**3GPP TSG-RAN WG2 Meeting #110-e *R2-200xxxx***

**Electronic meeting, 1 – 12 June 2020**

**Agenda item: 6.7.4.2**

**Source: Intel Corporation**

**Title: Report of email discussion [AT110e][046][IIOT] EHC (Intel)**

**Document for: Discussion and Decision**

# Introduction

The contribution is the report of following email discussion.

* [AT110e][046][IIOT] EHC (Intel)

Scope: Treat R2-2005589, determine agreeable parts and make agreements.

Wanted Outcome: Agreements

Deadline: June 5 0700 UTC

#  Discussion

## Max CID parameter

Contribution R2-2004678 [1] proposes to introduce *maxCID-EHC* parameter indicating the maximum number of EHC contexts the UE can establish in uplink for a DRB with the following reasons:

1. To restrict the number of EHC contexts that UE establishes in uplink direction, so that the gNB is able to establish a certain number of EHC contexts in downlink.
2. To restrict the number of EHC contexts that UE establishes for a certain DRB, so that gNB may distribute the overall available context spaces between the different DRBs that require it.

The *maxCID-EHC* parameter is handled in Question 5 of email discussion summary R2-2003834 [9]. In that discussion, some companies indicate the support of signaling of *maxCID-EHC* in addition to the agreed parameter *ehc-CID-Length*, while there were also concerns raised on the introduction of the parameter.

**Question 1:** Please provide your preference on whether to introduce *maxCID-EHC* parameter indicating the maximum number of EHC contexts the UE can establish in uplink for a DRB.

|  |  |  |
| --- | --- | --- |
| **Company** | **Whether to introduce *maxCID-EHC parameter* (Yes/No)** | **Comments** |
| LG | No | In ROHC, the maxCID is used to differentiate different packet formats, i.e. whether there is LARGE CIDs or not. However, in EHC, only one format is defined, and such indicator is not needed. Regardless of the maxCID-EHC, the maximum number of EHC contexts that the UE can support is anyway restricted by the *maxNumberEHC-Contexts*, and thus maxCID-EHC is not needed. |
| Nokia | Yes | We need maxCID to be introduced due to RAN2 agreement that *maxNumberEHC-Contexts* is a sum of contexts supported in DL and UL. If we do not introduce maxCID, then the network has no control on how many EHC contexts the UE establishes in UL. For example, in case the UE supports 2 contexts, then it may establish two contexts in UL and then gNB has no possibility to establish any context in DL. We need to either revisit our previous agreement or agree to introduce maxCID. |
| CATT | No | Referring to 38.331, *maxCID* is per DRB configured for RoHC (including both DL and UL), and the capability parameter *maxNumberROHC-ContextSessions* (across DRBs) is also for both UL and DL. So, in our understanding, both parameters play the same role as *maxCID-EHC* and *maxNumberEHC-Contexts* for EHC. Since there was no problem in handling this commonly for UL and DL in ROHC, we are not sure why it is needed for EHC. |
| Ericsson  | No | No need for extra parameter. UE indicates max number of supported CIDs overall in capability signaling, which is deemed sufficient.  |
| Sony | Yes | We see some value of this parameter to separate the number of CIDs in UL and DL |
| Huawei | Yes | We think the problem discussed in [1] does exist, especially when the number of contexts supported by a UE is just a few.  |
| MediaTek | Yes | Agree with Nokia |
| Futurewei | Yes | Like maxCID for RoHC, maxCID-EHC can be used to configure the maximum number of EHC contexts for a particular DRB.  |
| Qualcomm | Yes | Good point from Sony about separating UL and DL. |
| Intel | No | Our understanding is that gNB has some tools to restrict the number of EHC contexts used in uplink, e.g. by configuring *ehc-CID-Length* to 7 bits. In addition, communications are bidirectional in general, so one EHC context used in UL will generally trigger a related DL communication, and consequently one EHC context will be used in DL. Therefore, it is unlikely that all available EHC contexts that the UE can support will be suddenly used up. |
| OPPO | Yes | Agree with Futurewei |
| vivo | Yes | Setting the restriction on the maximum context in the uplink would help the gNB to establish the number of context in the DL, given that the maximum context capability is shared between uplink and downlink. |

## CID length reconfiguration

ContributionR2-2004678 [1] proposes to consider how to handle the reconfiguration of CID length. Given that RAN2 is in the stage of finalizing Rel-16, we should first discuss whether to allow the reconfiguration of CID length. For ROHC, the reconfiguration of *maxCID* is allowed for PDCP re-establishment case. On the other hand, the reconfiguration of PDCP SN size is not allowed as from condition *Setup2* of IE *pdcp-SN-SizeDL* and *pdcp-SN-SizeUL*: “*This field is mandatory present in case for radio bearer setup for RLC-AM and RLC-UM. Otherwise, this field is absent, Need M.*”

If reconfiguration of CID length is allowed, several issues identified in contribution R2-2004678 [1] need to be addressed. R2-2004678 [1] proposes to allow configuration of *drb-ContinueEHC-DL* and *drb-ContinueEHC-UL* fields for reconfigurations without sync, at least for the case where CID length is reconfigured for an existing EHC configuration. According to “the network reconfigures *ethernetHeaderCompression* only upon reconfiguration involving PDCP re-establishment” from TS 38.331 (in running CR R2-2002703), reconfiguration of CID length in cases other than PDCP re-establishment is not allowed. If CID length is reconfigured in cases other than PDCP re-establishment, it is not clear when the new CID length is applied (i.e. there is no field in EHC header indicating the CID length) due to the lack of synchronized time point as RACH in PDCP re-establishment.

**Question 2:** Please provide your preference regarding the reconfiguration of CID length.

Option a: CID length cannot be reconfigured in any case.

* Field description of *ehc-CID-Length* should be updated to indicate that the CID length cannot be reconfigured, for example, by adding a sentence such as “The value for this field cannot be changed after the initial configuration.”

Option b: CID length can be reconfigured in PDCP re-establishment but cannot be reconfigured in reconfigurations other than PDCP re-establishment.

* There is no change foreseen to TS 38.323 (except for potential changes from Question 4 and 5) or TS 38.331 in option b.

Option c: CID length can be reconfigured in any RRC reconfiguration, including reconfigurations other than PDCP re-establishment.

* A change to TS 38.331 is needed since it currently specifies that “The network reconfigures *ethernetHeaderCompression* only upon reconfiguration involving PDCP re-establishment.” In addition, depending on the discussion outcome of Question 3 below, there might be inter-related changes to TS 38.323 clause 5.1.2 since currently *drb-ContinueEHC-DL* and *drb-ContinueEHC-UL* are only used in PDCP re-establishment.

|  |  |  |
| --- | --- | --- |
| **Company** | **Preference (a/b/c)** | **Comments (including text proposal to show the proposed change, if any)** |
| LG | Option a | We don’t see a need to reconfigure the CID length during the lifetime of the DRB.  |
| Nokia | Option b | We clarified the reasons already in our contribution – it is hard for gNB to predict the number of contexts needed, in advance. For the highest compression benefits it is then required to start with short CID length and modify if needed. It is OK to have it only upon PDCP re-establishment as otherwise there may be issues as clarified by the discussion rapporteur. Such approach would have minimal changes to PDCP to clarify how CIDs are transformed between 7/15 bits long if DRB continue is configured. |
| CATT | b | We agreed in last meeting: “Network reconfigures *ethernetHeaderCompression* only upon reconfiguration involving PDCP re-establishment”. We see no reason to change this agreement. |
| Ericsson | A | We do not see a need for this reconfiguration. As becomes obvious from following questions, options b/c introduce complexity, which we do not see justified.  |
| Sony | A | Agree with LG that CID length stays the same for the lifetime of a DRB |
| Huawei | A | We don’t think the CID length needs to be changed after EHC is configured for a DRB. The principle for IE pdcp-SN-SizeDL and pdcp-SN-SizeUL can be followed. Option c is not preferred due to its technical problem as explained by rapporteur. Option b can work but will incur some additional problems, which is not preferred in this final stage. |
| MediaTek | A | Agree with LG, this adds unnecessary complexity. |
| Futurewei | B or A | Option B allows the reconfiguration of CID length while keeping the DRB (release and add the same DRB with reestablishPDCP being set), and no more change is foreseen for RRC and PDCP.As reconfiguration of CID length don’t occur often, Option A is also acceptable.  |
| Qualcomm | A | No clear use-case for B or C. |
| Intel | A | We share the same view as LG. |
| OPPO | A or B | There is no clear use to reconfigure EHC context. Yet, we think similar principle in RoHC can be reused here and CID reconfiguration is allowed in RoHC upon PDCP re-establishment. Thus, both Option A and B are acceptable to us. |
| vivo | A | We do not think there is a need to reconfigure the CID length during lifetime of DRB, which will bring extra complexity. And more discussion is probably needed to understand the potential issues due to the reconfiguration. |

So far, IE *ethernetHeaderCompression* contains following parameters: *ehc-CID-Length*, *ehc-Downlink, drb-ContinueEHC-DL*, *ehc-Uplink,* and *drb-ContinueEHC-UL*. If option c of Question 2 is agreed, it seems natural that the configuration of *drb-ContinueEHC-DL* and *drb-ContinueEHC-UL* is applicable to the reconfiguration of CID length in reconfiguration other than PDCP re-establishment.

**Question 3:** If your answer to Question 2 is option c, please provide your preference on whether the configuration of *drb-ContinueEHC-DL* and *drb-ContinueEHC-UL* is applicable to the reconfiguration of CID length in RRC reconfiguration other than PDCP re-establishment.

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments (including text proposal to show the proposed change, if any)** |
| Nokia | No | We acknowledge such approach has issues, so it is OK to allow CID length reconfiguration only upon PDCP re-establishment. We should however have a possibility to use DRB continue when CID length is modified. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

R2-2004678 [1] proposes to add clarification regarding how to handle CID (e.g. appending a string of zeros to the CID) when the CID length is reconfigured from 7-bit to 15-bit. TS 38.323 clause 6.3.1 specifies that “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.” Therefore, if CID is considered as an integer, it seems that no further clarification is needed.

**Question 4:** If your answer to Question 2 is option b or c, please provide your preference regarding how to handle CID (e.g. appending a string of zeros to the CID) when the CID length is reconfigured from 7-bit to 15-bit.

Option a: No clarification is needed (e.g. CID is considered as an integer).

Option b: Add clarification to TS 38.323, e.g. transforming 7-bit CID to 15-bit CID by appending a string of 8 zeros to 7-bit CID.

|  |  |  |
| --- | --- | --- |
| **Company** | **Preference (a/b)** | **Comments (including e.g. proposed text)** |
| Nokia | b | We think some simple clarification is needed as CID is usually referred to as to a bit string (e.g. CID = ‘all zeros’). But we could also clarify that CID expressed as an integer remains the same when changing the CID length. |
| CATT | a | CID as an integer is sufficient. |
| Futurewei | Option a | It is free to have. |
| Intel | a | If RAN2 agrees to allow the reconfiguration of CID length, we prefer to treat CID as an integer, considering that it would be desirable to treat CID as an integer as from Question 5 below. We may need to change *CID = "all zeros"* to *CID = 0* in Annex A.2.2.2. |
| OPPO | a | No need to clarify. |
|  |  |  |
|  |  |  |

R2-2004678 [1] proposes to add clarification regarding how to handle EHC contexts (which contexts are kept, e.g. the first 127 contexts are kept or the contexts with CID lower than 128 are kept) when the CID length is reconfigured from 15-bit to 7-bit.

**Question 5:** If your answer to Question 2 is option b or c, please provide your preference regarding how to handle EHC contexts when the CID length is reconfigured from 15-bit to 7-bit.

Option a: No clarification is needed.

Option b: Add clarification to TS 38.323 on which set of EHC contexts are kept.

|  |  |  |
| --- | --- | --- |
| **Company** | **Preference (a/b)** | **Comments (including e.g. proposed text)** |
| Nokia | b | The simplest would be to keep the contexts with CID, expressed in integer, lower than 128. |
| CATT | a | From previous RAN2 agreement, *ethernetHeaderCompression* is only reconfigured with PDCP re-establishment which resets all EHC contexts anyways, so we don’t see any need for handling the old EHC contexts.  |
| Futurewei | Option a | Agree with CATT |
| Intel | b | If RAN2 agrees to allow the reconfiguration of CID length, we agree with Nokia’s view. |
| OPPO | a | Agree with CATT |
|  |  |  |
|  |  |  |

## Decompressor behavior for CID overwriting

Contribution R2-2005154 [6] proposes to adopt a TP capturing the behaviour of decompressor about CID overwriting in TS 38.323.

**Question 6:** Please provide your preference regarding whether and how to update TS 38.323 to capture the behaviour of EHC decompressor about CID overwriting scenario.

Option a: TP proposed in Annex of R2-2005154, with the key change shown below:

|  |
| --- |
| NOTE: If the maximum number of EHC contexts are already established for the compressed flows and a new Ethernet flow does not match any established EHC context, the compressor should associate the new Ethernet flow with one of the EHC CIDs allocated for the existing compressed flows and indicate the association to the decompressor with FH packets or send PDCP SDUs belonging to the Ethernet flow as uncompressed packet. The decompressor should update the existing EHC contexts according to the indicated association. |

Option b: An alternative TP to Annex A.1 is shown below:

|  |
| --- |
| When the EHC decompressor receives the FH packet, the EHC decompressor establishes or updates the EHC context identified by the CID, and transmits the EHC feedback to the EHC compressor to indicate that the EHC context associated with the CID is successfully established or updated in the EHC decompressor. |

Option c: there is no need to update TS 38.323 to capture the behaviour of EHC decompressor about CID overwriting scenario.

|  |  |  |
| --- | --- | --- |
| **Company** | **Preference (a/b/c)** | **Comments** |
| LG | Option c | The NOTE in option a is similar to what we have in ROHC. Even without the clarification in option a, it is obvious that the context is associated with the CID indicated in FH packet.The option b is also not needed because the “establish” covers the case of “update”. |
| Nokia | Option b | We think this is a simple clarification and it is always better to avoid any confusion in specifications. We are not sure whether it is so obvious that establishment covers updating the context as well. |
| CATT | c | We think the current text is clear enough |
| Ericsson  | C | No need to clarify, the term establish is understood as potentially updating . |
| Sony | Option b | Agree with Nokia that “establish” and “update” mean different things and it is a very simple change. |
| Huawei | a or b | In RAN2 #109-e meeting, we have agreed to use a NOTE to specify CID overwriting mechanism in the specification, which shall involve both the compressor and the decompressor. Thus Option a is slightly preferred and option b is acceptable to us. On “establish” and “update”, we understand “establish” usually describes creation of a new context while “update” describes modification of existing context.  |
| MediaTek | Option b | Agree with Nokia |
| Futurewei | A or B | it is good to make it clear, when the required efforts are minimal. |
| Qualcomm | C | Current text seems clear. |
| Intel | Option b | We think option b is straightforward, and it is not clear that “establish” covers “update”: for example, we do have PDCP establishment and re-establishment. |
| OPPO | c | We have no strong view. It seems no issue even if we keep the spec as it is. If majority agree to clarify, we think option b is sufficient. |
| vivo | c | The de-compressor does not need to know whether the compressor performs CID overwriting. When the de-compressor receives a FH packet, it simply establishes the EHC context identified by the CID and perform decompression based on the latest established context. |

## Ethernet frame handling by EHC

Contribution R2-2004679 [2] proposes to adopt a TP regarding EHC compressor operation on Ethernet frame handling. The issue was discussed in RAN2#109bis-e meeting where in email discussion summary R2-2003834 [9], 5 companies preferred to capture it in the informative text (including 1 company which did not have strong view), 7 companies preferred not to capture it, and 4 companies do not have strong view. The issue was postponed to RAN2#110-e meeting due to lack of consensus. Contributions R2-2004962 [4], R2-2005154[6], and R2-2005336[7] propose not to capture operation of different Ethernet header structures as informative text since: 1) the Ethernet header protocol structures are very well defined in IEEE specifications already; 2) there might be potential maintenance work for RAN2 if there is update on Ethernet specifications; 3) how the compressor and decompressor determine the to-be-compressed fields is relevant to UE/Network implementation. R2-2004542 [8] proposes to discuss the issue in this meeting or postpone the decision to Rel-17.

**Question 7:** Please provide your preference on whether to capture an example of operation on the different Ethernet header structures as an informative text (e.g. as shown in the TP of R2-2004679 [2]):

|  |  |  |
| --- | --- | --- |
| **Company** | **Whether to capture an informative text (Yes/No)** | **Comments (including proposed changes to the TP, if any)** |
| LG | No | We want to avoid potential maintenance work in RAN2. |
| Nokia | Yes | This is an example of operation and an informative annex, so its maintenance is not really required. We think this has benefits for implementers as the normative part of EHC description is rather imprecise compared to, e.g. RoHC and may be difficult to interpret for implementers. Also, in case we will support other frame types, then some maintenance work will be needed anyway. It is for example unclear at the moment what EHC compressor/decompressor does with frames other than those indicated in the informative annex proposal, e.g. frames related to FRER protocol. |
| CATT | No | We think the current specification is sufficient. |
| Ericsson | No | Not needed, header structures are clearly defined in IEEE specifications. |
| Sony |  | No strong view  |
| Huawei | No |  |
| MediaTek | Yes | Agree this example will benefit implementers. |
| Futurewei | No strong view | An informative annex can be helpful. |
| Qualcomm | No | Can we update the IIoT TR instead? The text proposal is useful, but does not quite reach the level of inclusion in TS. |
| Intel | No | TS 38.323 specifies “The fields that are compressed by the EHC protocol are: DESTINATION ADDRESS, SOURCE ADDRESS, 802.1Q TAG, and LENGTH/TYPE.” We think this is sufficient and that there is no need to capture in PDCP specification how to determine which Ethernet fields are present, which is well defined in IEEE specifications. If the example is captured, RAN2 needs to maintain it if any update of Ethernet specifications results in change or update of the operation on Ethernet header handling in EHC in future. |
| OPPO | No |  |
| vivo | No strong view | An example would be helpful to provide a better understanding on how EHC processes the Ethernet frame. However, at this point of time, we may not have sufficient time to polish the details of the proposed example. |

## Clarifications

Contribution R2-2004742 [3] proposes to clarify that EHC compressed packet includes both the compressed header packet and full header packet to avoid the confusion caused by the similar names between “EHC compressed packets” and “EHC compressed header packets”.

**Question 8:** Please provide your preference on whether to add clarification that EHC compressed packet includes both the compressed header packet and full header packet (as the 1st change in the TP of R2-2004742 [3]):

|  |  |  |
| --- | --- | --- |
| **Company** | **Whether to add clarification (Yes/No)** | **Comments (including proposed changes to the TP, if any)** |
| LG | No | The “ROHC compressed packet” includes various types of packets including IR packets (which is similar to FH packet in EHC). There is no confusion in “EHC compressed packet” to include FH packets. |
| Nokia | No strong view | We think that it is confusing to refer to FH packets as compressed packets, so at least such clarification should be added. EHC is a standalone protocol, so not everybody will be aware that it reuses some rules from RoHC. It is better to be clear than leave room for interpretations.  |
| CATT | Yes | This indeed brings some clarification. |
| Ericsson | Yes | Clarification needed.  |
| Sony |  | We have no strong view but think that the Compressed header should not refer to a Full header transmission. |
| Huawei | No | No strong view though. It seems we follow the same description for RoHC.  |
| MediaTek |  | Agree with Nokia that it is confusing to refer to FH packets as compressed packets. Perhaps the change in R2-2004742 could be modified as below:*If EHC is configured, the EHC protocol generates two types of output packets:**- EHC ~~compressed~~ packets (including EHC full header packets and EHC compressed header packets), each associated with one PDCP SDU;**- standalone packets not associated with a PDCP SDU, i.e. EHC feedback.* |
| Futurewei |  | To use EHC packet, as MediaTek’s suggested, looks better. |
| Qualcomm | Yes | Good to clarify. |
| Intel | Yes | We think it is good to clarify this aspect, and agree with MediaTek’s suggestion. |
| OPPO |  | We have no strong view. If majority agree to clarify, we think MediaTek’s suggestion looks fine. |
| vivo | Yes | The changes proposed by MediaTek seems acceptable to us. Actually, there is no concept of RoHC compressed packet in ROHC specification. In the PDCP spec, this concept is used for various types of packet (including uncompressed packet) processed by ROHC protocol. As EHC is a standalone protocol, it is better to clarify that the output packets of EHC protocol include EHC full header packets and EHC compressed header packets. |

Contribution R2-2004742 [3] proposes to clarify that “the fields that are compressed” means “removing the fields from the Ethernet packet”. TS 38.323 specifies that “The CH packet includes only the header fields not stored in the EHC context”, so there seems to be no ambiguity regarding the meaning of “the fields that are compressed”.

**Question 9:** Please provide your preference on whether to add clarification regarding the meaning of “the fields that are compressed” (as the 2nd change in the TP of R2-2004742 [3]):

|  |  |  |
| --- | --- | --- |
| **Company** | **Whether to add clarification (Yes/No)** | **Comments (including proposed changes to the TP, if any)** |
| LG | Yes | We think it is a useful clarification. |
| Nokia | Yes | It should be clarified that by compression we mean “removal”. |
| CATT | Yes | Or “compressed” can simply be replaced with “removed” or “stripped” |
| Ericsson | Yes | Can use “remove” when referring to compressed fields of ethernet header, since they are indeed removed in the compressed format. |
| Sony | Yes |  |
| Huawei | Yes | Agree with the 2nd change in the TP of [3]. |
| MediaTek | Yes | Agree with Ericsson |
| Futurewei | Yes |  |
| Qualcomm | Yes | Agree with Ericsson/CATT/Nokia. “removed” is better terminology. |
| Intel | Yes | Agree with CATT/Ericsson and others to use “removed” instead of “compressed”. |
| OPPO | Yes |  |
| vivo | Yes | Different from the ROHC protocol which have more complex compression algorithms, the Ethernet frame is compressed by simply remove some field. Adding some description could make EHC protocol clearer. |

Contribution R2-2004742 [3] proposes to change field name “PAYLOAD (+PAD)” to “PAYLOAD” in Figure A.2.1.1-1 and A.2.1.1-2 of TS 38.323 to avoid the misunderstanding that the fields in Figure A.1-1 and Figure A.2.1.1-1/2 refer to the same content.

**Question 10:** Please provide your preference on whether to change field name “PAYLOAD (+PAD)” to “PAYLOAD” in Figure A.2.1.1-1 and A.2.1.1-2 of TS 38.323 (as the 3rd change in the TP of R2-2004742):

|  |  |  |
| --- | --- | --- |
| **Company** | **Whether to change field name (Yes/No)** | **Comments (including proposed changes to the TP, if any)** |
| LG |  | We don’t think there is any misunderstandings. However, we don’t have strong view on this.  |
| Nokia |  | We think that “PAYLOAD (+PAD)” should be the same in both figures, so the figures are OK. We do not think it is clear from EHC description at the moment that EHC is able to compress frames of types other than those covered by Figure A.1-1. It is also one of the issues we raise in R2-2004679 by Proposal 2: “RAN2 should clarify how EHC handles Ethernet frames which contain fields unrecognizable by EHC.” |
| CATT | No | We think removing “PAD” would add confusion. Alternately, it could be re-named to “Uncompressed header fields + PAYLOAD (+PAD)” |
| Ericsson | Yes | From PDCP point of view, potential payload in the Ethernet payload field is still seen as Ethernet payload, thus no need to mention “PAD”. |
| Huawei |  | No strong view. |
| MediaTek |  | Agree with LG – no strong view on this. |
| Futurewei |  | Either way is fine. |
| Qualcomm | Yes | No strong view however. |
| Intel |  | We don’t think there can be a potential misunderstanding since it is clearly specified in TS 38.323 which Ethernet fields can be compressed. But we don’t have a strong view on this issue. |
| OPPO |  | No strong view |
| vivo | Yes | As the compressor anyway needs to parse each Ethernet fields in order to determine the frame structure, the supporting of the frame type other than those covered by Figure A.1-1 will not add extra complexity. Thus, it is necessary to differentiate the “PAYLOAD (+PAD)” field in Figure A.1-1 and the “PAYLOAD (+PAD)” field in Figure A.2.1.1-2, as the latter one may include the uncompressed Ethernet fields (e.g. LLC/SNAP). |

.

## Switching from compressed header in EHC to full header

Contribution R2-2005147 [5] proposed to consider that the Ethernet header compression allows switching from a compressed header to a full header to avoid the possibility of a decompressor going out of sync (e.g. due to context corruption or error in lower layers in the decompressor) especially considering the ultra-reliability requirements of the type of traffic being carried as a payload. R2-2005147 [5] further proposes to agree on one of the following options:

* Option 1: Use R bit as an indication of NACK
* Option 2: Leave it to the compressor implementation and the compressor may switch between a full header and a compressed header based on implementation (e.g. periodically).

In TS 38.323, compressor can switch from compressed header to full header for CID overwriting scenario. With the current EHC framework, it is not clear how the correct decompressor implementation can go out of sync since “*the EHC compressor keeps transmitting the FH packets until the EHC feedback is received from the EHC decompressor*” (TS 38.323 clause A.1). The issue was discussed in email discussion summary R2-2003834 [9], with the conclusion that “*Decompressor behaviour is unspecified if it receives a compressed packet with an unknown context ID (not much support to specify)*”. It seems that context corruption is due to bugs in decompressor implementation, and in general, error in lower layers cannot be propagated to decompressor due to CRC checking and error PDU discarding in MAC, RLC, and PDCP.

**Question 11:** Please provide your preference on whether there is a need for switching from a compressed header transmission back to a full header transmission after the initial context has been setup, in addition to CID overwriting scenario.

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments (if answering “Yes”, please also indicate preferred solution e.g. use R bit as NACK, or leave to implementation)** |
| LG | No | We are wondering in which case the EHC context is de-synchronized.  |
| Nokia | No | The proposed behaviour may be achieved by EHC compressor implementation and reusing CID overwriting mechanism. We do not think it requires specifications changes. |
| CATT | No | But anyways this is possible with CID overwriting. So we agree with the rapporteur. We see no issue in having compressor switching from CH to FH packets. |
| Ericsson | No | No further enhancement needed.  |
| Sony | Yes | We think that the context desynchronization can happen and the protocol design should be robust enough to handle it.In terms of capturing it, we are fine if the context overwriting mechanism can somehow cover this aspect. The only concern is that the CID overwriting is kicked in when max context has reached. |
| Huawei | No | This issue has been discussed several times. We don’t think anything need to be specified. |
| MediaTek | Yes | We think having such a failsafe mechanism is useful. However, we do recognise that this has been discussed previously. |
| Futurewei | No | Compressor can always use FH packet with the same CID for the context in question. |
| Qualcomm | No | This is an error case. |
| Intel | No | We’re not sure whether we need to address the corruption cases via a standardized solution when this seems to be caused by wrong implementation. For example, “context corruption” means the association between CID and related Ethernet header fields are somehow corrupted at decompressor side. This seems to be a bug in implementation. As our target is the URLLC service, we think that the hardware/software should be tested systematically to avoid such bugs. Similarly, for “error in lower layers”, our understanding is that we have CRC check at physical layer, and MAC/RLC/PDCP needs to check the respective headers. Therefore, it is not clear how such “error in lower layer” can be passed to EHC decompressor. We only have general statements to drop PDUs in error (clause 5.13 of TS 38.321, clause 5.6 of TS 38.322, and clause 5.10 of TS 38.323), but don’t handle specific error/bug cases with standardized solutions. There could be various bugs even beyond our imagination, and it is better not to open a Pandora’s box of fixing bugs with standardized solutions. |
| OPPO | No | We think it is abnormal case. |
| vivo | No | This abnormal case does not need to be specified. |

# Conclusion

**[To be provided at the end of email discussion]**

# References

[1] R2-2004678, Nokia, Nokia Shanghai Bell, “EHC remaining issues”

[2] R2-2004679, Nokia, Nokia Shanghai Bell, “Clarification on Ethernet frame handling by EHC”

[3] R2-2004742, vivo, “Corrections on the EHC”

[4] R2-2004962, Ericsson, “Remaining EHC issues”

[5] R2-2005147, Sony, “Switching from Compressed header in EHC to Full header”

[6] R2-2005154, Huawei, HiSilicon, “Remaining issues about EHC”

[7] R2-2005336, OPPO, “Open issues on EHC”

[8] R2-2004542, III, “Remaining Issues in Ethernet Header Compression”

[9] R2-2003834, Intel, “Report of email discussion [AT109bis-e][030][IIOT] Ethernet Header Compression (Intel)”