

# ETSI TS 125 323 V6.3.0 (2005-09)

---

*Technical Specification*

## **Universal Mobile Telecommunications System (UMTS); Packet Data Convergence Protocol (PDCP) specification (3GPP TS 25.323 version 6.3.0 Release 6)**



---

**Reference**

RTS/TSGR-0225323v630

---

**Keywords**

UMTS

**ETSI**

---

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

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

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

---

**Important notice**

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

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

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**Copyright Notification**

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

© European Telecommunications Standards Institute 2005.  
All rights reserved.

**DECT**<sup>TM</sup>, **PLUGTESTS**<sup>TM</sup> and **UMTS**<sup>TM</sup> are Trade Marks of ETSI registered for the benefit of its Members.  
**TIPHON**<sup>TM</sup> and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.  
**3GPP**<sup>TM</sup> is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

---

## Intellectual Property Rights

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

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

---

## Foreword

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

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

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under <http://webapp.etsi.org/key/queryform.asp>.

# Contents

Intellectual Property Rights .....	2
Foreword.....	2
Foreword.....	5
1 Scope .....	6
2 References .....	6
3 Definitions and Abbreviations.....	6
3.1 Definitions .....	6
3.2 Abbreviations .....	7
4 General .....	8
4.1 Objective .....	8
4.2 Overview on sublayer architecture .....	8
5 Functions .....	8
5.1 Header Compression .....	9
5.1.1 Mapping of PID values .....	9
5.1.2 IP Header Compression (RFC 2507) .....	10
5.1.2.1 Context identifiers .....	10
5.1.2.2 Mapping of PID values for RFC 2507 .....	10
5.1.2.3 Management of Full Header transmission.....	10
5.1.3 Robust Header Compression (RFC 3095) .....	10
5.1.3.1 Context identifiers .....	11
5.1.3.2 Mapping of PID values for RFC 3095 with CIDs in PDCP PDU Header .....	11
5.1.3.3 Mapping of PID values for RFC 3095 with CIDs within ROHC packet format .....	11
5.1.3.4 RFC 3095 Segmentation .....	11
5.1.3.5 Protocol Parameters .....	12
5.1.3.6 Configuration by RRC .....	12
5.2 Void.....	13
5.3 Data Transfer.....	13
5.3.1 Data transfer over acknowledged mode RLC .....	13
5.3.2 Data transfer over unacknowledged and transparent mode RLC.....	13
5.4 SRNS Relocation.....	14
5.4.1 Lossless SRNS Relocation.....	14
5.4.1.1 Void.....	14
5.4.1.2 Void.....	15
5.4.1.3 Void.....	15
5.4.2 Context relocation.....	15
5.5 Lossless DL RLC PDU size change .....	17
5.6 General procedures.....	17
5.6.1.1 PDCP Sequence Numbering .....	17
5.6.1.2 PDCP Sequence Number synchronization .....	18
5.6.1.3 Sequence Number and Data Forwarding.....	18
5.7 Header Compression and Decompression for MBMS .....	19
5.7.1 Cell change inside the same cell group .....	19
5.7.2 Cell change between cell groups.....	19
6 Services .....	19
6.1 Services provided to upper layers.....	19
6.2 Services expected from RLC layer.....	19
7 Elements for layer-to-layer communication .....	20
7.1 Primitives between PDCP and upper layers .....	20
8 Elements for peer-to-peer communication .....	21
8.1 Protocol data units .....	21
8.2 Formats.....	21

8.2.1	PDCP-No-Header PDU .....	22
8.2.2	PDCP Data PDU .....	22
8.2.3	PDCP SeqNum PDU .....	22
8.3	Parameters .....	22
8.3.1	PDU Type .....	23
8.3.2	PID .....	23
8.3.3	Data .....	23
8.3.4	Sequence number .....	23
9	Handling of unknown, unforeseen and erroneous protocol data .....	23
9.1	Invalid PDU type .....	23
9.2	Invalid PID value .....	24
<b>Annex A (normative): ROHC performance testing .....</b>		<b>25</b>
A.1	Introduction .....	25
A.2	Test outline .....	25
A.2.1	Test 1a - Base test of ROHC RTP O-mode compressor .....	25
A.2.1.1	Test purpose .....	25
A.2.1.2	Sequence details .....	25
A.2.1.3	Test requirement .....	26
A.2.2	Test 1b - Base test of ROHC RTP R-mode compressor .....	26
A.2.2.1	Test purpose .....	26
A.2.2.2	Sequence details .....	26
A.2.2.3	Test requirement .....	27
A.2.3	Test 2a - TTL / Hop-Limit variations in O-mode .....	27
A.2.3.1	Test purpose .....	27
A.2.3.2	Sequence details .....	27
A.2.3.3	Test requirement .....	27
A.2.4	Test 2b - TTL / Hop-Limit variations in R-mode .....	27
A.2.4.1	Test purpose .....	27
A.2.4.2	Sequence details .....	28
A.2.4.3	Test requirement .....	28
A.2.5	Test 3a - Re-establishment TS after DTX in O-mode .....	28
A.2.5.1	Test purpose .....	28
A.2.5.2	Sequence details .....	28
A.2.5.3	Test requirement .....	28
A.2.6	Test 3b - Re-establishment TS after DTX in R-mode .....	28
A.2.6.1	Test purpose .....	28
A.2.6.2	Sequence details .....	28
A.2.6.3	Test requirement .....	29
A.2.7	Test 4a - Compressor response to single lost packets in O-mode .....	29
A.2.7.1	Test purpose .....	29
A.2.7.2	Sequence details .....	29
A.2.7.3	Test requirement .....	29
A.2.8	Test 4b - Compressor response to single lost packets in R-mode .....	29
A.2.8.1	Test purpose .....	29
A.2.8.2	Sequence details .....	29
A.2.8.3	Test requirement .....	29
A.2.9	Test 5a - Compressor response to several packet losses in O-mode .....	29
A.2.9.1	Test purpose .....	29
A.2.9.2	Sequence details .....	30
A.2.9.3	Test requirement .....	30
A.2.10	Test 5b - Compressor response to several packet losses in R-mode .....	30
A.2.10.1	Test purpose .....	30
A.2.10.2	Sequence details .....	30
A.2.10.3	Test requirement .....	30
A.3	Test packet structures .....	30
<b>Annex B (informative): Change history .....</b>		<b>33</b>
History .....		34

---

# Foreword

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

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

Version x.y.z

where:

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

---

# 1 Scope

The present document provides the description of the Packet Data Convergence Protocol (PDCP).

---

## 2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

- [1] 3GPP TS 23.060: "General Packet Radio Service (GPRS); Service description; Stage 2".
  - [2] 3GPP TS 25.331: "Radio Resource Control (RRC); protocol specification".
  - [3] 3GPP TS 25.301: "Radio Interface Protocol Architecture".
  - [4] 3GPP TS 25.303: "Interlayer Procedures in Connected Mode".
  - [5] 3GPP TS 25.322: "RLC Protocol Specification".
  - [6] IETF RFC 2507: "IP Header Compression".
  - [7] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".
  - [8] IETF RFC 3095: "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed".
  - [9] IETF RFC 3096: "Requirements for robust IP/UDP/RTP header compression".
- 

## 3 Definitions and Abbreviations

### 3.1 Definitions

For the purposes of the present document, the terms and definitions given in [7] and the following apply.

<i>N-context</i>	Refers collectively to both <i>N-context-C</i> and <i>N-context-D</i> .
<i>N-context*</i>	Refers collectively to both <i>N-context-C*</i> and <i>N-context-D*</i> .
<i>N-context-C</i>	The compression context for downlink in SRNC at any given point of time.
<i>N-context-C*</i>	The frozen snapshot of the compression context for downlink taken by SRNC.
<i>N-context-C-static*</i>	The frozen snapshot of the static part of the compression context for downlink taken by SRNC.
<i>N-context-D</i>	The decompression context for uplink in SRNC at any given point of time.
<i>N-context-D*</i>	The frozen snapshot of the decompression context for uplink taken by SRNC.
<i>N-context-D-static*</i>	The frozen snapshot of the static part of the decompression context for uplink taken by SRNC.
<i>M-context</i>	Refers collectively to both <i>M-context-C</i> and <i>M-context-D</i> .
<i>M-context*</i>	Refers collectively to both <i>M-context-C*</i> and <i>M-context-D*</i> .
<i>M-context-C</i>	The compression context for uplink in UE at any given point of time.
<i>M-context-C*</i>	The frozen snapshot of the compression context for uplink taken by UE.

<i>M-context-C-static*</i>	The frozen snapshot of the static part of the compression context for uplink taken by UE.
<i>M-context-D</i>	The decompression context for downlink in UE at any given point of time.
<i>M-context-D*</i>	The frozen snapshot of the decompression context for downlink taken by UE.
<i>M-context-D-static*</i>	The frozen snapshot of the static part of the decompression context for downlink taken by UE.
M-HC	Entity located in the mobile terminal that performs header compression for uplink (i.e. UE PDCP)
M-HCD	Refers collectively to both <i>M-HC</i> and <i>M-HD</i> .
M-HD	Entity located in the mobile terminal that performs header decompression for downlink (i.e. UE PDCP)
N-HC	Entity located in the network that performs header compression for downlink (i.e. RNC PDCP)
N-HCD	Refers collectively to N-HC and N-HD
N-HD	Entity located in the network that performs header decompression for uplink (i.e. RNC PDCP)

## 3.2 Abbreviations

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

AS	Access Stratum
CID	Context Identifier
C-SAP	Control Service Access Point
HC	Header Compression
IETF	Internet Engineering Task Force
IP	Internet Protocol
L2	Layer 2 (data link layer)
L3	Layer 3 (network layer)
MBMS	Multimedia Broadcast Multicast Service
M-HC	Mobile Header Compressor
M-HCD	Mobile Header Compressor/Decompressor
M-HD	Mobile Header Decompressor
NAS	Non Access Stratum
N-HC	Network Header Compressor
N-HCD	Network Header Compressor/Decompressor
N-HD	Network Header Decompressor
PDCP	Packet Data Convergence Protocol
PDU	Protocol Data Unit
PID	Packet Identifier
PPP	Point-to-Point Protocol
p-t-p	Point-to-Point
p-t-m	Point-to-Multipoint
RB	Radio Bearer
RFC	Request For Comments
RLC	Radio Link Control
RNC	Radio Network Controller
ROHC	RObust Header Compression
RTP	Real Time Protocol
SDU	Service Data Unit
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network



## 4 General

### 4.1 Objective

The present document describes the functionality of the PDCP.

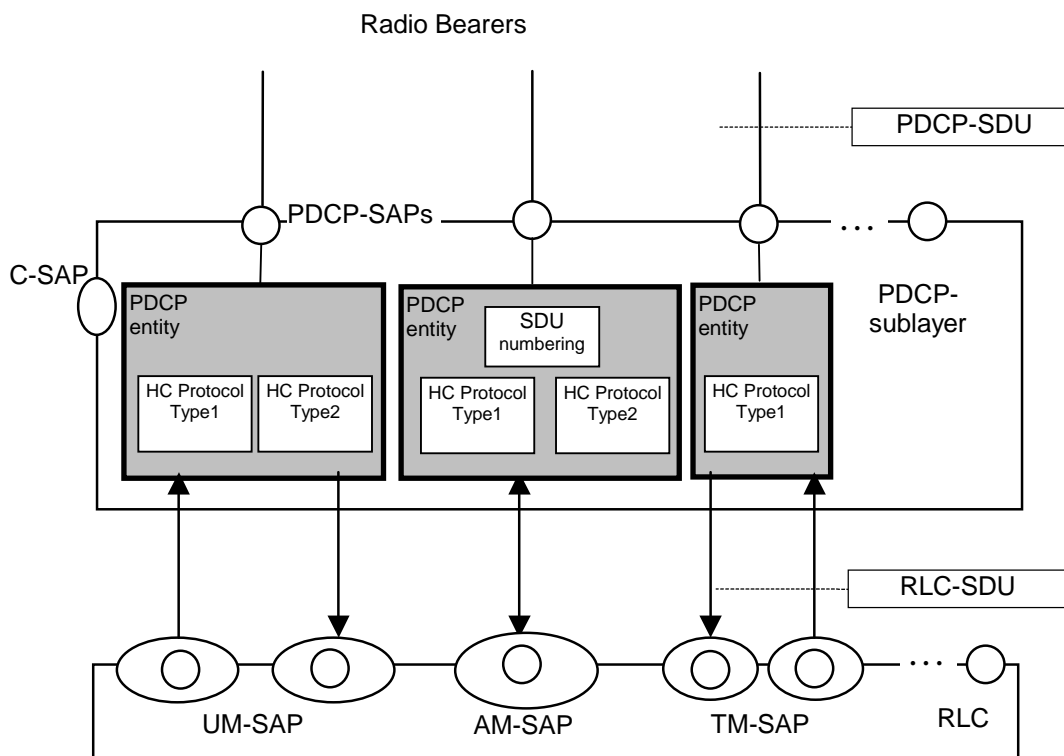
### 4.2 Overview on sublayer architecture

Figure 1 shows the model of the PDCP within the radio interface protocol architecture. The radio interface protocol architecture is defined in [3]. The PDCP sublayer is defined for the PS domain only.

Every PS domain RAB is associated with one RB, which in turn is associated with one PDCP entity. Each PDCP entity is associated with one or two (one for each direction) RLC entities depending on the RB characteristic (i.e.uni-directional or bi-directional) and RLC mode. The PDCP entities are located in the PDCP sublayer.

Every PDCP entity uses zero, one or several different header compression protocol types. Several PDCP entities may be defined for a UE with each using the same or different protocol type. In this version of the specification, only two header compression protocol types, RFC 2507 [6] and RFC 3095 [8], are supported.

The PDCP sublayer is configured by upper layer [2] through the PDCP-C-SAP.



**Figure 1: PDCP structure**

Figure 1 represents one possible structure for the PDCP sublayer and should not restrict implementation. A PDCP entity is mapped to either one AM RLC entity or one or two UM or TM RLC entities. When a PDCP entity is mapped to two UM or TM RLC entities each RLC entity is used for a different direction.

## 5 Functions

PDCP provides its services to the NAS at the UE or the relay at the Radio Network Controller (RNC).

The Packet Data Convergence Protocol shall perform the following functions:

- header compression and decompression of IP data streams (e.g., TCP/IP and RTP/UDP/IP headers for IPv4 and IPv6) at the transmitting and receiving entity, respectively.
- transfer of user data. This function is used for conveyance of data between users of PDCP services.
- maintenance of PDCP sequence numbers for radio bearers that are configured to support lossless SRNS Relocation or lossless DL RLC PDU size change.

PDCP uses the services provided by the Radio Link Control (RLC) sublayer.

## 5.1 Header Compression

The header compression protocol is specific to the particular network layer, transport layer or upper layer protocol combinations e.g. TCP/IP and RTP/UDP/IP. The network layer protocol type, e.g. IP or PPP, is indicated during PDP context activation as defined in [1]. The header compression protocols and their parameters are configured by upper layers for each PDCP entity. Compressor and decompressor initiated signalling between peer PDCP entities, during operation, is accomplished through in-band signalling.

### 5.1.1 Mapping of PID values

Depending on the configuration by upper layers (i.e. PDCP PDU type to be used and header compressor protocol), the PDCP sublayer shall be able to:

- identify different types of header compression protocols;
- if RFC2507:
  - distinguish different header compression protocol packet types within a header compression protocol.
- if RFC3095:
  - distinguish different contexts for a header compression protocol.

The above requirements are realised by utilising the PID field in the PDCP PDU.

The mapping of the PID values shall follow the general rules listed below:

- PID values shall be mapped to the different packet types independently at each PDCP entity;
- PID value "0" shall indicate "no compression". PID value "0" shall be used in a PDCP PDU containing in its Data field a PDCP SDU that is unchanged by the Sender and that shall not be decompressed by the Receiver.;
- PID values are mapped in ascending order, starting from 1, for every configured header compression protocol, in the order of configuration by upper layer. The first available PID value is assigned to the first packet type of the header compression protocol as defined in the specification for this header compression protocol. PID values are mapped for all the specified packet types defined for the header compression protocol and in the order defined in subclause 5.1.2.2, 5.1.3.1 and 5.1.3.3 for the respective header compression protocol;
- PID values are re-mapped for the PDCP entity after any reconfiguration of the header compression protocols for that entity.

The following table illustrates an example of the PID value mapping to the packet types when five arbitrary header compression methods are configured for one PDCP entity: RFC 2507[6], Methods A and B, RFC 3095 [8] and Method C. Method A, Method B and Method C are imaginary header compression protocols introduced for the purpose of illustration.

**Table 1: Example of the PID value mapping table**

PID Value	Optimisation method	Packet type
0	No header compression	-
1	RFC 2507	Full header
2	RFC 2507	Compressed TCP
3	RFC 2507	Compressed TCP nondelta
4	RFC 2507	Compressed non TCP
5	RFC 2507	Context state
6	Method A	Packet Type 1 of Method A
7	Method A	Packet Type 2 of Method A
8	Method B	Packet Type 1 of Method B
9	Method B	Packet Type 2 of Method B
10	RFC 3095	CID=0
11	RFC 3095	CID=1
12	RFC 3095	CID=2
13	Method C	Packet Type 1 of Method C
14	Method C	Packet Type 2 of Method C
15...31	Unassigned value	-

## 5.1.2 IP Header Compression (RFC 2507)

The detailed operation of the RFC 2507 header compression protocol is specified in IETF RFC 2507 [6]. The mechanisms related to error recovery and packet reordering are also described in RFC 2507. These mechanisms shall be included in the functionality of the header compression supported by PDCP. The implementation of the RFC 2507 header compression functionality is not covered in this specification and is left to the implementation.

### 5.1.2.1 Context identifiers

Context identifiers for RFC 2507 shall only be included in the RFC 2507 packet types format, as defined in [6].

### 5.1.2.2 Mapping of PID values for RFC 2507

PID values shall be mapped to the RFC 2507 header compression packet types in the order presented in Table 2 below where "n" is the number of PID values already mapped to other protocol packet types.

**Table 2: Mapping of PID values for RFC 2507 header compression protocol**

PID value	Optimisation method	Packet type
n+1	RFC 2507	Full header
n+2	RFC 2507	Compressed TCP
n+3	RFC 2507	Compressed TCP non-delta
n+4	RFC 2507	Compressed non-TCP
n+5	RFC 2507	Context state

### 5.1.2.3 Management of Full Header transmission

Transmission of a full header packet may be controlled by the lower layer information.

For a TCP stream, if the PDCP receives from lower layer the information of failed transmission of a single packet, the PDCP may send the next packet as a full header.

For a non-TCP stream, if the PDCP receives from lower layer the information of successful transmission of a full header packet, the PDCP may stop sending a full header packet that contains the same full header as the previously transmitted one.

## 5.1.3 Robust Header Compression (RFC 3095)

The detailed operation of the, "RObust Header Compression (ROHC)" protocol is specified in IETF RFC 3095 [8].

### 5.1.3.1 Context identifiers

The context of the RFC 3095 protocol is defined in [8]. RFC 3095 can be configured to support one or several contexts. Each context is identified by a value known as the context identifier (CID). If CIDs are to be used, then the CID shall be either:

- included in the PDCP header; or
- included in the RFC 3095 packet format [8].

The choice of which of the above two methods to use is configured by upper layers. The mapping of the PID values is specified in subclauses 5.1.3.2 and 5.1.3.3, respectively for the above two methods.

### 5.1.3.2 Mapping of PID values for RFC 3095 with CIDs in PDCP PDU Header

The following PID values shall be mapped to the RFC 3095 header compression protocol in the order presented in the table where  $n$  is the number of PID values already mapped to other protocol packet types. As shown in the Table 3 below, the mapping of PID values for the RFC 3095 map to the CID values used by RFC 3095. The maximum CID value ( $x$ ) is configured by upper layers. If this method is configured by upper layers, PDCP shall not introduce CIDs in the ROHC packet format.

**Table 3: Mapping of PID values for RFC 3095 header compression protocol**

PID value	Optimisation method	Packet type
$n+1$	RFC 3095	CID=0
$n+2$	RFC 3095	CID=1
...	RFC 3095	...
...	RFC 3095	...
$n+x+1$	RFC 3095	CID= $x$

### 5.1.3.3 Mapping of PID values for RFC 3095 with CIDs within ROHC packet format

The following PID value shall be mapped to the RFC 3095 header compression protocol as presented in the table where  $n$  is the number of PID values already assigned to other protocol packet types.

**Table 4: Mapping of PID values for RFC 3905 header compression protocol**

PID value	Optimisation method	Packet type
$n+1$	RFC 3095	RFC 3095 packet format

If this method is configured by upper layers,, PDCP shall not be configured to accommodate ROHC CIDs in the PDCP PDU header .

### 5.1.3.4 RFC 3095 Segmentation

The RFC 3095 protocol supports segmentation. The segmentation:

- can vary on a packet-by-packet basis; and
- does not add any overhead to packets that are not segmented.

The Segmentation option of RFC 3095 shall:

- not be used when RLC is configured in non-transparent mode [5], in which case the MRRU (maximum reconstructed reception unit) shall be set equal to 0;
- only be used when RLC is configured in transparent mode and the `PACKET_SIZES_ALLOWED` is used to configure ROHC packet sizes;
- be applied if the produced packet does not fit the largest packet as indicated by `PACKET_SIZES_ALLOWED`.

### 5.1.3.5 Protocol Parameters

RFC 3095 has two types of parameters [8]:

- configuration parameters: these are mandatory and must be configured between compressor and decompressor peers.
- implementation parameters: these are optional and, when used, stipulate how RFC 3095 operates.

These parameters are categorized in four different groups, as defined below:

- M: Mandatory and configured by upper layers.
- MO: Parameters that must be supported and when used can only be configured or triggered by upper layers.
- O: Optional RFC 3095 parameters that are not configured by upper layers. They may be used locally (i.e. UTRAN and/or in UE) for RFC 3095.
- N/A: These are not used in RFC 3095.

The usage and definition of the parameters shall be as specified below.

- CID\_INCLUSION\_INFO (M): This defines whether RFC3095 CID values are transferred within PDCP headers or RFC3095 packet format. See chapter 5.1.3.1 for more information.
- MAX\_CID (M): This is the maximum CID value that can be used. One CID value shall always be reserved for uncompressed flows.
- LARGE\_CIDS: This is not configured by upper layers but inferred from the configured value of MAX\_CID according to the following rule:

If MAX\_CID > 15 then LARGE\_CIDS = TRUE else LARGE\_CIDS = FALSE.

- PROFILES (M): Profiles are used to define which profiles are allowed to be used by the UE in uplink. In downlink, all the profiles defined in [8] shall be supported.
- FEEDBACK\_FOR (N/A):
- MRRU (M): Segmentation is not used by default.
- NO\_OF\_PACKET\_SIZES\_ALLOWED (O).
- PACKET\_SIZES\_ALLOWED (MO): This parameter, if configured, governs which packet sizes in bytes may be used by RFC 3095. Thus, packet sizes not in the set of values for this parameter shall not be used.
- PAYLOAD\_SIZES (O).
- NO\_OF\_PACKET\_SIZES\_USED (O).
- PACKET\_SIZES\_USED (O).
- CONTEXT\_REINITIALIZATION (MO).
- MODE (O).
- CLOCK\_RESOLUTION (O).
- REVERSE\_DECOMPRESSION\_DEPTH (M): Default value is that reverse decompression is not used.

### 5.1.3.6 Configuration by RRC

If the variable "PDCP\_ROHC\_TARGET\_MODE" [2] is stored in the UE, and if applicable for the ROHC profile applied, the de-compressor shall only perform the operational state transitions defined in [8] to the stored mode.

If the variable "PDCP\_ROHC\_TARGET\_MODE" [2] is not stored in the UE, the de-compressor shall not restrict the operational state transitions defined in [8].

## 5.2 Void

## 5.3 Data Transfer

If header compression is configured the PDCP entity in the Sender shall:

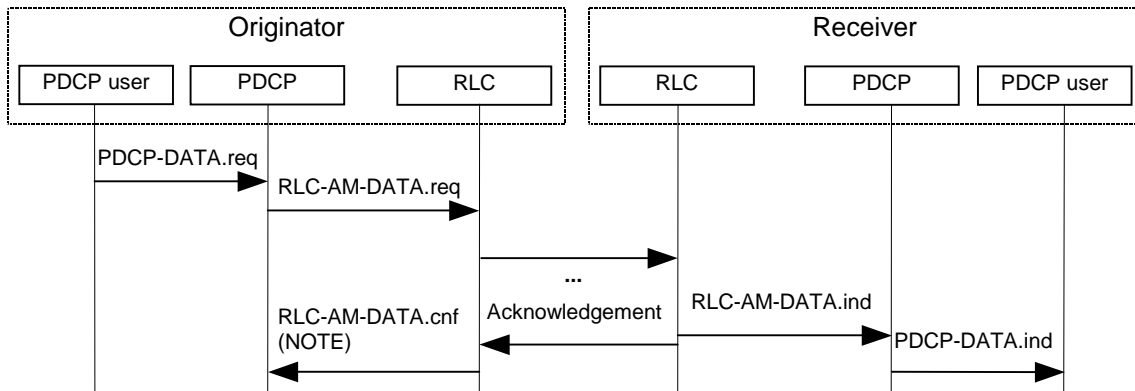
- perform header compression upon reception of a PDCP SDU from upper layers;
- if the radio bearer is configured for lossless SRNS Relocation or lossless DL RLC PDU size change:
  - maintain PDCP sequence numbering as specified in subclause 5.6.1.1;
- submit the PDCP PDU to lower layer in the sequence received from the upper layer.

When the PDCP entity at the Receiver receives the PDCP PDU from lower layers, it shall:

- perform header decompression (if header compression is configured) of the PDCP PDU to obtain the PDCP SDU; and
- deliver the PDCP SDU to the upper layer in the order received from the lower layer;
- if the received PDCP PDU is of type PDCP SeqNum PDU:
  - follow the procedure in subclause 5.6.1.2.

### 5.3.1 Data transfer over acknowledged mode RLC

Figure 2 shows the PDCP data transfer over acknowledged mode RLC.



**Figure 2: PDCP data transfer over acknowledged mode RLC**

NOTE: If the primitive RLC-AM-DATA.req is used with parameter CNF, the primitive RLC-AM-DATA.cnf is delivered. Otherwise, this primitive is not delivered.

### 5.3.2 Data transfer over unacknowledged and transparent mode RLC

Figure 3 shows the PDCP data transfer over unacknowledged or transparent mode RLC.

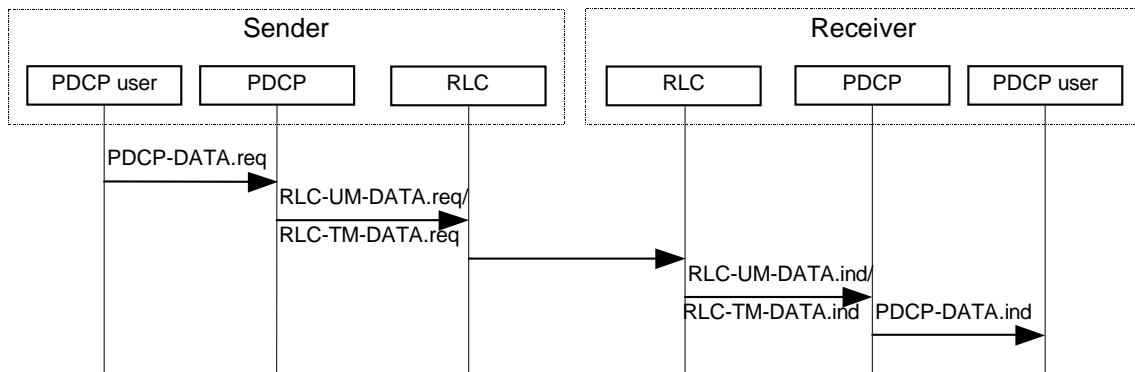


Figure 3: PDCP data transfer over unacknowledged or transparent mode RLC

## 5.4 SRNS Relocation

In case of SRNS Relocation upper layer indicates to PDCP to perform either the re-initialisation or the context relocation of compression protocols of an RB. In this version of the specification, context relocation is only applicable to RFC3095. Each of the compression protocols is handled independently, but the context relocation capability is optional for the UE and it is indicated as a part of the UE radio access capabilities.

The re-initialisation of a given compression protocol entails the following:

- Configured compression parameters remain valid during re-initialisation.
- All compression state information is initialised, e.g. header compression contexts. Therefore, the first 'compressed' packet type after SRNS Relocation is a full header.
- The PDCP sequence numbers are not changed due to the PDCP header compression protocol re-initialisation.

The context relocation of a given compression protocol entails the following:

- Configured compression parameters remain valid during context relocation.
- A snapshot of the compression state information (context) is taken in the source RNC and transferred to the target RNC, which initialises the header compression protocol according to the transferred snapshot. Therefore, the (de)compression continues after SRNS Relocation from the context used before relocation.
- Some additional specific actions are performed both in UE and UTRAN during the SRNS Relocation in order to keep the (de)compressors consistent.

### 5.4.1 Lossless SRNS Relocation

Lossless SRNS Relocation is only applicable when RLC is configured for in-sequence delivery and acknowledged mode. The support of lossless SRNS Relocation is configured by upper layer.

For the support of lossless SRNS Relocation, the PDCP entities maintain sequence numbers for PDCP SDUs, as described in subclause 5.6.1.1.

These sequence numbers are synchronised between PDCP Sender and Receiver, as described in subclause 5.6.1.2.

When a lossless SRNS Relocation is performed sequence numbers are exchanged between UE and UTRAN. They are used to confirm PDCP SDUs transmitted but not yet acknowledged by the Receiver, as described in subclause 5.6.1.3. After relocation the data transfer begins with the first unconfirmed PDCP SDU.

#### 5.4.1.1 Void

5.4.1.2 Void

5.4.1.3 Void

## 5.4.2 Context relocation

The header compression context relocation is performed by the decision of upper layers in source RNC based on the UE radio capabilities. The decision is done independently every time the SRNS relocation occurs and is specific for each header compression protocol. It is indicated to UE as a part of the SRNS relocation signalling of the upper layer and the selected relocation method is configured to UE PDCP by the upper layer.

The header compression context relocation shall not be performed if the radio bearer is configured to support the lossless SRNS Relocation.

**In the UE**, upon reception of the indication about SRNS relocation being performed:

- the upper layer configures PDCP (*CPDCP-CONFIG Req*) to perform either re-initialisation (R) or the context relocation (C) of header compression protocols;
- if the context relocation is to be applied for RFC3095 header compression protocol:
  - if the **compressor** (M-HC) is operating in **R mode**:
    - uplink data may be compressed and transmitted normally.
  - if the **compressor** (M-HC) is operating in **O mode**:
    - the header compression context shall temporarily not be updated anymore even though uplink data may be compressed and transmitted normally. When SRNS relocation is completed, M-HC should return to normal operation. To reduce the risk of decompression failure, M-HC may also transit to FO state and send IR-DYN packets.
  - if the **compressor** (M-HC) is operating in **U mode**:
    - M-HC shall transit to FO state and send IR-DYN to re-synchronise the dynamic part of the uplink context.
  - if the reverse decompression is applied in the **decompressor** (M-HD):
    - flush the reverse decompression buffer by discarding all packets in the buffer.
  - in the **decompressor** (M-HD), in **all modes**:
    - downlink data may be received and decompressed normally.

**In the UTRAN source RNC**, while SRNS relocation is being performed:

- if the context relocation is to be applied for RFC3095 header compression protocol:
  - PDCP is requested to take a context snapshot by the upper layer (*CPDCP-CONTEXT Req*);
  - if the **compressor** (source N-HC) is operating in **R mode**:
    - the source N-HC should take a snapshot of its header compression compressor context (denoted *N-context-C\**);
    - header compression contexts should not be updated anymore even though downlink data may be compressed and transmitted otherwise normally. This can be done by sending R-1\* packets.
  - if the **compressor** (source N-HC) is operating in **O mode**:



- the source N-HC should take a snapshot of its header compression compressor context (denoted *N-context-C\**);
- after the snapshot is taken, the source N-HC should only send UO-0 or UO-1\* packets. This means only RTP SN, RTP TS, and IP-ID (for IPv4 only) fields are updated in the decompressor context at M-HD.
- if the **compressor** (source N-HC) is operating in **U mode**:
  - the source N-HC should take a snapshot of the static part of its header compression compressor context (denoted *N-context-C-static\**).
- if the **decompressor** (source N-HD) is operating in **R or O mode**:
  - if the source N-HD is sure about the integrity of the *N-context-D*:
    - the source N-HD should take a snapshot of its header compression decompressor context (denoted as *N-context-D\**).
  - if the source N-HD is only sure about the integrity of the static part of the *N-context-D* (e.g. due to multiple detected errors):
    - the source N-HD should take a snapshot only of the static part of the *N-context-D* (denoted as *N-context-D-static\**).
  - RFC3095 acknowledgments should not be generated anymore even though uplink data may be received and decompressed otherwise normally.
- if the **decompressor** (source N-HD) is operating in **U mode**:
  - the source N-HD should take a snapshot of the static part of its header compression decompressor context (denoted *N-context-D-static\**).
- either *N-context-C\** or *N-context-C-static\** and either *N-context-D\** or *N-context-D-static\** should be delivered to the upper layer as *Context-Info (CPDCP-CONTEXT.Conf)*, which is to be transmitted further to the target RNC.

**In the UTRAN target RNC**, while SRNS relocation is being performed:

- the upper layer configures PDCP (*CPDCP-CONFIG Req*) to perform either initialisation (I) or the context relocation (C) of header compression protocols;
- the new header compression entity should be created;
- if the context relocation is to be applied for RFC3095 header compression protocol:
  - in the **compressor** (target N-HC), in **all modes**:
    - the header compression compressor (target N-HC) should be initialised to the same mode as used in the source N-HC using *N-context-C\** as the initial header compression compressor context;
    - in addition, if the source N-HC was operating in U-mode, the target N-HC should first send IR-DYN to resynchronise the dynamic part of the downlink context.
  - in the **decompressor** (target N-HD), in **all modes**:
    - if *Context-Info* carries *N-context-D\**:
      - the header compression decompressor (target N-HD) should be initialised to the same mode as used in the source N-HD using *N-context-D\** as the initial header compression decompressor context.
    - if *Context-Info* carries *N-context-D-static\**:
      - the header compression decompressor (target N-HD) should be initialised to the same mode as used in the source N-HD using *N-context-D-static\** as the initial header compression decompressor context;
      - the target N-HD should send a request for IR-DYN.

## 5.5 Lossless DL RLC PDU size change

Lossless DL RLC PDU size change is only applicable when RLC is configured for in-sequence delivery and acknowledged mode. The support of lossless DL RLC PDU size change is configured by upper layer.

For the support of lossless DL RLC PDU size change, the PDCP entities maintain sequence numbers for DL PDCP SDUs, as described in subclause 5.6.1.1.

These DL sequence numbers are synchronised between PDCP Sender in the UTRAN and Receiver in the UE, as described in subclause 5.6.1.2.

When a lossless DL RLC PDU size change is performed the next expected DL\_Receive PDCP SN is sent from the UE to the UTRAN. It is used to confirm DL PDCP SDUs transmitted but not yet acknowledged by the Receiver in the UE, as described in subclause 5.6.1.3. After lossless DL RLC PDU size change the data transfer begins with the first unconfirmed DL PDCP SDU.

## 5.6 General procedures

### 5.6.1.1 PDCP Sequence Numbering

The value of the PDCP sequence number ranges from 0 to 65535. The PDCP SN window size indicates the maximum number of PDCP SDUs, not confirmed to have been successfully transmitted to the peer entity by lower layer, that can be numbered at any given time. The PDCP SN window size is configured by upper layers. PDCP sequence numbers are set to "0" when the PDCP entity is set-up for the first time.

In the following the "submission/reception of a PDCP SDU to/from lower layer" is used as a synonym for the submission/reception of a PDCP Data PDU or a PDCP SeqNum PDU to/from lower layer that carries in its Data field a compressed or uncompressed PDCP SDU.

If lossless SRNS relocation and/or lossless DL RLC PDU size change are/is supported by the UE, for each radio bearer configured to support "lossless SRNS relocation or lossless DL RLC PDU size change" as specified in [1], PDCP sequence numbers are applied:

- in the UE:
  - the DL\_Receive PDCP SN shall be set to "0" for the first PDCP SDU received from lower layer;
  - the DL\_Receive PDCP SN shall be incremented by "1" for the next PDCP SDU received from lower layer.
- in the UTRAN:
  - the DL\_Send PDCP SN should be set to "0" for the first PDCP SDU submitted to lower layer;
  - the DL\_Send PDCP SN should be incremented by "1" for the next PDCP SDU submitted to lower layer.

Additionally, if lossless SRNS relocation is supported by the UE, for each radio bearer configured to support "lossless SRNS relocation or lossless DL RLC PDU size change" as specified in [1], PDCP sequence numbers are applied:

- in the UE:
  - the UL\_Send PDCP SN shall be set to "0" for the first PDCP SDU submitted to lower layer;
  - the UL\_Send PDCP SN shall be incremented by "1" for the next PDCP SDU submitted to lower layer;
- in the UTRAN:
  - the UL\_Receive PDCP SN should be set to "0" for the first PDCP SDU received from lower layer;
  - the UL\_Receive PDCP SN should be incremented by "1" for the next PDCP SDU received from lower layer.

PDCP sequence numbers shall not be decremented in a PDCP entity.

### 5.6.1.2 PDCP Sequence Number synchronization

For radio bearers that are configured to support "lossless SRNS Relocation or lossless DL RLC PDU size change" as specified in [1]:

the UE PDCP entity shall:

- if the UE supports lossless SRNS relocation,
  - if a PDCP entity has to synchronise the UL PDCP SN following a RLC reset or RLC transmitting side re-establishment not caused by a lossless SRNS Relocation; or
  - if the UE PDCP entity receives an invalid "next expected UL Receive PDCP SN" from upper layer after a lossless SRNS Relocation;
    - trigger the PDCP SN synchronisation procedure by submitting one PDCP SeqNum PDU to lower layer;
    - consider that the synchronisation procedure is complete on confirmation by lower layer of the successful transmission of the PDCP SeqNum PDU.
  - if the UE supports lossless DL RLC PDU size change but not lossless SRNS relocation, the UE PDCP entity shall not submit PDCP SeqNum PDU to lower layer.
- the UTRAN PDCP entity should:
  - if a PDCP entity has to synchronise the DL PDCP SN following a RLC reset or RLC transmitting side re-establishment not caused by a lossless SRNS Relocation or a lossless DL RLC size change; or
  - if the UTRAN PDCP entity receives an invalid "next expected DL\_Receive PDCP SN" from upper layer after lossless SRNS Relocation or lossless DL RLC PDU size change:
    - trigger the PDCP SN synchronisation procedure by submitting one PDCP SeqNum PDU to lower layer;
    - consider that the synchronisation procedure is complete on confirmation by lower layer of the successful transmission of the PDCP SeqNum PDU.

In the UE/UTRAN, the "next expected UL/DL\_Receive PDCP SN" is considered invalid if its value is less than the UL/DL\_Send PDCP SN of the first transmitted but not yet acknowledged PDCP SDU or greater than that of the first unsent PDCP SDU.

On receiving a PDCP SeqNum PDU:

- the UE PDCP entity shall:
  - set the value of the DL\_Receive PDCP SN to the value indicated in the PDCP SeqNum PDU;
- the UTRAN PDCP entity should:
  - set the value of the UL\_Receive PDCP SN to the value indicated in the PDCP SeqNum PDU.

NOTE: If UTRAN has the intention to use only the lossless DL RLC PDU size change, UTRAN may not maintain UL PDCP SN. In this case, UTRAN should still transfer the user data to upper layer.

### 5.6.1.3 Sequence Number and Data Forwarding

In case of a lossless SRNS Relocation procedure or lossless DL RLC size change, as described in [1]:

- the UE shall send to the UTRAN the next expected DL\_Receive PDCP SN.

Additionally, in case of lossless SRNS relocation procedure, as described in [1]:

- the UTRAN should send to the UE the next expected UL\_Receive PDCP SN.

This information exchange synchronises the Sequence Numbers at the UE and UTRAN PDCP entities.

When requested by the upper layer, for each radio bearer configured to support lossless SRNS Relocation, the PDCP sublayer in the source RNC should forward the following to the target RNC:

- the UL\_Receive PDCP SN of the next PDCP SDU expected to be received from the UE;
- the DL\_Send PDCP SN of the first transmitted but not yet acknowledged PDCP SDU;
- the transmitted but not yet acknowledged PDCP SDUs together with their related DL\_Send PDCP SNs;
- the not yet transmitted PDCP SDUs.

## 5.7 Header Compression and Decompression for MBMS

Header compression and decompression for a p-t-m MBMS service is performed in ROHC U-mode.

Applying ROHC for p-t-m RB configuration is optional and should be configured per MBMS session.

### 5.7.1 Cell change inside the same cell group

When a UE moves to a cell that is the same cell group with that of the previous cell while receiving a p-t-m MBMS service which is provided in both cells, the UE shall:

- continue header decompression during and after cell change.

### 5.7.2 Cell change between cell groups

When a UE moves to a cell that is the different cell group with that of the previous cell while receiving a p-t-m MBMS service which is provided in both cells, the UE shall:

- reconfigure and re-initialise the PDCP entity;
- reuse the static part of the header decompressor context used in the previous cell;
- re-initialise header decompression after receiving IR-DYN or IR packet and correcting the CID of the context with the CID being used in the new cell.

NOTE: How to correct CID is implementation issue and is not specified.

---

# 6 Services

## 6.1 Services provided to upper layers

The following services are provided by PDCP to upper layers:

- transfer of user data;
- maintenance of PDCP SDU sequence numbers.

## 6.2 Services expected from RLC layer

For a detailed description of the following functions see [5].

- transparent data transfer Service;
- unacknowledged data transfer Service;
- acknowledged data transfer Service.

## 7 Elements for layer-to-layer communication

The interaction between the PDCP layer and other layers are described in terms of primitives where the primitives represent the logical exchange of information and control between the PDCP layer and other layers. The primitives shall not specify or constrain implementations.

### 7.1 Primitives between PDCP and upper layers

The primitives between PDCP and upper layers are shown in Table 5.

**Table 5: Primitives between PDCP and upper layers**

Generic Name	Parameter			
	Req.	Ind.	Resp.	Conf.
PDCP-DATA	Data	Data	Not Defined	Not Defined
CPDCP-CONFIG	PDCP-Info, RLC-SAP SN_Sync, R/I/C/RS, Context-Info	Not Defined	Not Defined	Not Defined
CPDCP-CONTEXT	None	Not Defined	Not Defined	Context-Info
CPDCP-RELEASE	RLC-SAP	Not Defined	Not Defined	Not Defined
CPDCP-SN	PDCP SN	Not Defined	Not Defined	Not Defined
CPDCP-RELOC	Next_Receive_SN	Not Defined	Not Defined	Next_Receive_SN, Next_Send_SN

Each Primitive is defined as follows:

- a) PDCP-DATA-Req./Ind.
  - PDCP-DATA-Req is used by upper user-plane protocol layers to request a transmission of upper layer PDU. PDCP-DATA-Ind is used to deliver PDCP SDU that has been received to upper user plane protocol layers.
- b) CPDCP-CONFIG-Req.
  - CPDCP-CONFIG-Req is used to configure and – in case of already existing PDCP entity – to reconfigure a PDCP entity and to assign it to the radio bearer associated with that entity.
- c) CPDCP-RELEASE-Req.
  - CPDCP-RELEASE-Req is used by upper layers to release a PDCP entity.
- d) CPDCP-SN-Req.
  - This primitive is used at the UTRAN. CPDCP-SN-Req is used to transfer the PDCP SN to PDCP.
- e) CPDCP-RELOC-Req/Conf.
  - CPDCP-RELOC-Req initiates the SRNS Relocation procedure in PDCP for those radio bearers that are configured to support lossless SRNS Relocation. The Next\_Receive\_SN is only included at the UE side.
  - CPDCP-RELOC-Conf is used to transfer the Next\_Receive\_SN and/or Next\_Send\_SN to upper layers for lossless SRNS Relocation. The Next\_Send\_SN is only included at the source RNC.
- f) CPDCP-CONTEXT-Req./Conf.
  - CPDCP-CONTEXT-Req initiates specific actions in the source RNC in order to perform context relocation as a part of the SRNS relocation. The primitive is applicable only in the source RNC.
  - CPDCP-CONTEXT-Conf is used to transfer the header compression context information from PDCP to upper layer in order to perform context relocation as a part of the SRNS relocation. The primitive is applicable only in the source RNC.

The following parameters are used in the primitives:

- 1) PDCP-Info:
  - Contains the parameters for each of the header compression protocols configured to be used by one PDCP entity.
- 2) RLC-SAP:
  - The RLC-SAP (TM/UM/AM) used by PDCP entity when communicating with RLC sublayer.
- 3) SN\_Sync:
  - Indicates that PDCP should start PDCP SN synchronisation procedure.
- 4) Next\_Send\_SN:
  - The Send PDCP SN of the next PDCP SDU to be sent. There is one in the uplink (UL\_Send PDCP SN) and one in the downlink (DL\_Send PDCP SN). Refer to subclause 5.4.1.
- 5) Next\_Receive\_SN:
  - The Receive PDCP SN of the next PDCP SDU expected to be received. There is one in the uplink (UL\_Receive PDCP SN) and one in the downlink (DL\_Receive PDCP SN). Refer to subclause 5.4.1.
- 6) PDCP SN:
  - This includes a PDCP sequence number.
- 7) R/I/C/RS
  - Indicates that PDCP should Re-initialise (R)/Initialise (I) the header compression protocols. Alternatively (Context-relocation, C) it indicates that UE PDCP shall perform specific actions related to context relocation during SRNS relocation. (RS) indicates to Re-initialise while keeping the static part of the header compression (only for RFC 3095). R/I/C/RS indication is given separately for each of the configured header compression protocol, if several exist for a given radio bearer.
- 8) Context-Info:
  - Contains the header compression context information of each of the header compression protocols that are subject to the context relocation during SRNS relocation.

---

## 8 Elements for peer-to-peer communication

### 8.1 Protocol data units

Different PDU formats are defined for the PDCP protocol, one not introducing any overhead to the (compressed) PDCP SDU, others introducing such overhead.

### 8.2 Formats

A PDCP PDU shall be a multiple of 8 bits, if the RLC entity is configured for unacknowledged or acknowledged mode. Otherwise, if the RLC entity is configured for transparent mode, it is bit-aligned. In Tables 6, 7 and 8, bit strings are represented as follows: the first bit is the leftmost one on the first line of the table, the last bit is the rightmost on the last line of the table, and more generally the bit string is to be read from left to right and then in the reading order of the lines.

SDUs are bit strings, with any non-null length. If not compressed within PDCP an SDU is included from first bit onward.

### 8.2.1 PDCP-No-Header PDU

The PDCP-No-Header PDU does not introduce any overhead to the PDCP SDU. The use of the PDCP-No-Header PDU is configured by the upper layer.

The format of the PDCP-No-Header PDU is shown in Table 6.

**Table 6: PDCP-No-Header PDU**

Data
------

### 8.2.2 PDCP Data PDU

The PDCP Data PDU is used to convey:

- data containing an uncompressed PDCP SDU; or
- header compression related control signalling; or
- data that has been obtained from PDCP SDU after header compression.

The format of the PDCP Data PDU is shown in Table 7.

**Table 7: PDCP Data PDU format**

PDU type	PID
Data	

### 8.2.3 PDCP SeqNum PDU

The PDCP SeqNum PDU is used to convey a PDCP SDU sequence number and:

- data containing an uncompressed PDCP SDU; or
- data that has been obtained from PDCP SDU after header compression.

The format of the PDCP SeqNum PDU is shown in Table 8.

**Table 8: PDCP SeqNum PDU format**

PDU type	PID
Sequence number	
Data	

## 8.3 Parameters

If not otherwise mentioned in the definition of each field then the bits in the parameters shall be interpreted as follows: the left most bit string is the first and most significant and the right most bit is the last and least significant bit.

Unless otherwise mentioned, integers are encoded in standard binary encoding for unsigned integers. In all cases the bits appear ordered from MSB to LSB when read in the PDU.

### 8.3.1 PDU Type

Length: 3 bits.

The PDU type field indicates the PDCP Data PDU type.

Bit	PDU Type
000	PDCP Data PDU (Table 7)
001	PDCP SeqNum PDU (Table 8)
010-111	Reserved (PDUs with this encoding are invalid for this version of the protocol)

### 8.3.2 PID

Length: 5 bits.

The PID field indicates the used header compression and packet type or a context identifier.

Bit	Description
00000	No header compression
00001-11111	Dynamically negotiated header compression identifier, as described in subclause 5.1.1

The PID field value indicates the used header compression protocol type and packet type or CID. A specific header compression protocol may utilize a certain range of consecutive values from the PID field value space for different packet types. The Receiving PDCP entity performs the necessary operation (e.g. header decompression) according to the PID field value.

### 8.3.3 Data

The Data field may include either one of the following:

- Uncompressed PDCP SDU;
- Header compressed PDCP SDU;
- Header compression protocol feedback information.

### 8.3.4 Sequence number

Length: 16 bits

PDCP SDU sequence number.

## 9 Handling of unknown, unforeseen and erroneous protocol data

### 9.1 Invalid PDU type

If a PDCP entity receives a PDCP PDU with a PDU Type set to Reserved (see subclause 8.3.1), it shall:

- discard the PDCP PDU.

If a PDCP entity is not configured for lossless SRNS Relocation or lossless DL RLC PDU size change and receives a PDCP SeqNum PDU, it shall:



- discard the PDCP SeqNum PDU.

## 9.2 Invalid PID value

If a PDCP entity receives a PDCP PDU with a PID value that is not mapped with a valid packet type (see subclause 5.1.1), it shall:

- discard the PDCP PDU.

---

# Annex A (normative): ROHC performance testing

## A.1 Introduction

All ROHC performance tests are carried out by providing a sequence of uncompressed IP/UDP/RTP packets to the ROHC RTP compressor, together with some artificial feedback messages synchronized with the packet sequence. All packets in these sequences are built on the same base structure, with most field values being constant, and with a fixed payload size of 32 octets.

The normative structure of the IPv6/UDP/RTP header is outlined in A.3, along with tables of the values to be used for each field. Fields with values marked ANY can have any value, as these are the addressing fields that are used to identify a unique packet stream for compression, and the content of them does not otherwise affect compression, they are either sent in uncompressed form or completely omitted in compressed packets. The checksum values are dependent on the complete content of the packet and must be calculated according to the protocol specifications, RFC 768 and RFC 791, which are referred to in the tables. A dummy payload is to be appended at the end of the header data.

Four fields (IP ID, IP TTL/HL, RTP SN, and RTP TS) are affected by the test sequence variations currently being considered for these ROHC RTP performance tests. Values for these fields must thus be found in the test sequence details of each test.

In the following sections a format with regards to outline, sequences and requirements are exemplified. Test 1a and 1b are base tests using a completely regular packet stream as input. All subsequent tests are based on test 1a or test 1b, each with specific test events added to the base test packet stream.

## A.2 Test outline

### A.2.1 Test 1a - Base test of ROHC RTP O-mode compressor

#### A.2.1.1 Test purpose

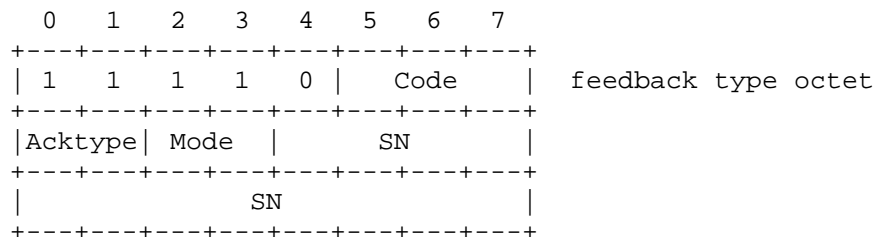
The purpose of the base test is to verify that the compressor implements an active and efficient compression for a regular IP/UDP/RTP packet stream, i.e. that it makes use of the most efficient compressed packet formats provided by ROHC RTP [8] for O-mode.

#### A.2.1.2 Sequence details

A 5 second packet sequence with 50 packets per second is used where all header fields are set according to the basic test packet structure, as described in subclause A.3, with addition of the following:

1. The Time To Live (TTL) / Hop Limit field is set to the value 0x20
2. The RTP Sequence Number is a linearly increasing counter with a packet-to-packet delta of 1, set to 0x0000 for the first packet and thus ending with 0x00F9 (249) in the last packet of the sequence
3. The RTP Time Stamp is a linearly increasing counter with a packet-to-packet delta of 160, set to 0x00000000 for the first packet and thus ending with 0x00009BA0 (39840) in the last packet of the sequence.
4. The IP Identification is set to the same value as the RTP Sequence Number

Between the 6th and 7th packet of the sequence, a ROHC feedback packet of ROHC RTP feedback type 2 is to be given to the ROHC compressor to trigger an immediate transition to O-mode operation. The format of that packet is as follows:



Where:

- Code is set to 0x2 (indicates that feedback data above the type octet is 2 octets)
- Acktype is set to 0x0 (means ACK)
- Mode is set to 0x2 (means O-mode)
- SN is set to 0x000

### A.2.1.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv4: xx octets [TBD]
- With IPv6: yy octets [TBD]

## A.2.2 Test 1b - Base test of ROHC RTP R-mode compressor

### A.2.2.1 Test purpose

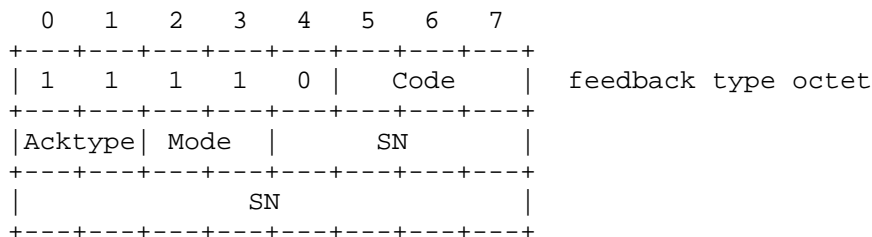
The purpose of the base test is to verify that the compressor implements an active and efficient compression for a regular IP/UDP/RTP packet stream, i.e. that it makes use of the most efficient compressed packet formats provided by ROHC RTP [8] for R-mode.

### A.2.2.2 Sequence details

A 5 second packet sequence with 50 packets per second is used where all header fields are set according to the basic test packet structure, as described in subclause A.3, with addition of the following:

1. The Time To Live (TTL) / Hop Limit field is set to the value 0x20
2. The RTP Sequence Number is a linearly increasing counter with a packet-to-packet delta of 1, set to 0x0000 for the first packet and thus ending with 0x00F9 (249) in the last packet of the sequence
3. The RTP Time Stamp is a linearly increasing counter with a packet-to-packet delta of 160, set to 0x00000000 for the first packet and thus ending with 0x00009BA0 (39840) in the last packet of the sequence.
4. The IP Identification is set to the same value as the RTP Sequence Number

Between the 6th and 7th (SN=5 and SN=6) packet of the sequence, a ROHC feedback packet of ROHC RTP feedback type 2 is to be given to the ROHC compressor to initiate transition to R-mode operation. The format of that packet is as follows:



Where:

- Code is set to 0x2 (indicates that feedback data above the type octet is 2 octets)
- Acktype is set to 0x0 (means ACK)
- Mode is set to 0x3 (means R-mode)
- SN is set to 0x000

After that, an additional feedback packet with the same content as above except for the SN value, which now must be set to 0x006, is to be given to the compressor between the 12th and 13th (SN=11 and SN=12) packet of the sequence. This will complete transition to R-mode.

### A.2.2.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv4: xx octets [TBD]
- With IPv6: yy octets [TBD]

## A.2.3 Test 2a - TTL / Hop-Limit variations in O-mode

### A.2.3.1 Test purpose

The purpose of the TTL/Hop-Limit test is to verify that the compressor can efficiently handle changes in the TTL/Hop-Limit value, i.e. use the most efficient header extension provided by ROHC RTP [8] for O-mode.

### A.2.3.2 Sequence details

The test sequence is the same as in subclause A.2.1, with the following exception:

- For packets with SN between 20 and 29, the Time To Live (TTL) / Hop Limit value is set to 0x22

### A.2.3.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv6: yy octets [TBD]

## A.2.4 Test 2b - TTL / Hop-Limit variations in R-mode

### A.2.4.1 Test purpose

The purpose of the TTL/Hop-Limit test is to verify that the compressor can efficiently handle changes in the TTL/Hop-Limit value, i.e. use the most efficient header extension provided by ROHC RTP [8] for R-mode.

#### A.2.4.2 Sequence details

The test sequence is the same as in subclause A.2.2, with the following exception:

- For packets with SN between 20 and 29, the Time To Live (TTL) / Hop Limit value is set to 0x22

#### A.2.4.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv6: yy octets [TBD]

### A.2.5 Test 3a - Re-establishment TS after DTX in O-mode

#### A.2.5.1 Test purpose

The purpose of the TS re-establish test is to verify that the compressor can efficiently re-establish the proper TS value after a DTX period, i.e. use the most efficient header extension(s) provided by ROHC RTP [8] for O-mode.

#### A.2.5.2 Sequence details

The test sequence is the same as in subclause A.2.1, with the following exception:

1. The RTP Time Stamp is a linearly increasing counter with a packet-to-packet delta of 160, set to 0x00000000 for the first packet.
2. For packet with an SN of 20, TS is increased to represent a 32 (0.64 seconds) packet skip ( $32 \times 160$ ) and is thus set to  $(20+32) \times 160 = 8320$  (0x00002080). Then TS continues to grow as stated in 1 above.
3. For packet with an SN of 30, TS is increased to represent a 128 (2.56 seconds) packet skip ( $128 \times 160$ ) and is thus set to  $(30+32+128) \times 160 = 30400$  (0x000076C0). Then TS continues to grow as stated in 1 above.
4. For packet with an SN of 40, TS is increased to represent a 2048 (40.96 seconds) packet skip ( $2048 \times 160$ ) and is thus set to  $(40+32+128+2048) \times 160 = 359680$  (0x00057D00). Then TS continues to grow as stated in 1 above.
5. TS thus ends at 393120 (0x0005FFA0) in the last packet of the sequence with RTP sequence number 249.

#### A.2.5.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv6: xx octets [TBD]

### A.2.6 Test 3b - Re-establishment TS after DTX in R-mode

#### A.2.6.1 Test purpose

The purpose of the TS re-establish test is to verify that the compressor can efficiently re-establish the proper TS value after a DTX period, i.e. use the most efficient header extension(s) provided by ROHC RTP [8] for R-mode.

#### A.2.6.2 Sequence details

The test sequence is the same as in subclause A.2.2, with the following exception:

1. The RTP Time Stamp is a linearly increasing counter with a packet-to-packet delta of 160, set to 0x00000000 for the first packet.
2. For packet with an SN of 20, TS is increased to represent a 32 (0.64 seconds) packet skip ( $32 \times 160$ ) and is thus set to  $(20+32) \times 160 = 8320$  (0x00002080). Then TS continues to grow as stated in 1 above.

3. For packet with an SN of 30, TS is increased to represent a 128 (2.56 seconds) packet skip (128x160) and is thus set to  $(30+32+128)\times 160=30400$  (0x000076C0). Then TS continues to grow as stated in 1 above.
4. For packet with an SN of 40, TS is increased to represent a 2048 (40.96 seconds) packet skip (2048x160) and is thus set to  $(40+32+128+2048)\times 160=359680$  (0x00057D00). Then TS continues to grow as stated in 1 above.
5. TS thus ends at 393120 (0x0005FFA0) in the last packet of the sequence with RTP sequence number 249.

### A.2.6.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv6: yy octets [TBD]

## A.2.7 Test 4a - Compressor response to single lost packets in O-mode

### A.2.7.1 Test purpose

The purpose of this test is to verify that the compressor does not panic just because there is a single missing packet, i.e. the compressed packet size should not increase due to such events.

### A.2.7.2 Sequence details

The test sequence is the same as in subclause A.2.1, with the following exception:

- Packets with SN 20, 30, and 40 are removed from the sequence.

### A.2.7.3 Test requirement

Maximal compressed header overhead for the test are the same as in A2.1.

## A.2.8 Test 4b - Compressor response to single lost packets in R-mode

### A.2.8.1 Test purpose

The purpose of this test is to verify that the compressor does not panic just because there is a single missing packet, i.e. the compressed packet size should not increase due to such events.

### A.2.8.2 Sequence details

The test sequence is the same as in subclause A.2.2, with the following exception:

- Packets with SN 20, 30, and 40 are removed from the sequence.

### A.2.8.3 Test requirement

Maximal compressed header overhead for the test are the same as in A2.2.

## A.2.9 Test 5a - Compressor response to several packet losses in O-mode

### A.2.9.1 Test purpose

The purpose of this test is to verify that the compressor can efficiently handle events when there are several consecutive pre-compressor packet losses in the packet stream, i.e. that the compressor makes use the most efficient header extension provided by ROHC RTP [8] for O-mode.

### A.2.9.2 Sequence details

The test sequence is the same as in subclause A.2.1, with the following exception:

- Packets with SN 20-25 are removed from the sequence.

### A.2.9.3 Test requirement

Maximal compressed header overhead for the test sequence:

- With IPv6: yy octets [TBD]

## A.2.10 Test 5b - Compressor response to several packet losses in R-mode

### A.2.10.1 Test purpose

The purpose of this test is to verify that the compressor can efficiently handle events when there are several consecutive pre-compressor packet losses in the packet stream, i.e. that the compressor makes use the most efficient header extension provided by ROHC RTP [8] for R-mode.

### A.2.10.2 Sequence details

The test sequence is the same as in subclause A.2.2, with the following exception:

- Packets with SN 20-25 are removed from the sequence.

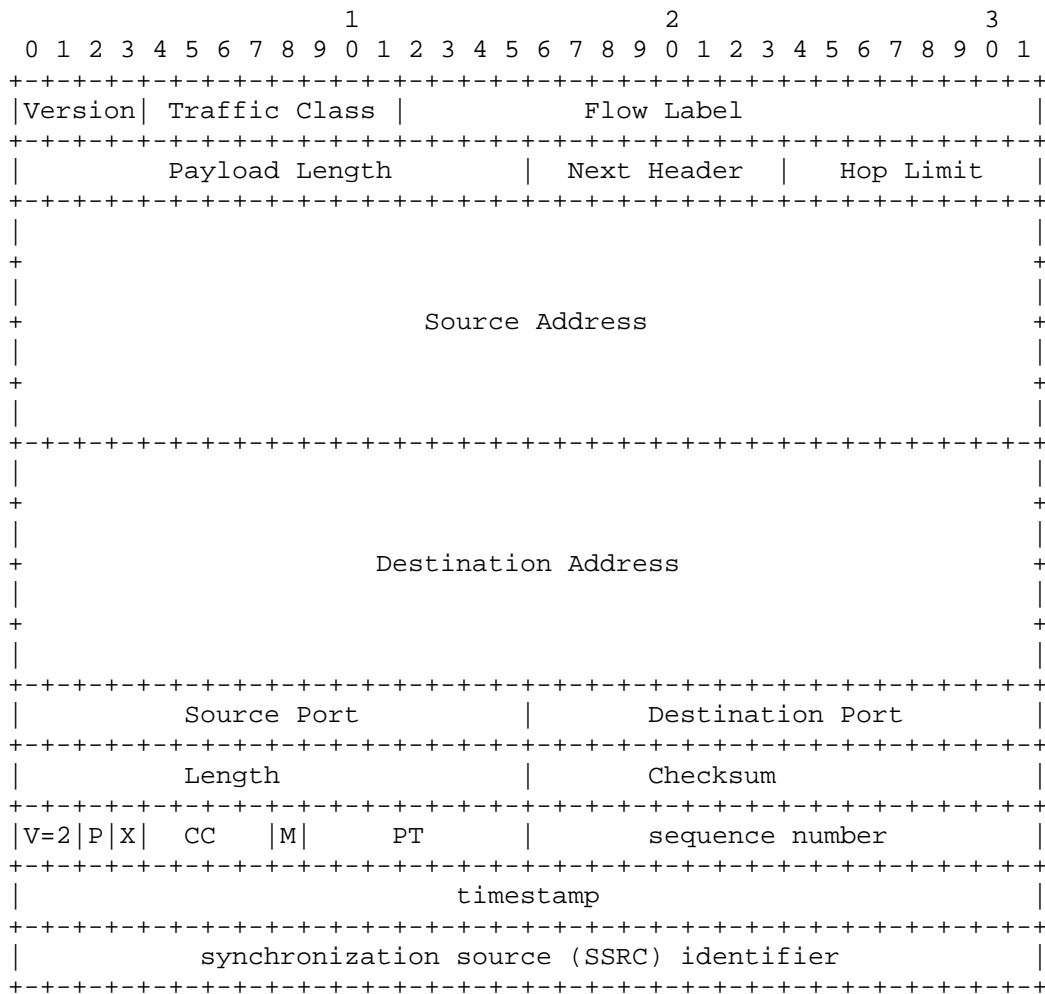
### A.2.10.3 Test requirement

Maximal compressed header overhead for the test sequence:

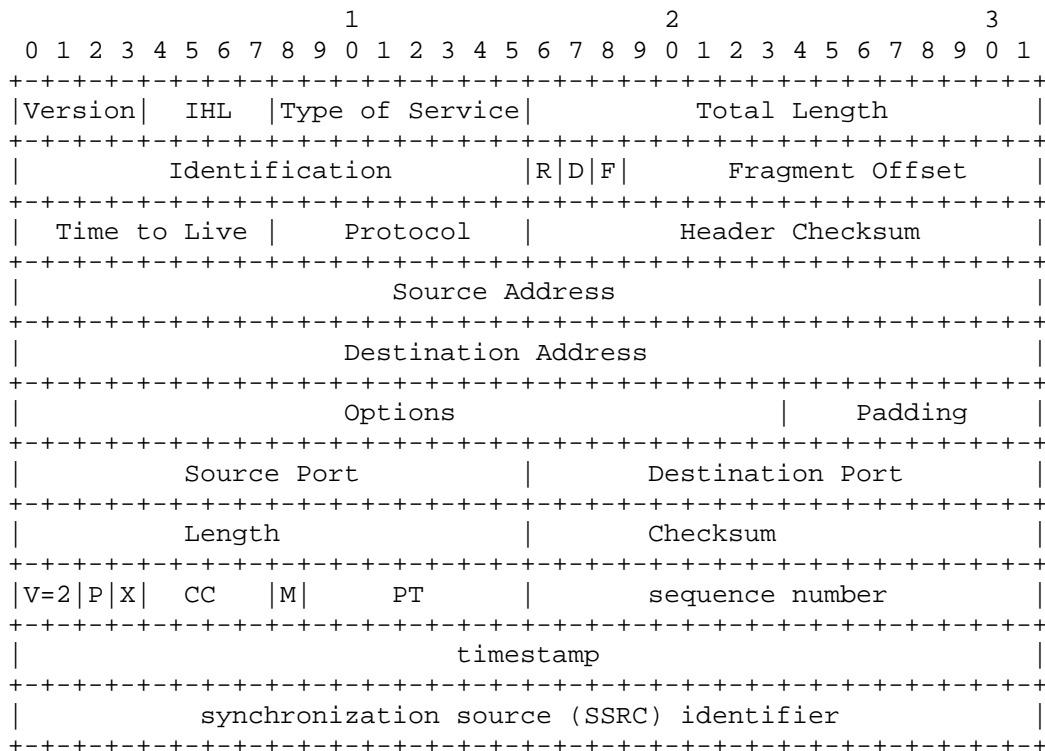
- With IPv6: yy octets [TBD]

## A.3 Test packet structures

IPv6/UDP/RTP



IPv4/UDP/RTP





## IPv6 header fields

Field	Size (bits)	Value
Version	4	0x6
Traffic Class	8	0x00
Flow Label	20	0x00000
Payload Length	16	0x0034
Next Header	8	0x11
Hop Limit	8	Test dependent
Source Address	128	ANY
Destination Address	128	ANY

## IPv4 header fields

Field	Size (bits)	Value
Version	4	0x4
Header Length (IHL)	4	0x5
Type Of Service	8	0x00
Packet Length	16	0x0048
Identification	16	Test dependent
Reserved flag (R)	1	0x0
Don't Fragment (D)	1	0x1
More Fragments (F)	1	0x0
Fragment Offset	13	0x0000
Time To Live	8	Test dependent
Protocol	8	0x11
Header Checksum	16	See RFC 791
Source Address	32	ANY
Destination Address	32	ANY

## UDP header fields

Field	Size (bits)	Value
Source Port	16	ANY
Destination Port	16	ANY
Length	16	0x0034
Checksum	16	See RFC 768

## RTP header fields

Field	Size (bits)	Value
Version (V)	2	0x2
Padding (P)	1	0x0
Extension (X)	1	0x0
CSRC Counter (CC)	4	0x0
Marker (M)	1	0x0
Payload Type (PT)	7	0x60
Sequence Number	16	Test dependent
Timestamp	32	Test dependent
SSRC	32	ANY

## Annex B (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
12/1999	RP-06	RP-99645	-		Approved at TSG-RAN #6 and placed under Change Control	-	3.0.0
03/2000	RP-07	RP-000041	004		Bit order of PDCP PDUs	3.0.0	3.1.0
	RP-07	RP-000041	005		Changes to PDCP	3.0.0	3.1.0
06/2000	RP-08	RP-000221	006	4	Changes in PDCP PDU format due to PDCP sequence numbering	3.1.0	3.2.0
09/2000	RP-09	RP-000359	009	3	Clarification of PDCP Sequence Numbering	3.1.0	3.2.0
	RP-09	RP-000359	011		Clarification on how to handle invalid PDUs	3.2.0	3.3.0
	RP-09	RP-000359	012	2	Primitives required for SRNS relocation	3.2.0	3.3.0
	RP-09	RP-000359	015		Handling of invalid PDCP PDU sequence number	3.2.0	3.3.0
03/2001	RP-11	RP-010027	018	1	Editorial Corrections	3.3.0	3.4.0
	RP-11	RP-010027	019	1	Updates necessary for Rel-4 specification	3.3.0	3.4.0
	RP-11	RP-010039	017	2	Robust Header Compression	3.4.0	4.0.0
06/2001	RP-12	RP-010310	021		Clarification on PDCP Sequence numbering	4.0.0	4.1.0
09/2001	RP-13	RP-010543	028		Header compression protocol reinitialisation during SRNS relocation	4.1.0	4.2.0
	RP-13	RP-010543	030		PDCP SDU Sequence Numbering	4.1.0	4.2.0
	RP-13	RP-010543	036		Corrections to PDCP	4.1.0	4.2.0
	RP-13	RP-010553	026		Selection of the RFC 3095 CID transmission	4.1.0	4.2.0
12/2001	RP-14	RP-010762	038		General PDCP corrections	4.2.0	4.3.0
	RP-14	RP-010772	039		Management of Full Header transmission	4.2.0	4.3.0
03/2002	RP-15	RP-020069	043		Clarification on PDCP sequence numbering	4.3.0	4.4.0
	RP-15	-	-		Upgrade to Release 5 - no technical change	4.4.0	5.0.0
06/2002	RP-16	RP-020328	049		Clarification on PDCP sequence number synchronization procedure	5.0.0	5.1.0
	RP-16	RP-020345	050		RFC 3095 context relocation	5.0.0	5.1.0
09/2002	RP-17	RP-020540	053		Mapping relation between PDCP and RLC	5.1.0	5.2.0
	RP-17	RP-020552	055		Corrections to RFC3095 operation	5.1.0	5.2.0
	RP-17	RP-020552	057		Mismatches between Rel-4 and R'99 in PDCP	5.1.0	5.2.0
12/2003	RP-22	-	-		Upgrade to Release 6 - no technical changes	5.2.0	6.0.0
03/2005	RP-27	RP-050067	059	2	Lossless DL RLC PDU size change	6.0.0	6.1.0
06/2005	RP-28	RP-050315	0060	1	Introduction of MBMS	6.1.0	6.2.0
	RP-28	RP-050303	0062	2	Target mode for ROHC operation	6.1.0	6.2.0
	RP-28	RP-050303	0064		Performance testing of ROHC	6.1.0	6.2.0
09/2005	RP-29	RP-050468	0065		Correction of MBMS header compression	6.2.0	6.3.0

---

## History

<b>Document history</b>		
V6.0.0	December 2003	Publication
V6.1.0	March 2005	Publication
V6.2.0	June 2005	Publication
V6.3.0	September 2005	Publication