE {
    intraFreqReportingCriteria          IntraFreqReportingCriteria,
    interFreqReportingCriteria          InterFreqReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

InterFreqReportingCriteria ::=      SEQUENCE {
    interFreqEventList                  InterFreqEventList                  OPTIONAL
}

InterFreqReportingQuantity ::=      SEQUENCE {
    utra-Carrier-RSSI                   BOOLEAN,
    frequencyQualityEstimate            BOOLEAN,
    nonFreqRelatedQuantities            CellReportingQuantities
}
```

```
InterFreqSetUpdate ::=              SEQUENCE {
    ue-AutonomousUpdateMode             UE-AutonomousUpdateMode
}

InterFrequencyMeasurement ::=       SEQUENCE {
    interFreqCellInfoList               InterFreqCellInfoList,
    interFreqMeasQuantity               InterFreqMeasQuantity               OPTIONAL,
    interFreqReportingQuantity          InterFreqReportingQuantity          OPTIONAL,
    reportingCellStatus                 ReportingCellStatus                 OPTIONAL,
    measurementValidity                 MeasurementValidity                 OPTIONAL,
    interFreqSetUpdate                  InterFreqSetUpdate                  OPTIONAL,
    reportCriteria                      InterFreqReportCriteria
}

InterSystemCellID ::=               INTEGER (0..maxInterSysCells)

InterSystemCellInfoList ::=         SEQUENCE {
    removedInterSystemCellList          RemovedInterSystemCellList,
    newInterSystemCellList              NewInterSystemCellList
}

InterSystemEvent ::=                CHOICE {
    event3a                             Event3a,
    event3b                             Event3b,
    event3c                             Event3c,
    event3d                             Event3d
}

InterSystemEventList ::=            SEQUENCE (SIZE(1..maxEventCount)) OF
                                        InterSystemEvent

InterSystemEventResults ::=         SEQUENCE {
    eventID                             EventIDInterSystem,
    cellToReportList                    CellToReportList
}

InterSystemInfo ::=                 ENUMERATED {
                                        gsm, spare1 }

InterSystemMeasQuantity ::=             SEQUENCE {
    measQuantityUTRAN-QualityEstimate       IntraFreqMeasQuantity,
    systemSpecificInfo                      CHOICE {
        gsm                                     SEQUENCE {
            measurementQuantity                     MeasurementQuantityGSM,
            filterCoefficient                       FilterCoefficient,
            bsic-VerificationRequired               BOOLEAN
        },
        is-2000                                 SEQUENCE {
            tadd-EcIo                               INTEGER (0..63),
            tcomp-EcIo                              INTEGER (0..15),
            softSlope                               INTEGER (0..63)         OPTIONAL,
            addIntercept                            INTEGER (0..63)         OPTIONAL
        }
    }
}

InterSystemMeasuredResults ::=      CHOICE {
    gsm                                 SEQUENCE {
        frequency                           Frequency,
        gsm-CarrierRSSI                     GSM-CarrierRSSI                 OPTIONAL,
        pathloss                            Pathloss                        OPTIONAL,
        bsic                                BSIC                            OPTIONAL,
        observedTimeDifferenceToGSM         ObservedTimeDifferenceToGSM     OPTIONAL
    },
    other                               NULL
}

InterSystemMeasuredResultsList ::= SEQUENCE (SIZE (1..maxInterSys)) OF
                                        InterSystemMeasuredResults

InterSystemMeasurement ::=          SEQUENCE {
    interSystemCellInfoList             InterSystemCellInfoList             OPTIONAL,
    interSystemMeasQuantity             InterSystemMeasQuantity             OPTIONAL,
    interSystemReportingQuantity        InterSystemReportingQuantity        OPTIONAL,
    reportingCellStatus                 ReportingCellStatus                 OPTIONAL,
    reportCriteria                      InterSystemReportCriteria
}
```

```
InterSystemMeasurementSysInfo ::=   SEQUENCE {
    interSystemMeasurementID            MeasurementIdentityNumber           OPTIONAL,
    interSystemCellInfoList             InterSystemCellInfoList             OPTIONAL,
    interSystemMeasQuantity             InterSystemMeasQuantity             OPTIONAL
}

InterSystemReportCriteria ::=       CHOICE {
    interSystemReportingCriteria        InterSystemReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

InterSystemReportingCriteria ::=    SEQUENCE {
    interSystemEventList                InterSystemEventList                OPTIONAL
}

InterSystemReportingQuantity ::=    SEQUENCE {
    utran-EstimatedQuality              BOOLEAN,
    systemSpecificInfo                  CHOICE {
        gsm                                 SEQUENCE {
            pathloss                            BOOLEAN,
            observedTimeDifferenceGSM           BOOLEAN,
            gsm-Carrier-RSSI                    BOOLEAN,
            bsic                                BOOLEAN
        },
        spare1                              SEQUENCE {}
    }
}

IntraFreqCellID ::=                 INTEGER (0..maxIntraCells)

IntraFreqCellInfoList ::=           SEQUENCE {
    removedIntraFreqCellList            RemovedIntraFreqCellList            OPTIONAL,
    newIntraFreqCellList                NewIntraFreqCellList                OPTIONAL
}

IntraFreqCellInfoSI ::=             SEQUENCE {
    cellInfo                            CellInfoSI
}

IntraFreqCellInfoSI-List ::=        SEQUENCE {
    removedIntraFreqCellList            RemovedIntraFreqCellList            OPTIONAL,
    newIntraFreqCellList                NewIntraFreqCellSI-List             OPTIONAL
}

IntraFreqEvent ::=                  CHOICE {
    e1a                                 Event1a,
    e1b                                 Event1b,
    e1c                                 Event1c,
    e1d                                 Hysteresis,
    e1e                                 TriggeringCondition,
    e1f                                 TriggeringCondition,
    e1g                                 Hysteresis,
    e1h                                 Hysteresis,
    e1i                                 Hysteresis,
    e1j                                 Hysteresis
}

IntraFreqEventCriteria ::=          SEQUENCE {
    event                               IntraFreqEvent,
    timeToTrigger                       TimeToTrigger,
    reportingAmount                     ReportingAmount,
    reportingInterval                   ReportingInterval
}

IntraFreqEventCriteriaList ::=      SEQUENCE (SIZE(1..maxEventCount)) OF
                                        IntraFreqEventCriteria

IntraFreqEventResults ::=           SEQUENCE {
    eventID                             EventIDIntraFreq,
    cellMeasurementEventResults         CellMeasurementEventResults
}

IntraFreqMeasQuantity ::=           SEQUENCE {
    filterCoefficient                   FilterCoefficient,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            intraFreqMeasQuantity-FDD           IntraFreqMeasQuantity-FDD
```

```
        },
        tdd                             SEQUENCE {
            intraFreqMeasQuantity-TDD       IntraFreqMeasQuantity-TDD
        }
    }
}

IntraFreqMeasQuantity-FDD ::=       ENUMERATED {
                                        cpich-Ec-NO,
                                        cpich-RSCP,
                                        cpich-SIR,
                                        pathloss,
                                        utra-CarrierRSSI }

IntraFreqMeasQuantity-TDD ::=       ENUMERATED {
                                        primaryCCPCH-RSCP,
                                        pathloss,
                                        timeslotISCP,
                                        utra-CarrierRSSI }

IntraFreqMeasuredResults ::=        SEQUENCE {
    cellMeasuredResults                 CellMeasuredResults
}

IntraFreqMeasuredResultsList ::=    SEQUENCE (SIZE (1..maxIntraCells)) OF
                                        IntraFreqMeasuredResults

IntraFreqMeasurementSysInfo ::=     SEQUENCE {
    intraFreqMeasurementID              MeasurementIdentityNumber           OPTIONAL,
    intraFreqCellInfoSI-List            IntraFreqCellInfoSI-List            OPTIONAL,
    intraFreqMeasQuantity               IntraFreqMeasQuantity               OPTIONAL,
    intraFreqReportingQuantityForRACH   IntraFreqReportingQuantityForRACH   OPTIONAL,
    maxReportedCellsOnRACH              MaxReportedCellsOnRACH              OPTIONAL,
    reportingInfoForCellDCH             ReportingInfoForCellDCH             OPTIONAL
}

IntraFreqReportCriteria ::=         CHOICE {
    intraFreqReportingCriteria          IntraFreqReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

IntraFreqReportingCriteria ::=      SEQUENCE {
    eventCriteriaList                   IntraFreqEventCriteriaList
}

IntraFreqReportingQuantity ::=      SEQUENCE {
    activeSetReportingQuantities        CellReportingQuantities,
    monitoredSetReportingQuantities     CellReportingQuantities,
    unlistedSetReportingQuantities      CellReportingQuantities             OPTIONAL
}

IntraFreqReportingQuantityForRACH ::= SEQUENCE {
    sfn-SFN-ObsTimeDifference           SFN-SFN-ObsTimeDifference,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            intraFreqRepQuantityRACH-FDD        IntraFreqRepQuantityRACH-FDD
        },
        tdd                                 SEQUENCE {
            intraFreqRepQuantityRACH-TDD        IntraFreqRepQuantityRACH-TDD
        }
    }
}

IntraFreqRepQuantityRACH-FDD ::=    ENUMERATED {
                                        cpich-EcN0, cpich-RSCP,
                                        cpich-SIR, pathloss, noReport }

IntraFreqRepQuantityRACH-TDD ::=    ENUMERATED {
                                        timeslotISCP,
                                        primaryCCPCH-RSCP,
                                        noReport }

IntraFrequencyMeasurement ::=       SEQUENCE {
    intraFreqCellInfoList               IntraFreqCellInfoList               OPTIONAL,
    intraFreqMeasQuantity               IntraFreqMeasQuantity               OPTIONAL,
    intraFreqReportingQuantity          IntraFreqReportingQuantity          OPTIONAL,
    reportingCellStatus                 ReportingCellStatus                 OPTIONAL,
```

```
    measurementValidity                 MeasurementValidity                 OPTIONAL,
    reportCriteria                      IntraFreqReportCriteria
}

IODD ::=                            INTEGER (0..255)

IODE ::=                            INTEGER (0..255)

IP-Length ::=                       ENUMERATED {
                                        ipl5, ipl10 }

IP-Spacing ::=                      ENUMERATED {
                                        e5, e7, e10, e15, e20,
                                        e30, e40, e50 }

IS-2000SpecificMeasInfo ::=         ENUMERATED {
                                        frequency, timeslot, colourcode,
                                        outputpower, pn-Offset }

K-InterRAT ::=                      INTEGER (0..12)

LCS-Accuracy ::=                    BIT STRING (SIZE (7))

LCS-CipherParameters ::=            SEQUENCE {
    cipheringKeyFlag                    BIT STRING (SIZE (1)),
    cipheringSerialNumber               INTEGER (0..65535)
}

LCS-Error ::=                       SEQUENCE {
    errorReason                         LCS-ErrorCause,
    additionalAssistanceData            AdditionalAssistanceData
    -- The IE above is defined in GSM 09.31, the actual definition
    -- will have to be checked
}

LCS-ErrorCause ::=                  ENUMERATED {
                                        notEnoughOTDOA-Cells,
                                        notEnoughGPS-Satellites,
                                        assistanceDataMissing,
                                        methodNotSupported,
                                        undefinedError,
                                        requestDeniedByUser,
                                        notProcessedAndTimeout }

LCS-EventID ::=                     ENUMERATED {
                                        e7a, e7b, e7c }

LCS-EventParam ::=                  SEQUENCE {
    eventID                             LCS-EventID,
    reportingAmount                     ReportingAmount,
    reportFirstFix                      BOOLEAN,
    measurementInterval                 LCS-MeasurementInterval,
    eventSpecificInfo                   LCS-EventSpecificInfo
}

LCS-EventParamList ::=              SEQUENCE (SIZE (1..maxEventCount)) OF
                                        LCS-EventParam

LCS-EventSpecificInfo ::=           CHOICE {
    e7a                                 ThresholdPositionChange,
    e7b                                 ThresholdSFN-SFN-Change,
    e7c                                 ThresholdSFN-GPS-TOW
}

LCS-GPS-AcquisitionAssistance ::=   SEQUENCE {
    referenceTime                       CHOICE {
        utran-ReferenceTime                 UTRAN-ReferenceTime,
        gps-ReferenceTimeOnly               INTEGER (0..604700000)
    },
    satelliteInformationList            AcquisitionSatInfoList
}

LCS-GPS-Almanac ::=                 SEQUENCE {
    almanacSatInfoList                  AlmanacSatInfoList
}

LCS-GPS-AssistanceSIB ::=           SEQUENCE {
    lcs-CipherParameters                LCS-CipherParameters                OPTIONAL,
```

```
    referenceGPS-TOW                    ReferenceGPS-TOW,
    status                              DiffCorrectionStatus,
    btsClockDrift                       BTS-ClockDrift                  OPTIONAL,
    timeOffset                          LCS-TimeOffset                  OPTIONAL,
    iodd                                IODD                            OPTIONAL,
    dgps-InformationList                DGPS-InformationList            OPTIONAL
}

LCS-GPS-AssistanceData ::=          SEQUENCE {
    lcs-GPS-ReferenceTime               LCS-GPS-ReferenceTime           OPTIONAL,
    lcs-GPS-ReferenceLocation           LCS-GPS-ReferenceLocation       OPTIONAL,
    lcs-GPS-DGPS-Corrections            LCS-GPS-DGPS-Corrections        OPTIONAL,
    lcs-GPS-NavigationModel             LCS-GPS-NavigationModel         OPTIONAL,
    lcs-GPS-IonosphericModel            LCS-GPS-IonosphericModel        OPTIONAL,
    lcs-GPS-UTC-Model                   LCS-GPS-UTC-Model               OPTIONAL,
    lcs-GPS-Almanac                     LCS-GPS-Almanac                 OPTIONAL,
    lcs-GPS-AcquisitionAssistance       LCS-GPS-AcquisitionAssistance   OPTIONAL,
    lcs-GPS-Real-timeIntegrity          LCS-GPS-Real-timeIntegrity      OPTIONAL
}

LCS-GPS-DGPS-Corrections ::=        SEQUENCE {
    gps-TOW                             INTEGER (0..604799),
    statusHealth                        DiffCorrectionStatus,
    dgps-CorrectionSatInfoList          DGPS-CorrectionSatInfoList
}

LCS-GPS-IonosphericModel ::=        SEQUENCE {
    alfa0                               BIT STRING (SIZE (8)),
    alfa1                               BIT STRING (SIZE (8)),
    alfa2                               BIT STRING (SIZE (8)),
    alfa3                               BIT STRING (SIZE (8)),
    beta0                               BIT STRING (SIZE (8)),
    beta1                               BIT STRING (SIZE (8)),
    beta2                               BIT STRING (SIZE (8)),
    beta3                               BIT STRING (SIZE (8))
}

LCS-GPS-Measurement ::=             SEQUENCE {
    referenceSFN                        ReferenceSFN                    OPTIONAL,
    gps-TOW-1msec                       GPS-TOW-1msec,
    gps-TOW-HighResolution              GPS-TOW-HighResolution          OPTIONAL,
    gps-MeasurementParamList            GPS-MeasurementParamList
}

LCS-GPS-NavigationModel ::=         SEQUENCE {
    n-SAT                               INTEGER (1..16),
    navigationModelSatInfoList          NavigationModelSatInfoList
}

-- **TODO**, definition in 23.032
LCS-GPS-ReferenceLocation ::=       SEQUENCE {
}

LCS-GPS-Real-timeIntegrity ::=      SEQUENCE {
    badSatList                          BadSatList
}

LCS-GPS-ReferenceTime ::=           SEQUENCE {
    gps-Week                            INTEGER (0..1023),
    gps-TOW                             INTEGER (0..604700000000),
    sfn                                 INTEGER (0..4095),
    gps-TOW-AssistList                  GPS-TOW-AssistList              OPTIONAL
}

LCS-GPS-UTC-Model ::=               SEQUENCE {
    a0                                  BIT STRING (SIZE (32)),
    a1                                  BIT STRING (SIZE (24)),
    delta-t-LS                          BIT STRING (SIZE (8)),
    t-ot                                BIT STRING (SIZE (8)),
    wn-t                                BIT STRING (SIZE (8)),
    wn-lsf                              BIT STRING (SIZE (8)),
    dn                                  BIT STRING (SIZE (8)),
    delta-t-LSF                         BIT STRING (SIZE (8))
}

LCS-IPDL-Parameters ::=             SEQUENCE {
    ip-Spacing                          IP-Spacing,
    ip-Length                           IP-Length,
```

```
    ip-Offset                           INTEGER (0..9),
    seed                                INTEGER (0..63),
    burstModeParameters                 BurstModeParameters
}

LCS-MeasuredResults ::=             SEQUENCE {
    lcs-MultipleSets                    LCS-MultipleSets                    OPTIONAL,
    lcs-ReferenceCellIdentity           PrimaryCPICH-Info                   OPTIONAL,
    lcs-OTDOA-Measurement               LCS-OTDOA-Measurement               OPTIONAL,
    lcs-Position                        LCS-Position                        OPTIONAL,
    lcs-GPS-Measurement                 LCS-GPS-Measurement                 OPTIONAL,
    lcs-Error                           LCS-Error                           OPTIONAL
}

LCS-Measurement ::=                 SEQUENCE {
    lcs-ReportingQuantity               LCS-ReportingQuantity,
    reportCriteria                      LCS-ReportCriteria,
    lcs-OTDOA-AssistanceData            LCS-OTDOA-AssistanceData            OPTIONAL,
    lcs-GPS-AssistanceData              LCS-GPS-AssistanceData              OPTIONAL
}

LCS-MeasurementEventResults ::=     SEQUENCE {
    event7a                             LCS-Position,
    event7b                             LCS-OTDOA-Measurement,
    event7c                             LCS-GPS-Measurement
}

LCS-MeasurementInterval ::=         ENUMERATED {
                                        e5, e15, e60, e300,
                                        e900, e1800, e3600, e7200 }

LCS-MethodType ::=                  ENUMERATED {
                                        ue-Assisted,
                                        ue-Based,
                                        ue-BasedPreferred,
                                        ue-AssistedPreferred }

LCS-MultipleSets ::=                SEQUENCE {
    numberOfOTDOA-IPDL-GPS-Sets         INTEGER (2..3),
    numberOfReferenceCells              INTEGER (1..3),
    referenceCellRelation               ReferenceCellRelation
}

LCS-OTDOA-AssistanceData ::=        SEQUENCE {
    lcs-OTDOA-ReferenceCell             LCS-OTDOA-ReferenceCell             OPTIONAL,
    lcs-OTDOA-MeasurementAssistDataList LCS-OTDOA-MeasurementAssistDataList OPTIONAL,
    lcs-IPDL-Parameters                 LCS-IPDL-Parameters                 OPTIONAL
}

LCS-OTDOA-AssistanceSIB ::=         SEQUENCE {
    lcs-CipherParameters                LCS-CipherParameters                OPTIONAL,
    searchWindowSize                    OTDOA-SearchWindowSize,
    referenceCellPosition               ReferenceCellPosition,
    lcs-IPDL-Parameters                 LCS-IPDL-Parameters                 OPTIONAL,
    cellToMeasureInfoList               CellToMeasureInfoList
}

LCS-OTDOA-Measurement ::=           SEQUENCE {
    sfn                                 INTEGER (0..4095),
    -- Actual value = IE value * 0.25 + 876
    ue-Rx-Tx-TimeDifference             INTEGER (0..1184),
    qualityType                         QualityType,
    qualityChoice                       CHOICE {
        std-10                              ReferenceQuality10,
        std-50                              ReferenceQuality50,
        cpich-EcN0                          CPICH-Ec-N0-OTDOA,
        defaultQuality                      ReferenceQuality
    },
    neighborList                        NeighborList                        OPTIONAL
}

LCS-OTDOA-MeasurementAssistData ::= SEQUENCE {
    primaryCPICH-Info                   PrimaryCPICH-Info,
    frequencyInfo                       FrequencyInfo                       OPTIONAL,
    sfn-SFN-ObsTimeDifference           SFN-SFN-ObsTimeDifference1,
    fineSFN-SFN                         FineSFN-SFN                         OPTIONAL,
    searchWindowSize                    OTDOA-SearchWindowSize,
    relativeNorth                       INTEGER (-20000..20000)             OPTIONAL,
```

```
    relativeEast                        INTEGER (-20000..20000)             OPTIONAL,
    relativeAltitude                    INTEGER (-4000..4000)               OPTIONAL
}

LCS-OTDOA-MeasurementAssistDataList ::= SEQUENCE (SIZE (1..15)) OF
                                        LCS-OTDOA-MeasurementAssistData

LCS-OTDOA-ReferenceCell ::=         SEQUENCE {
    primaryCPICH-Info                   PrimaryCPICH-Info,
    frequencyInfo                       FrequencyInfo                       OPTIONAL,
    cellPosition                        ReferenceCellPosition               OPTIONAL
}

LCS-Position ::=                    SEQUENCE {
    referenceSFN                        ReferenceSFN,
    gps-TOW                             INTEGER (0..604700000000),
    positionEstimate                    PositionEstimate
}

LCS-ReportCriteria ::=              CHOICE {
    lcs-ReportingCriteria               LCS-ReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

LCS-ReportingCriteria ::=           SEQUENCE {
    eventParameterList                  LCS-EventParamList                  OPTIONAL
}

LCS-ReportingQuantity ::=           SEQUENCE {
    methodType                          LCS-MethodType,
    positioningMethod                   PositioningMethod,
    responseTime                        LCS-ResponseTime,
    accuracy                            LCS-Accuracy                        OPTIONAL,
    gps-TimingOfCellWanted              BOOLEAN,
    multipleSets                        BOOLEAN,
    environmentCharacterization         EnvironmentCharacterization         OPTIONAL
}

LCS-ResponseTime ::=                ENUMERATED {
                                        s1, s2, s4, s8, s16,
                                        s32, s64, s128 }

LCS-TimeOffset ::=                  INTEGER (0..4095)

MaxNumberOfReportingCells ::=       ENUMERATED {
                                        mandatoryCellsOnly,
                                        mandatoryCellsPlus1,
                                        mandatoryCellsPlus2,
                                        mandatoryCellsPlus3,
                                        mandatoryCellsPlus4,
                                        mandatoryCellsPlus5,
                                        mandatoryCellsPlus6 }

MaxReportedCellsOnRACH ::=          ENUMERATED {
                                        noReport,
                                        currentCell,
                                        currentAnd-1-BestNeighbour,
                                        currentAnd-2-BestNeighbour,
                                        currentAnd-3-BestNeighbour,
                                        currentAnd-4-BestNeighbour,
                                        currentAnd-5-BestNeighbour,
                                        currentAnd-6-BestNeighbour }

MeasuredResults ::=                 CHOICE {
    intraFreqMeasuredResultsList        IntraFreqMeasuredResultsList,
    interFreqMeasuredResultsList        InterFreqMeasuredResultsList,
    interSystemMeasuredResultsList      InterSystemMeasuredResultsList,
    trafficVolumeMeasuredResultsList    TrafficVolumeMeasuredResultsList,
    qualityMeasuredResults              QualityMeasuredResults,
    ue-InternalMeasuredResults          UE-InternalMeasuredResults,
    lcs-MeasuredResults                 LCS-MeasuredResults
}

MeasuredResultsList ::=             SEQUENCE (SIZE (1..maxAdditionalMeas)) OF
                                        MeasuredResults

MeasuredResultsOnRACH ::=           SEQUENCE {
```

```
    currentCell                         SEQUENCE {
        modeSpecificInfo                    CHOICE {
            fdd                                 SEQUENCE {
                measurementQuantity                 CHOICE {
                    cpich-Ec-N0                         CPICH-Ec-N0,
                    cpich-RSCP                          CPICH-RSCP,
                    cpich-SIR                           CPICH-SIR,
                    pathloss                            Pathloss
                }
            },
            tdd                                 SEQUENCE {
                timeslotISCP                        TimeslotISCP,
                primaryCCPCH-RSCP                   PrimaryCCPCH-RSCP
            }
        }
    },
    monitoredCells                      MonitoredCellRACH-List              OPTIONAL
}

MeasurementCommand ::=              CHOICE {
    setup                               MeasurementType,
    modify                              SEQUENCE {
        measurementType                     MeasurementType                 OPTIONAL
    },
    release                             NULL
}

MeasurementControlSysInfo ::=       SEQUENCE {
    intraFreqMeasurementSysInfo         IntraFreqMeasurementSysInfo         OPTIONAL,
    interFreqMeasurementSysInfo         InterFreqMeasurementSysInfo         OPTIONAL,
    interSystemMeasurementSysInfo       InterSystemMeasurementSysInfo       OPTIONAL,
    trafficVolumeMeasSysInfo            TrafficVolumeMeasSysInfo            OPTIONAL,
    ue-InternalMeasurementSysInfo       UE-InternalMeasurementSysInfo       OPTIONAL
}

-- **TODO**, not defined yet
MeasurementIdentityNumber ::=       SEQUENCE {
}

MeasurementQuantityGSM ::=          ENUMERATED {
                                        gsm-CarrierRSSI,
                                        pathloss }

MeasurementReportingMode ::=        SEQUENCE {
    measurementReportTransferMode       TransferMode,
    periodicalOrEventTrigger            PeriodicalOrEventTrigger
}

MeasurementType ::=                 CHOICE {
    intraFrequencyMeasurement           IntraFrequencyMeasurement,
    interFrequencyMeasurement           InterFrequencyMeasurement,
    interSystemMeasurement              InterSystemMeasurement,
    lcs-Measurement                     LCS-Measurement,
    trafficVolumeMeasurement            TrafficVolumeMeasurement,
    qualityMeasurement                  QualityMeasurement,
    ue-InternalMeasurement              UE-InternalMeasurement
}

MeasurementValidity ::=             SEQUENCE {
    resume-Release                      Resume-Release
}

MonitoredCellRACH-List ::=          SEQUENCE (SIZE(1..7)) OF
                                        MonitoredCellRACH-Result

MonitoredCellRACH-Result ::=        SEQUENCE {
    sfn-SFN-ObsTimeDifference           SFN-SFN-ObsTimeDifference           OPTIONAL,
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            primaryCPICH-Info                   PrimaryCPICH-Info,
            measurementQuantity                 CHOICE {
                cpich-Ec-N0                         CPICH-Ec-N0,
                cpich-RSCP                          CPICH-RSCP,
                cpich-SIR                           CPICH-SIR,
                pathloss                            Pathloss
            }                                                               OPTIONAL
        },
        tdd                                 SEQUENCE {
```

```
            primaryCCPCH-Info                   PrimaryCCPCH-Info,
            primaryCCPCH-RSCP                   PrimaryCCPCH-RSCP           OPTIONAL
        }
    }
}

MonitoredSetCellReport ::=          ENUMERATED {
                                        excludeAll,
                                        other }

MultipathIndicator ::=              ENUMERATED {
                                        nm,
                                        low,
                                        medium,
                                        high }

NavigationModelSatInfo ::=          SEQUENCE {
    satID                               INTEGER (0..63),
    satelliteStatus                     SatelliteStatus,
    compression                         CHOICE {
        uncompressed                        UncompressedNavModel,
        compressed                          CompressedNavModel
    }
}

NavigationModelSatInfoList ::=      SEQUENCE (SIZE (1..maxN-SAT)) OF
                                        NavigationModelSatInfo

Neighbor ::=                        SEQUENCE {
    neighborIdentity                    PrimaryCPICH-Info                   OPTIONAL,
    neignborQuantity                    NeighborQuantity,
    sfn-SFN-ObsTimeDifference2          SFN-SFN-ObsTimeDifference2
}

NeighborList ::=                    SEQUENCE (SIZE (1..15)) OF
                                        Neighbor

-- **TODO**, to be defined fully
NeighborQuantity ::=                SEQUENCE {
}

NewInterFreqCell ::=                SEQUENCE {
    interFreqCellID                     InterFreqCellID                     OPTIONAL,
    frequencyInfo                       FrequencyInfo                       OPTIONAL,
    cellInfo                            CellInfo
}

NewInterFreqCellList ::=            SEQUENCE (SIZE (1..maxInterCells)) OF
                                        NewInterFreqCell

NewInterFreqCellSI ::=              SEQUENCE {
    interFreqCellID                     InterFreqCellID                     OPTIONAL,
    frequencyInfo                       FrequencyInfo                       OPTIONAL,
    cellInfo                            CellInfoSI
}

NewInterFreqCellSI-List ::=         SEQUENCE (SIZE (1..maxInterCells)) OF
                                        NewInterFreqCellSI

NewInterSystemCell ::=              SEQUENCE {
    technologySpecificInfo              CHOICE {
        gsm                                 SEQUENCE {
            q-Offset                            Q-Offset                    OPTIONAL,
            hcs-NeighbouringCellInformation     HCS-NeighbouringCellInformation
                                                                            OPTIONAL,
            q-Min                               Q-Min,
            maxAllowedUL-TX-Power               MaxAllowedUL-TX-Power,
            bsic                                BSIC,
            bcch-ARFCN                          BCCH-ARFCN,
            gsm-OutputPower                     GSM-OutputPower             OPTIONAL
        },
        is-2000                             SEQUENCE {
            is-2000SpecificMeasInfo             IS-2000SpecificMeasInfo
        }
    }
}
```

```
NewInterSystemCellList ::=          SEQUENCE (SIZE (1..maxInterSysCells)) OF
                                        NewInterSystemCell

NewIntraFreqCell ::=                SEQUENCE {
    intraFreqCellID                     IntraFreqCellID                     OPTIONAL,
    cellInfo                            CellInfo
}

NewIntraFreqCellList ::=            SEQUENCE (SIZE (1..maxIntraCells)) OF
                                        NewIntraFreqCell

NewIntraFreqCellSI ::=              SEQUENCE {
    intraFreqCellID                     IntraFreqCellID                     OPTIONAL,
    cellInfo                            CellInfoSI
}

NewIntraFreqCellSI-List ::=         SEQUENCE (SIZE (1..maxIntraCells)) OF
                                        NewIntraFreqCell

NonUsedFreqParameter ::=            SEQUENCE {
    nonUsedFreqThreshold                Threshold,
    nonUsedFreqW                        W
}

NonUsedFreqParameterList ::=        SEQUENCE (SIZE (1..maxNonUsedFrequency)) OF
                                        NonUsedFreqParameter

ObservedTimeDifferenceToGSM ::=     INTEGER (0..4095)

OtherRAT-InSysInfo ::=              SEQUENCE {
    rat-Type                            RAT-Type,
    k-InterRAT                          K-InterRAT
}

OtherRAT-InSysInfoList ::=          SEQUENCE (SIZE (1..maxInterRAT)) OF
                                        OtherRAT-InSysInfo

OTDOA-SearchWindowSize ::=          ENUMERATED {
                                        c10, c20, c30, c40, c50,
                                        c60, c70, moreThan70 }

Pathloss ::=                        INTEGER (46..158)

PenaltyTime ::=                     CHOICE {
    notUsed                             NULL,
    pt10                                TemporaryOffset,
    pt20                                TemporaryOffset,
    pt30                                TemporaryOffset,
    pt40                                TemporaryOffset,
    pt50                                TemporaryOffset,
    pt60                                TemporaryOffset
}

PendingTimeAfterTrigger ::=         ENUMERATED {
                                        ptat0-25, ptat0-5, ptat1,
                                        ptat2, ptat4, ptat8, ptat16 }

PeriodicalOrEventTrigger ::=        ENUMERATED {
                                        periodical,
                                        eventTrigger }

PeriodicalReportingCriteria ::=     SEQUENCE {
    reportingAmount                     ReportingAmount                     OPTIONAL,
    reportingInterval                   ReportingIntervalLong               OPTIONAL
}

-- **TODO**, contents to be defined, source 23.032
PositionEstimate ::=                CHOICE {
    ellipsoidPoint                      SEQUENCE {},
    ellipsoidPointUncertCircle          SEQUENCE {},
    ellipsoidPointUncertEllipse         SEQUENCE {},
    ellipsoidPointAltitude              SEQUENCE {},
    ellipsoidPointAltitudeEllipse       SEQUENCE {}
}

PositioningMethod ::=               ENUMERATED {
                                        otdoa,
                                        gps,
```

```
                                        otdoaOrGPS }

PRC ::=                             INTEGER (-32767..32767)

-- **TODO**, not defined yet
PrimaryCCPCH-RSCP ::=               SEQUENCE {
}

Q-Accept-s-n ::=                    INTEGER (0..63)

Q-HCS ::=                           INTEGER (0..99)

Q-Offset ::=                        INTEGER (-50..50)

-- Actual value = IE value * 0.5
Q-OffsetS-N ::=                     INTEGER (-40..40)

-- **TODO**, not defined yet
Q-Min   ::=                         SEQUENCE {
}

Qmin-FDD ::=                        INTEGER (-20..0)

-- Actual value = IE value * 2 - 115
Qmin-TDD ::=                        INTEGER (0..45)

-- **TODO**, not defined yet
QualityEventResults ::=             SEQUENCE {
}

-- **TODO**, not defined yet
QualityMeasQuantity ::=             SEQUENCE {
}

QualityMeasuredResults ::=          SEQUENCE {
    blerMeasurementResultsList          BLER-MeasurementResultsList         OPTIONAL,
    dl-PhysicalChannelBER               DL-PhysicalChannelBER               OPTIONAL,
    sir                                 SIR                                 OPTIONAL
}

QualityMeasurement ::=              SEQUENCE {
    qualityMeasurementObject            QualityMeasurementObject            OPTIONAL,
    qualityMeasQuantity                 QualityMeasQuantity                 OPTIONAL,
    qualityReportingQuantity            QualityReportingQuantity            OPTIONAL,
    reportCriteria                      QualityReportCriteria
}

-- **TODO**, not defined yet
QualityMeasurementObject ::=        SEQUENCE {
}

QualityReportCriteria ::=           CHOICE {
    qualityReportingCriteria            QualityReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

-- **TODO**, not defined yet
QualityReportingCriteria ::=        SEQUENCE {
}

QualityReportingQuantity ::=        SEQUENCE {
    dl-TransChBLER                      BOOLEAN,
    bler-TransChIdList                  BLER-TransChIdList                  OPTIONAL,
    sir                                 BOOLEAN
}

QualityType ::=                     ENUMERATED {
                                        std-10, std-50, cpich-Ec-N0 }

RAT-Type ::=                        ENUMERATED {
                                        gsm, is2000, spare1, spare2,
                                        spare3, spare4, spare5, spare6,
                                        spare7, spare8, spare9, spare10,
                                        spare11, spare12, spare13, spare14 }

-- **TODO**, definition to be checked from 23.032
ReferenceCellPosition ::=           SEQUENCE {
```

```
}

ReferenceCellRelation ::=           ENUMERATED {
                                        first-12-second-3,
                                        first-13-second-2,
                                        first-1-second-23 }

ReferenceGPS-TOW ::=                INTEGER (0..604700000000)

ReferenceQuality ::=                ENUMERATED {
                                        m0-19, m20-39, m40-79,
                                        m80-159, m160-319, m320-639,
                                        m640-1319, m1320Plus }

-- Actual value = IE value * 10
ReferenceQuality10 ::=              INTEGER (1..32)

-- Actual value = IE value * 50
ReferenceQuality50 ::=              INTEGER (1..32)

ReferenceSFN ::=                    INTEGER (0..4095)

-- Actual value = IE value * 512
ReferenceTimeDifferenceToCell ::=   CHOICE {
    -- Actual value = IE value * 40
    accuracy40                          INTEGER (0..960),
    -- Actual value = IE value * 256
    accuracy256                         INTEGER (0..150),
    -- Actual value = IE value * 2560
    accuracy2560                        INTEGER (0..15)
}

RemovedInterFreqCell :: =           SEQUENCE {
    interFreqCellID                     InterFreqCellID
}

RemovedInterFreqCellList ::=        SEQUENCE (SIZE (1..maxInterCells)) OF
                                        RemovedInterFreqCell

RemovedInterSystemCell ::=          SEQUENCE {
    interSystemCellID                   InterSystemCellID
}

RemovedInterSystemCellList ::=      SEQUENCE (SIZE (1..maxInterSysCells)) OF
                                        RemovedInterSystemCell

RemovedIntraFreqCell ::=            SEQUENCE {
    intraFreqCellID                     IntraFreqCellID
}

RemovedIntraFreqCellList ::=        SEQUENCE (SIZE (1..maxIntraCells)) OF
                                        RemovedIntraFreqCell

ReplacementActivationThreshold ::=  ENUMERATED {
                                        notApplicable, t1, t2,
                                        t3, t4, t5, t6, t7 }

ReportDeactivationThreshold ::=     ENUMERATED {
                                        notApplicable, t1, t2,
                                        t3, t4, t5, t6, t7 }

ReportingAmount ::=                 ENUMERATED {
                                        ra1, ra2, ra4, ra8, ra16, ra32,
                                        ra64, ra-Infinity }

ReportingCellStatus ::=             SEQUENCE {
    maxNumberOfReportingCells           MaxNumberOfReportingCells,
    measurement                         CHOICE {
        intraFreq                           ReportingCellStatusIntraFreq,
        otherMeasurement                    NULL
    }
}

ReportingCellStatusIntraFreq ::=    SEQUENCE {
    activeSetCellReport                 ActiveSetCellReport,
    monitoredSetCellReport              MonitoredSetCellReport
}
```

```
ReportingInfoForCellDCH ::=         SEQUENCE {
    intraFreqReportingQuantity          IntraFreqReportingQuantity,
    reportCriteria                      CellDCH-ReportCriteria
}

ReportingInterval ::=               ENUMERATED {
                                        noPeriodicalreporting, ri0-25,
                                        ri0-5, ri1, ri2, ri4, ri8, ri16 }

ReportingIntervalLong ::=           ENUMERATED {
                                        ril0, ril0-25, ril0-5, ril1,
                                        ril2, ril3, ril4, ril6, ril8,
                                        ril12, ril16, ril20, ril24,
                                        ril28, ril32, ril64 }

-- Actual value = IE value * 0.5
ReportingRange ::=                  INTEGER (0..29)

Resume-Release ::=                  CHOICE {
    resume                              UE-State,
    release                             NULL
}

RL-AdditionInfo ::=                 SEQUENCE {
    primaryCPICH-Info                   PrimaryCPICH-Info
}

RL-AdditionInfoList ::=             SEQUENCE (SIZE(1..maxAddRLcount)) OF
                                        RL-AdditionInfo

RL-InformationLists ::=             SEQUENCE {
    rl-AdditionInfoList                 RL-AdditionInfoList                 OPTIONAL,
    rl-RemovalInfoList                  RL-RemovalInfoList                  OPTIONAL
}

RL-RemovalInfo ::=                  SEQUENCE {
    primaryCPICH-Info                   PrimaryCPICH-Info
}

RL-RemovalInfoList ::=              SEQUENCE (SIZE(1..maxDelRLcount)) OF
                                        RL-RemovalInfo

RLC-BuffersPayload ::=              ENUMERATED {
                                        pl0, pl4, pl8, pl16, pl32, pl64, pl128,
                                        pl256, pl512, pl1024, pl2k, pl4k,
                                        pl8k, pl16k, pl32k, pl64k, pl128k,
                                        pl256k, pl512k, pl1024k }

RRC ::=                             INTEGER (-127..127)

-- **TODO**, not defined yet
RSCP ::=                            SEQUENCE {
}

SatelliteStatus ::=                 ENUMERATED {
                                        ns-NN-U,
                                        es-SN,
                                        es-NN-U,
                                        es-NN-C }

SatID ::=                           INTEGER (0..31)

ScaleFactor ::=                     ENUMERATED {
                                        prc0-02-rrc0-002,
                                        prc0-32-rrc0-032 }

SFN-SFN-ObsTimeDifference ::=       CHOICE {
    type1                               SFN-SFN-ObsTimeDifference1,
    -- Actual value for type2 = IE value * 0.25
    type2                               SFN-SFN-ObsTimeDifference2
}

SFN-SFN-ObsTimeDifference1 ::=      INTEGER (0..9830399)

SFN-SFN-ObsTimeDifference2 ::=      INTEGER (-5119..5120)

SFN-SFN-OTD-Type ::=                ENUMERATED {
                                        noReport,
```

```
                                    type1,
                                    type2 }

SignallingOption ::=                CHOICE {
    alternative1                        SEQUENCE {
        q-OffsetS-N                     Q-OffsetS-N                         OPTIONAL
    },
    alternative2                    NULL
}

SIR ::=                             INTEGER (-10..20)

TemporaryOffset ::=                 ENUMERATED {
                                        to10, to20, to30, to40, to50,
                                        to60, to70, infinite }

-- **TODO**, not defined yet
Threshold ::=                       SEQUENCE {
}

ThresholdPositionChange ::=         ENUMERATED {
                                        pc10, pc20, pc30, pc40, pc50,
                                        pc100, pc200, pc300, pc500,
                                        pc1000, pc2000, pc5000, pc10000,
                                        pc20000, pc50000, pc100000 }

ThresholdSFN-GPS-TOW ::=            ENUMERATED {
                                        ms1, ms2, ms3, ms5, ms10,
                                        ms20, ms50, ms100 }

ThresholdSFN-SFN-Change ::=         ENUMERATED {
                                        c0-25, c0-5, c1, c2, c3, c4, c5,
                                        c10, c20, c50, c100, c200, c500,
                                        c1000, c2000, c5000 }

-- **TODO**, not defined yet
TimeslotISCP ::=                    SEQUENCE {
}

TimeslotListWithISCP ::=            SEQUENCE (SIZE (1..14)) OF
                                        TimeslotWithISCP

TimeslotWithISCP ::=                SEQUENCE {
    timeslot                            Timeslot,
    timeslotISCP                        TimeslotISCP
}

TimeToTrigger ::=                   ENUMERATED {
                                        ttt0, ttt10, ttt20, ttt40, ttt60,
                                        ttt80, ttt100, ttt120, ttt160,
                                        ttt200, ttt240, tt320, ttt640,
                                        ttt1280, ttt2560, ttt5000 }

TrafficVolumeEventParam ::=         SEQUENCE {
    eventID                             TrafficVolumeEventType,
    reportingThreshold                  TrafficVolumeThreshold
}

TrafficVolumeEventResults ::=       SEQUENCE {
    transportChannelCausingEvent        TransportChannelIdentity,
    trafficVolumeEventIdentity          EventIDTrafficVolume
}

TrafficVolumeEventType ::=          ENUMERATED {
                                        e4a,
                                        e4b }

TrafficVolumeMeasObject ::=         SEQUENCE {
    targetTransportChannelID            TransportChannelIdentity
}

TrafficVolumeMeasObjectList ::=     SEQUENCE (SIZE (1..maxTrCHcount)) OF
                                        TrafficVolumeMeasObject

TrafficVolumeMeasQuantity ::=       ENUMERATED {
                                        rlc-BufferPayload,
                                        averageRLC-BufferPayload,
                                        varianceOfRLC-BufferPayload }
```

```
TrafficVolumeMeasSysInfo ::=        SEQUENCE {
    trafficVolumeMeasurementID          MeasurementIdentityNumber           OPTIONAL,
    trafficVolumeMeasObjectList         TrafficVolumeMeasObjectList         OPTIONAL,
    trafficVolumeMeasQuantity           TrafficVolumeMeasQuantity           OPTIONAL
}

TrafficVolumeMeasuredResults ::=    SEQUENCE {
    rb-Identity                         RB-Identity,
    rlc-BuffersPayload                  RLC-BuffersPayload                  OPTIONAL,
    averageRLC-BufferPayload            AverageRLC-BufferPayload            OPTIONAL,
    varianceOfRLC-BufferPayload         VarianceOfRLC-BufferPayload         OPTIONAL
}

TrafficVolumeMeasuredResultsList ::= SEQUENCE (SIZE (1..maxTraf)) OF
                                    TrafficVolumeMeasuredResults

TrafficVolumeMeasurement ::=        SEQUENCE {
    TrafficVolumeMeasurementObjectList  TrafficVolumeMeasurementObjectList  OPTIONAL,
    trafficVolumeMeasQuantity           TrafficVolumeMeasQuantity           OPTIONAL,
    trafficVolumeReportingQuantity      TrafficVolumeReportingQuantity      OPTIONAL,
    measurementValidity                 MeasurementValidity                 OPTIONAL,
    reportCriteria                      TrafficVolumeReportCriteria
}

TrafficVolumeMeasurementObject ::= SEQUENCE {
    targetTransportChannelID            TransportChannelIdentity
}

TrafficVolumeMeasurementObjectList ::= SEQUENCE (SIZE (1..maxTrCHcount)) OF
                                        TrafficVolumeMeasurementObject

TrafficVolumeReportCriteria ::=     CHOICE {
    trafficVolumeReportingCriteria      TrafficVolumeReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

TrafficVolumeReportingCriteria ::= SEQUENCE {
    transChCriteriaList                 TransChCriteriaList                 OPTIONAL,
    timeToTrigger                       TimeToTrigger                       OPTIONAL,
    pendingTimeAfterTrigger             PendingTimeAfterTrigger             OPTIONAL,
    tx-InterruptionAfterTrigger         TX-InterruptionAfterTrigger         OPTIONAL,
    reportingAmount                     ReportingAmount                     OPTIONAL,
    reportingInterval                   ReportingInterval                   OPTIONAL
}

TrafficVolumeReportingQuantity ::= SEQUENCE {
    rlc-RB-BufferPayload                BOOLEAN,
    rlc-RB-BufferPayloadAverage         BOOLEAN,
    rlc-RB-BufferPayloadVariance        BOOLEAN
}

TrafficVolumeThreshold ::=          ENUMERATED {
                                        th8, th16, th32, th64, th128,
                                        th256, th512, th1024, th1536,
                                        th2048, th3072, th4096, th6144,
                                        th8192 }

TransChCriteria ::=                 SEQUENCE {
    transportChannelID                  TransportChannelIdentity,
    eventSpecificParameters             SEQUENCE (SIZE (1..2)) OF
                                            TrafficVolumeEventParam         OPTIONAL
}

TransChCriteriaList ::  =           SEQUENCE (SIZE (1..maxTrCHcount)) OF
                                        TransChCriteria

TransferMode ::=                    ENUMERATED {
                                        acknowledgedModeRLC,
                                        unacknowledgedModeRLC }

TransmittedPowerThreshold ::=       INTEGER (-50..33)

TriggeringCondition ::=             ENUMERATED {
                                        activeSetCellsOnly,
                                        monitoredCellsOnly,
                                        activeSetAndMonitoredCells }
```

```
TX-InterruptionAfterTrigger ::=     ENUMERATED {
                                        txiat0-25, txiat0-5, txiat1,
                                        txiat2, txiat4, txiat8, txiat16 }

UDRE ::=                            ENUMERATED {
                                        lessThan1,
                                        between1-and-4,
                                        between4-and-8,
                                        over8 }

UE-6AB-Event ::=                    SEQUENCE {
    timeToTrigger                       TimeToTrigger,
    transmittedPowerThreshold           TransmittedPowerThreshold
}

UE-6FG-Event ::=                    SEQUENCE {
    timeToTrigger                       TimeToTrigger,
    ue-RX-TX-TimeDifferenceThreshold    UE-RX-TX-TimeDifferenceThreshold
}

UE-AutonomousUpdateMode ::=         CHOICE {
    on                                  NULL,
    onWithNoReporting                   NULL,
    off                                 RL-InformationLists
}


UE-InternalEventParam ::=           CHOICE {
    event6a                             UE-6AB-Event,
    event6b                             UE-6AB-Event,
    event6c                             TimeToTrigger,
    event6d                             TimeToTrigger,
    event6e                             TimeToTrigger,
    event6f                             UE-6FG-Event,
    event6g                             UE-6FG-Event
}

UE-InternalEventParamList ::=       SEQUENCE (SIZE (1..maxEventCount)) OF
                                        UE-InternalEventParam

UE-InternalEventResults ::=         CHOICE {
    event6a                             NULL,
    event6b                             NULL,
    event6c                             NULL,
    event6d                             NULL,
    event6e                             NULL,
    event6f                             PrimaryCPICH-Info,
    event6g                             PrimaryCPICH-Info
}

UE-InternalMeasQuantity ::=         SEQUENCE {
    measurementQuantity                 UE-MeasurementQuantity,
    filterCoefficient                   FilterCoefficient
}

UE-InternalMeasuredResults ::=      SEQUENCE {
    modeSpecificInfo                    CHOICE {
        fdd                                 SEQUENCE {
            ue-TransmittedPowerFDD              UE-TransmittedPowerFDD      OPTIONAL,
            ue-RX-TX-ReportEntryList            UE-RX-TX-ReportEntryList    OPTIONAL
        },
        tdd                                 SEQUENCE {
            ue-TransmittedPowerTDD-List         UE-TransmittedPowerTDD-List OPTIONAL
        }
    }
}

UE-InternalMeasurement ::=          SEQUENCE {
    ue-InternalMeasQuantity             UE-InternalMeasQuantity             OPTIONAL,
    ue-InternalReportingQuantity        UE-InternalReportingQuantity        OPTIONAL,
    reportCriteria                      UE-InternalReportCriteria
}

UE-InternalMeasurementSysInfo ::=   SEQUENCE {
    ue-InternalMeasurementID            MeasurementIdentityNumber           OPTIONAL,
    ue-InternalMeasQuantity             UE-InternalMeasQuantity
}
```

```
UE-InternalReportCriteria ::=       CHOICE {
    ue-InternalReportingCriteria        UE-InternalReportingCriteria,
    periodicalReportingCriteria         PeriodicalReportingCriteria,
    noReporting                         NULL
}

UE-InternalReportingCriteria ::=    SEQUENCE {
    ue-InternalEventParamList           UE-InternalEventParamList           OPTIONAL
}

UE-InternalReportingQuantity ::=    SEQUENCE {
    ue-TransmittedPower                 BOOLEAN,
    ue-RX-TX-TimeDifferece              BOOLEAN,
    ue-Position                         BOOLEAN
}

UE-MeasurementQuantity ::=          ENUMERATED {
                                        ue-TransmittedPower,
                                        utra-Carrier-RSSI,
                                        ue-RX-TX-TimeDifference }

UE-RX-TX-ReportEntry ::=            SEQUENCE {
    primaryCPICH-Info                   PrimaryCPICH-Info,
    ue-RX-TX-TimeDifference             UE-RX-TX-TimeDifference
}

UE-RX-TX-ReportEntryList ::=        SEQUENCE (SIZE (1..maxUsedRLcount)) OF
                                        UE-RX-TX-ReportEntry

UE-RX-TX-TimeDifference ::=         INTEGER (876..1172)

UE-RX-TX-TimeDifferenceThreshold ::= INTEGER (769..1280)

UE-State ::=                        ENUMERATED {
                                        cell-DCH, all-But-Cell-DCH, all-States }

UE-TransmittedPowerFDD ::=          INTEGER (-50..33)

-- **TODO**, not defined yet
UE-TransmittedPowerTDD ::=          SEQUENCE {
}

UE-TransmittedPowerTDD-List ::=     SEQUENCE (SIZE (1..maxUsedUplTScount)) OF
                                        UE-TransmittedPowerTDD

UncompressedNavModel ::=            SEQUENCE {
    iode                                BIT STRING (SIZE (8)),
    t-oe                                BIT STRING (SIZE (16)),
    c-rc                                BIT STRING (SIZE (16)),
    c-rs                                BIT STRING (SIZE (16)),
    c-ic                                BIT STRING (SIZE (16)),
    c-is                                BIT STRING (SIZE (16)),
    c-uc                                BIT STRING (SIZE (16)),
    c-us                                BIT STRING (SIZE (16)),
    e                                   BIT STRING (SIZE (32)),
    m0                                  BIT STRING (SIZE (32)),
    a-Sqrt                              BIT STRING (SIZE (32)),
    delta-n                             BIT STRING (SIZE (16)),
    omega0                              BIT STRING (SIZE (32)),
    omegaDot                            BIT STRING (SIZE (24)),
    i0                                  BIT STRING (SIZE (32)),
    iDot                                BIT STRING (SIZE (14)),
    omega                               BIT STRING (SIZE (32)),
    t-oc                                BIT STRING (SIZE (16)),
    af0                                 BIT STRING (SIZE (22)),
    af1                                 BIT STRING (SIZE (16)),
    af2                                 BIT STRING (SIZE (8))
}

UTRA-CarrierRSSI ::=                INTEGER (-95..-30)

UTRAN-ReferenceTime ::=             SEQUENCE {
    gps-TOW                             INTEGER (0..604700000000),
    sfn                                 INTEGER (0..4095)
}

VarianceOfRLC-BufferPayload ::=     ENUMERATED {
```

```
                                        plv0, plv4, plv8, plv16, plv32, plv64,
                                        plv128, plv256, plv512, plv1024,
                                        plv2k, plv4k, plv8k, plv16k }

-- Actual value = IE value * 0.1
W ::=                                INTEGER (0..20)

END
```