TSG-RAN Working Group 2 (Radio layer 2 and Radio layer 3)
Berlin, 25th to 28th May 1999

TSGR2#4(99)406

Agenda Item: 9.1

Source: Ericsson

Title: **RLC STATUS PDU Refinement**

Document for: Decision

---

1 Introduction

This contribution proposes a refinement of the contents of the STATUS PDU format for sending status information in RLC acknowledged mode.

2 Discussion

The purpose of having the STATUS PDU is to provide a generic, variable size control PDU format that supports multiple ways of conveying status information such as lists, bitmaps, acknowledgements, and credit value/retransmission window.

One open issue in TS 25.322 is how the flow control function is going to be performed. In the old STAT/USTAT PDUs a credit value was always present to inform about the upper edge of the send window for the transmitter. Currently, the inclusion of a credit value/retransmission window size is stated FFS in [1].

Instead of always having the flow control information present in the STATUS PDU, a new super-field (SUFI) type can be introduced to solve the flow control functionality. This new SUFI type, denoted WINDOW, signals the receiver window size. The state variable VT(MS) shall be updated based on the reception of a STATUS PDU including the super-fields ACK and/or WINDOW. The initial value of the window should either be specified or signalled by RRC when an RLC AM entity is created through the CRLC-CONFIGURE primitive.

For high data rates the sizes of the bitmaps in the STATUS PDU may be very large for some error cases. The compressibility of a bitmap is often high so in order to reduce the size and number of STATUS PDUs needed we propose that a compressed bitmap type should be introduced as a new super-field. Having the compressed bitmap results in higher system capacity because smaller and less STATUS PDUs will be transmitted in the system.

The compression algorithm to be used should be regarded as FFS.

3 Message Formats and State Variables

The following sub-sections propose the changes needed to TS 25.322 in order to refine the STATUS PDU specification.

3.1 Changes to section 9.2

The text describing the STATUS PDU in section 9.2 in [1] should be replaced with the following text:

The format of the STATUS PDU is given in Figure 1 below.

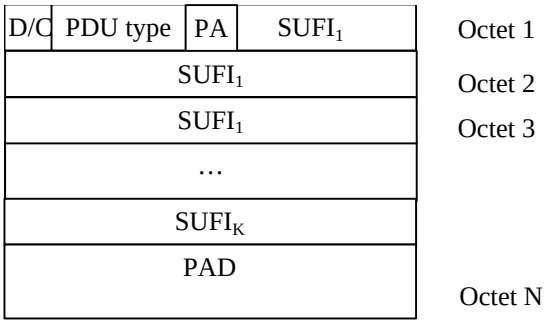

Figure 1. Status Information Control PDU (STATUS PDU)

Up to K different super-fields ($SUFI_1$-$SUFI_K$) can be included into one STATUS PDU. The size of a STATUS PDU is variable and upper bounded by the maximum RLC PDU size used by an RLC entity. Padding shall be included to exactly fit one of the PDU sizes used by the entity.

D/C field: 1 bit
The D/C field indicates the type of an acknowledged mode PDU. It can be either a data or control PDU.

Bit	Description
0	Control PDU
1	Acknowledged mode data PDU

PDU type: 3 bits (FFS)
The PDU type field indicates the control PDU type.
PA: 1 bit
The PA (Poll Answer) field indicates whether the status report is the answer to a poll or not.

Bit	Description
0	The status report is not the answer to a polling request
1	The status report is the answer to a polling request

SUFI: variable number of bits
The SUFI (SUper-FIeld) includes three sub-fields: type information (type of super-field, e.g. list, bitmap, acknowledgement etc), length information (providing the length of a variable length field within the following value field) and a value. Figure 2 shows the structure of the super-field. The size of the type sub-field is non-zero but the size of the other sub-fields may be zero.

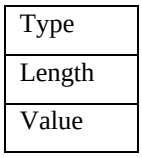

Figure 2. The Structure of a Super-Field

- **Type:** 3 bits (FFS)

Bit	Description
000	No More Data (**NO_MORE**)
001	Window Size (**WINDOW**)
010	Acknowledgement (**ACK**)
011	List (**LIST**)
100	Bitmap (**BITMAP**)
101	Compressed Bitmap (**BITMAP'**)
110	*Reserved for future super-field types*
111	*Reserved for future super-field types*

- **Length:** depending on the super-field type
  Gives the length of the variable size part of the following value field
- **Value:** variable number of bits given by the Type and the Length sub-fields

The No More Data super-field

The 'No More Data' super-field indicates the end of the data part of a STATUS PDU and is shown in Figure 3 below.

Type=**NO_MORE**

Figure 3. The NO_MORE field in a STATUS PDU

The Acknowledgement super-field

The 'Acknowledgement' super-field consists of a type identifier field (ACK) and a sequence number (LSN) as shown in Figure 4 below. The acknowledgement super-field is also indicating the end of the data part of a STATUS PDU. Thus, no 'NO_MORE' super-field is needed in the STATUS PDU when the 'ACK' super-field is present.

Type = **ACK**
LSN

Figure 4. The ACK fields in a STATUS PDU

LSN: 12 bits
Acknowledges the reception of all PUs with sequence numbers < LSN (Last Sequence Number) that are *not* indicated to be erroneous in earlier parts of the STATUS PDU.

The Window Size super-field

The 'Window Size' super-field consists of a type identifier (WINDOW) window size number (WSN) as shown in Figure 1 below.

Type = **WINDOW**
WSN

Figure 5. The WINDOW fields in a STATUS PDU

WSN: 12 bits
The allowed window size to be used by the transmitter.

The List super-field

The List super-field consists of a type identifier field (LIST), a list length field (LENGTH) and a list of LENGTH number of pairs as shown in Figure 6 below.

Type = **LIST**
LENGTH
SN_1
L_1
SN_2
L_2
…

SN_{LENGTH}
L_{LENGTH}

Figure 6. The List fields in a STATUS PDU for a list

LENGTH: 4 bits (FFS)
The number of (SN_i , L_i)-pairs in the super-field of type LIST.
SN_i : 12 bits
Sequence number of PU, which was not correctly received.
L_i : 4 bits (FFS)
Number of consecutive PUs not correctly received following PU with sequence number SN_i.

The Bitmap super-field

The Bitmap super-field consists of a type identifier field (BITMAP), a bitmap length field (LENGTH), a first sequence number (FSN) and a bitmap as shown in Figure 7 below.

Type = **BITMAP**
LENGTH
FSN
bitmap

Figure 7. The Bitmap fields in a STATUS PDU

LENGTH: 4 bits (FFS)
The size of the bitmap in octets (maximum bitmap size: $8*2^{LENGTH}$).
FSN: 12 bits
The sequence number for the first bit in the bitmap.
bitmap: variable number of octets given by LENGTH
Status of the SNs in the interval [FSN, FSN + LENGTH*8 - 1] indicated in the bitmap where each position can have two different values (0 and 1) with the following meaning (bit_position$\in$[0,LENGTH*8 - 1]):
1: SN = (FSN + bit_position) has been correctly received

0: SN = (FSN + bit_position) has not been correctly received

The Compressed Bitmap super-field

The 'Compressed Bitmap' super-field consists of a type identifier field (BITMAP'), a bitmap length field (LENGTH), a first sequence number (FSN) and a bitmap as shown in Figure 7 below.

Type = **BITMAP'**
LENGTH (FFS)
FSN (FFS)
compressed bitmap (FFS)

Figure 8. The BITMAP' fields in a STATUS PDU

LENGTH: 4 bits (FFS)
The size of the compressed bitmap in octets (maximum bitmap size: $8*2^{LENGTH}$ bits).
FSN: 12 bits (FFS)
The sequence number for the first bit in the bitmap.
compressed bitmap: variable or fixed size (FFS)
The exact compression algorithm to be used is FFS.

3.2 Changes to section 9.4

The following changes to section 9.4 should be included in order to

b) VT(A) - Acknowledge state variable
The sequence number of the next in-sequence PU expected to be acknowledged, which forms the lower edge of the window of acceptable acknowledgments. VT(A) is updated based on receipt of a STATUS PDU including an ACK super-field. ~~upon acknowledgment of in-sequence PUs.~~

d) VT(MS) - Maximum Send state variable
The sequence number of the first PU not allowed by the peer receiver [i.e. the receiver will allow up to VT(MS) – 1]. This value represents the upper edge of the transmit window. The transmitter shall not transmit a new PU if VT(S) = VT(MS). VT(MS) is updated based on receipt of a STATUS PDU including an ACK and/or a WINDOW super-field. ~~USTAT PDU, STAT PDU, BGN PDU, BGAK PDU~~.

4 Proposal

We propose that

- the text in chapter 3 replaces the description of STATUS PDU format in section 9.2 in [1].

5 References

[1] 3GPP, TS 25.322 V1.0.0 “Description of the RLC protocol”.