**t3GPP TSG-RAN WG2 Meeting #114 R2-210xxxx**

**Online, May 19 – May 27, 2021**

**Source: CATT**

**Title: Summary of Email Discussion 506 – R17 IIOT QoS**

**Agenda Item: 8.5.4**

**Document for: Discussion and Decision**

# Introduction

This contribution provides a summary of the following email discussion:

* [POST113bis-e][506][R17 IIoT] Enhancements based on QoS (CATT)

Scope:

* + - Discuss remaining open issues related to RAN enhancements on new QoS based on inputs submitted to AI 8.5.4.
    - Agreeable Proposals

Deadline: same as submission deadline (as set per RAN2 chair)

Rapporteur would like to set the deadline for companies inputs to 05-10-2021 23:00 UTC to have enough time for preparing the summary report.

# Contact information

|  |  |
| --- | --- |
| Company | Name and email address |
| CATT | Pierre Bertrand; pierrebertrand@catt.cn |
| Ericsson | Zhenhua Zou; zhenhua.zou@ericsson.com |
| Samsung | Milos Tesanovic ; m.tesanovic@samsung.com |
| LG | SeungJune Yi; seungjune.yi@lge.com |
| Intel | Rafia Malik ; rafia.malik@intel.com |
| MediaTek | Pradeep Jose ; pradeep[dot]jose[at]mediatek[dot]com |
| Huawei, HiSilicon | Tao Cai ; tao.cai@huawei.com |
| Nokia | Ping-Heng Wallace Kuo ; Ping-Heng.Kuo@nokia.com |
| Lenovo | Joachim Löhr; jlohr@lenovo.com |
| Qualcomm | Sherif ElAzzouni; selazzou@qti.qualcomm.com |
| CMCC | Min Wu； wumin@chinamobile.com |
| Xiaomi | Yumin Wu ; wuyumin@xiaomi.com |
| OPPO | Zhe Fu ; fuzhe@OPPO.com |
| Futurewei | Yunsong Yang (yyang1@futurewei.com) |
| ZTE | Ting Lu (lu.ting@zte.com.cn) |
| vivo | Wen-ming (ming.wen@vivo.com) |
| III | Grace Liu (graceliu@iii.org.tw) |

# Discussion

As a follow-up of the offline #506 [2], the following agreements on RAN enhancements based on new QoS were achieved in RAN2#113-e [1]:

**Agreements**

- Communication service availability (CSA) is not needed on top of survival time. Send a reply LS to SA2 to notify such confirmation

*-* RAN2 confirms that specification enhancement for survival time support may only needed for uplink. Downlink is addressed by implementation and no specification impacts.

*-* Support for survival time in UCE is up to network configuration.

- Communication service reliability (CSR) is not needed on top of survival time

- Only periodic traffic is considered for survival time work in Rel-17

- RAN2 assumes one application message is conveyed by one PDCP SDU, and may further consider the cases where one application message is conveyed by varying number of PDCP SDUs depending on the progress

Considering the above agreements, and based on the contributions posted at RAN2#113-e [5]-[22], we suggest the following plan for this email discussion:

* Survival Time
  + Survival Time trigger
  + Link reliability increase mechanisms
  + Survival Time management
* Burst spread / BET

## Survival Time

As agreed in RAN2#113bis-e, the relevant Survival Time requirements to consider are those of the Periodic deterministic communication service which are expressed by SA1 in Table 5.2-1 from TS 22.104 [3], copied below for convenience/reference.

TS 22.104 [3] defines Survival Time (ST) as “the time that an application consuming a communication service may continue without an anticipated message.” Figure C.3-1 in Annex C.3 of the TS 22.104 provides an illustration of the survival time.

Moreover, SA2 finalized in SA2#143E the inclusion of the Survival Time parameter for a given TSC flow in the TSCAI in TS23.501 [4] as follows, referring to the definition in TS 22.104:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Survival Time may be provided by TSN AF/AF either in terms of maximum number of messages or in time units, where a time unit is equivalent to the Periodicity. During a single period, single burst is expected. If Survival Time is provided in terms of maximum number of messages, the SMF coverts it into time units by multiplying its value by the Periodicity provided in the TSCAI Container. The SMF corrects the Survival Time in time units by the previously received cumulative rateRatio from the UPF and sets the TSCAI Survival Time to the corrected value.  Table 5.27.2-1 : TSC Assistance Information (TSCAI)   |  |  | | --- | --- | | Assistance Information | Description | | Flow Direction | The direction of the TSC flow (uplink or downlink). | | Periodicity | It refers to the time period between start of two bursts. | | Burst Arrival time (Optional) | The latest possible time when the first packet of the data burst arrives at either the ingress of the RAN (downlink flow direction) or egress interface of the UE (uplink flow direction). | | Survival Time (Optional) | It refers to the time period an application can survive without any burst, as defined in Annex C.2.3, TS 22.104 [105]. | |

Table 5.2-1: Periodic deterministic communication service performance requirements

| Characteristic parameter | | | | Influence quantity | | | | | |  | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Communica­tion service availability: target value (note 1) | Communication service reliability: mean time between failures | End-to-end latency: maximum (note 2) (note 12a) | Service bit rate: user experienced data rate (note 12a) | Message size [byte] (note 12a) | Transfer interval: target value (note 12a) | Survival time (note 12a) | UE  speed (note 13) | # of Ues | Service area  (note 3) | Remarks | |
| 99.999 % to 99.999 99 % | ~ 10 years | < transfer interval value | – | 50 | 500 μs | 500 μs | ≤ 75 km/h | ≤ 20 | 50 m x 10 m x 10 m | Motion control (A.2.2.1) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value | – | 40 | 1 ms | 1 ms | ≤ 75 km/h | ≤ 50 | 50 m x 10 m x 10 m | Motion control (A.2.2.1) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value | – | 20 | 2 ms | 2 ms | ≤ 75 km/h | ≤ 100 | 50 m x 10 m x 10 m | Motion control (A.2.2.1) | |
| 99.999 9 % | – | < 5 ms | 1 kbit/s (steady state) 1.5 Mbit/s (fault case) | < 1,500 | < 60 s  (steady state) ≥ 1 ms (fault case) | transfer interval | stationary | 20 | 30 km x 20 km | Electrical Distribution – Dis­ribute automated switch­ing for isolation and service restoration (A.4.4); (note 5) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value |  | 1 k | ≤ 10 ms | 10 ms | - | 5 to 10 | 100 m x 30 m x 10 m | Control-to-control in motion control (A.2.2.2); (note 9) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value (note 5) | 50 Mbit/s |  | ≤ 1 ms | 3 x transfer interval | stationary | 2 to 5 | 100 m x 30 m x 10 m | Wired-2-wireless 100 Mbit/s link replacement (A.2.2.4) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value (note 5) | 250 Mbit/s |  | ≤ 1 ms | 3 x transfer interval | stationary | 2 to 5 | 100 m x  30 m x 10 m | Wired-2-wireless 1 Gbit/s link replacement (A.2.2.4) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < transfer interval value |  | 1 k | ≤ 50 ms | 50 ms | - | 5 to 10 | 1,000 m x 30 m x 10 m | Control-to-control in motion control (A.2.2.2); (note 9) | |
| > 99.999 9 % | ~ 10 years | < transfer interval value | – | 40 to 250 | 1 ms to 50 ms (note 6) (note 7) | transfer interval value | ≤ 50 km/h | ≤ 100 | ≤ 1 km2 | Mobile robots (A.2.2.3) | |
| 99.999 9 % to 99.999 999 % | ~ 1 month | < transfer interval value | – | 40 to 250 | 4 ms to 8 ms (note 7) | transfer interval value | < 8 km/h (linear movement) | TBD | 50 m x 10 m x 4 m | Mobile control panels – remote control of e.g. assembly robots, milling machines (A.2.4.1); (note 9) | |
| 99.999 999 % | 1 day | < 8 ms  (note 14) | 250 kbit/s | 40 to 250 | 8 ms | 16 ms | quasi-static; up to 10 km/h | 2 or more | 30 m x 30 m | Mobile Opera­tion Panel: Emer­gency stop (connectivity availability) (A.2.4.1A) | |
| 99.999 99 % | 1 day | < 10 ms  (note 14) | < 1 Mbit/s | <1024 | 10 ms | ~10 ms | quasi-static; up to 10 km/h | 2 or more | 30 m x 30 m | Mobile Operation Panel: Safety data stream (A.2.4.1A) | |
| 99.999 999 % | 1 day | 10 ms to 100 ms  (note 14) | 10 kbit/s | 10 to 100 | 10 ms to 100 ms | transfer interval | stationary | 2 or more | 100 m² to 2,000 m² | Mobile Operation Panel: Control to visualization (A.2.4.1A) | |
| 99.999 999 % | 1 day | < 1 ms  (note 14) | 12 Mbit/s to 16 Mbit/s | 10 to 100 | 1 ms | ~ 1 ms | stationary | 2 or more | 100 m² | Mobile Operation Panel: Motion control (A.2.4.1A) | |
| 99.999 999 % | 1 day | < 2 ms  (note 14) | 16 kbit/s (UL) 2 Mbit/s (DL) | 50 | 2 ms | ~ 2 ms | stationary | 2 or more | 100 m² | Mobile Operation Panel: Haptic feedback data stream (A.2.4.1A) | |
| 99.999 9 % to 99.999 999 % | ~ 1 year | < transfer interval | – | 40 to 250 | < 12 ms (note 7) | 12 ms | < 8 km/h (linear movement) | TBD | typically 40 m x 60 m; maximum 200 m x 300 m | Mobile control panels –remote control of e.g. mobile cranes, mobile pumps, fixed portal cranes (A.2.4.1); (note 9) | |
| 99.999 9 % to 99.999 999 % | ≥ 1 year | < transfer interval value | – | 20 | ≥ 10 ms (note 8) | 0 | typically stationary | typically 10 to 20 | typically ≤ 100 m x 100 m x 50 m | Process automation – closed loop control (A.2.3.1) | |
| 99.999 % | TBD | ~ 50 ms | – | ~ 100 | ~ 50 ms | TBD | stationary | ≤ 100,000 | several km2 up to 100,000 km2 | Primary frequency control (A.4.2); (note 9) | |
| 99.999 % | TBD | ~ 100 ms | – | ~ 100 | ~ 200 ms | TBD | stationary | ≤ 100,000 | several km2 up to 100,000 km2 | Distributed Voltage Control (A.4.3) (note 9) | |
| > 99.999 9 % | ~ 1 year | < transfer interval value | – | 15 k to 250 k | 10 ms to 100 ms (note 7) | transfer interval value | ≤ 50 km/h | ≤ 100 | ≤ 1 km2 | Mobile robots – video-operated remote control (A.2.2.3) | |
| > 99.999 9 % | ~ 1 year | < transfer interval value | – | 40 to 250 | 40 ms to 500 ms (note 7) | transfer interval value | ≤ 50 km/h | ≤ 100 | ≤ 1 km2 | Mobile robots (A.2.2.3) | |
| 99.99 % | ≥ 1 week | < transfer interval value | – | 20 to 255 | 100 ms to 60 s (note 7) | ≥ 3 x transfer interval value | typically stationary | ≤ 10,000 to 100,000 | ≤ 10 km x 10 km x 50 m | Plant asset management (A.2.3.3) | |
| >99.999 999 % | > 10 years | < 2 ms | 2 Mbit/s to 16 Mbit/s | 250 to 2,000 | 1 ms | transfer interval value | stationary | 1 | < 100 m2 | Robotic Aided Surgery (A.6.2) | |
| >99.999 9 % | > 1 year | < 20 ms | 2 Mbit/s to 16 Mbit/s | 250 to 2,000 | 1 ms | transfer interval value | stationary | 2 per 1,000 km2 | < 400 km (note 12) | Robotic Aided Surgery (A.6.2) | |
| >99.999 % | >> 1 month  (< 1 year) | < 20 ms | 2 Mbit/s to 16 Mbit/s | 80 | 1 ms | transfer interval value | stationary | 20 per 100 km2 | < 50 km (note 12) | Robotic Aided Diagnosis (A.6.3) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < 0.5 x transfer interval | 2.5 Mbit/s | 250 500 with localisa­tion informa­tion | > 5 ms > 2.5 ms > 1.7 ms (note 10) | 0 transfer interval 2 x transfer interval (note 10) | ≤ 6 km/h (linear movement) | 2 to 8 | 10 m x 10 m x 5 m; 50  m x 5 m x 5 m (note 11) | Cooperative carrying – fragile work pieces; (ProSe communication) (A.2.2.5) | |
| 99.999 9 % to 99.999 999 % | ~ 10 years | < 0.5 x transfer interval | 2.5 Mbit/s | 250 500 with localisa­tion informa­tion | > 5 ms  > 2.5 ms > 1.7 ms (note 10) | 0 transfer interval 2 x transfer interval (note 10) | ≤ 12 km/h (linear movement) | 2 to 8 | 10 m x 10 m x 5 m; 50 m x 5 m x 5 m (note 11) | Cooperative carrying – elastic work pieces; (ProSe communication) (A.2.2.5) | |
| NOTE 1: One or more retransmissions of network layer packets may take place in order to satisfy the communication service availability requirement.  NOTE 2: Unless otherwise specified, all communication includes 1 wireless link (UE to network node or network node to UE) rather than two wireless links (UE to UE).  NOTE 3: Length x width (x height).  NOTE 4: (void)  NOTE 5: Communication includes two wireless links (UE to UE).  NOTE 6: This covers different transfer intervals for different similar use cases with target values of 1 ms, 1 ms to 10 ms, and 10 ms to 50 ms.  NOTE 7: The transfer interval deviates around its target value by < ±25 %.  NOTE 8: The transfer interval deviates around its target value by < ±5 %.  NOTE 9: Communication may include two wireless links (UE to UE).  NOTE 10: The first value is the application requirement, the other values are the requirement with multiple transmission of the same information (two or three times, respectively).  NOTE 11: Service Area for direct communication between Ues. The group of Ues with direct communication might move throughout the whole factory site (up to several km²).  NOTE 12: Maximum straight-line distance between Ues.  NOTE 12a: It applies to both UL and DL unless stated otherwise.  NOTE 13: It applies to both linear movement and rotation unless stated otherwise.  NOTE 14: The mobile operation panel is connected wirelessly to the 5G system. If the mobile robot/production line is also connected wirelessly to the 5G system, the communication includes two wireless links. | | | | | | | | | | |

Therefore, from RAN perspective, it should be a common understanding that, when a service flow, configured with Survival Time, enters Survival Time, RAN action should be to improve the associated link reliability to make sure any subsequent message(s) can be delivered successfully before the Survival Time is violated. It is therefore proposed to confirm this baseline understanding.

**Q1: Do you agree with the basic principle that when a service flow, configured with survival time, enters survival time, RAN action (gNB and/or UE) should be to improve the associated link reliability to make sure any subsequent message(s) can be delivered successfully before the survival time is violated?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | Yes, with comments | Survival time information is optional, and it is not required that the network must use it, e.g., monitor the survival time and “improve” the link quality subsequently. But we agree that it is how it can be used. One additional comment is that, once survival time mode is entered we cannot “make sure” the message is delivered before survival time expiry, we can only allow the option of “improving the associated link reliability” to meet its service level requirement, e.g., communication service availability, communication service reliability and etc.  A more precise wording can be that:  When a survival time information is provided in TSC AI, RAN action (gNB and/or UE) can utilize it to monitor survival time and, when entering survival time, improve the associated link reliability so that any subsequent message(s) can be delivered to meet its service level requirement. |
| Samsung | Yes but… | Improving link reliability is indeed the most common use case. However, we feel that the opposite (relaxing reliability or at the very least doing nothing) is also a possibility, resulting in higher flexibility and covering a wider range of use cases for ST. We do not think we have to limit the use of ST to improving reliability, even though we concede it is the most common use case. |
| LG | No | The question seems to propose a reactive method, i.e. RAN action (to improve reliability) is required only after a service flow enters ST. However, the reliability improvement can be achieved proactively, or even by the network configuration from the radio bearer setup based on the ST information.  In addition, we agree with Ericsson that we cannot “make sure” the message is delivered before survival time expiry, but we can only allow the option of “improving the associated link reliability” to meet its service level requirement.  Thus, what we can agree is:  When a survival time information is provided in TSC AI, RAN action (gNB and/or UE) can utilize it to improve the associated link reliability so that the survival time requirement is met. |
| Intel | Yes |  |
| MediaTek | Yes, but | While we agree that one potential action on entering Survival time is to improve link reliability, we do not need to limit the use of Survival time to just this option. We agree with Ericsson, Samsung and LG that Survival time could be used in different ways by different NW implementations. |
| Huawei, HiSilicon | Yes | In order to guarantee the service availability of an application with survival time requirement, RAN shall improve its transmission reliability if the application enters survival time, e.g. a packet is lost or delayed during Uu transmission.  We suggest that RAN2 to focus on the more challenging scenario where the transmission reliability needs to be improved, rather than the scenario where the reliability requirement could be relaxed, e.g. with a long ST.  Further, we also suggest that RAN2 to focus on the “reactive” approach as the network can take preemptive measures to boost the transmission reliability, however this will bring cost on resource utilization efficiency and the network cannot rely solely upon the proactive measures. The network will always need to use reactive approach for certain services and/or under certain system conditions. |
| CATT | Yes | Also OK to improve the exact wording. And agree with Ericsson about “make sure”. |
| Nokia | Yes | This is the main objective – how to facilitate survival time violation avoidance of the application from RAN perspective. Agree with companies above that the wording “make sure” in the question is a bit too strong, but at least we think RAN should try its best to avoid consecutive message loss that leads to application failure, because application failure due to survival time violation is really unacceptable in many use cases. |
| Lenovo | Yes, but | We agree that increasing the link reliability when entering the ST mode seems to the most common use case. However we should not restrict this to the case where ST mode has been already entered. Another strategy might be to even avoid entering ST mode, e.g. by improving reliability for cases when HARQ NACK is received (reliability of retransmissions are improved) |
| Qualcomm | Yes | Agree, with the general reservation Samsung raises that survival time can have more uses than temporarily boosting reliability. |
| CMCC | Yes | The RAN needs to monitor the status of message and start the survival time monitoring to perform the enhancement mechanism for uplink during the survival time. |
| Xiaomi | Yes |  |
| OPPO | Yes | Normally, the link reliability should be improved when entering the ST state. We can focus on this principle at this stage. |
| Futurewei | Yes, but | LGE’s simplified version is better. |
| ZTE | Yes | We have similar view as Huawei. |
| vivo | Yes | Based on our understanding, there are two ways to utilize the survival time information:   1. Use the relaxed scheme (i.e. transmission scheme with lower reliability) first. When enters survival time, RAN choose enhanced scheme (i.e. transmission scheme with higher reliability), which is aligned with the operation mentioned in the Q1; 2. Use the enhanced scheme (i.e. transmission scheme with higher reliability) first. Still it is possible that application enters survival time. At such case, RAN chooses relaxed scheme (i.e. transmission scheme with lower reliability);   The second way (2) is a method to prevent a QoS flow from entering survival time in advance, which is at the expensive of large resource cost. Considering that entering survival time is not a frequent thing, the way of (1) is considered to be preferable as it covers more uses cases. |
| III | Yes |  |

### Survival Time Trigger

From the contributions posted at RAN2#113bis-e, the proposed solutions for triggering Survival Time “state” can be classified into three main categories:

1. Left to gNB implementation only, so that specification enhancements for UE-based solutions are not needed: [14][17][19][20]
2. UE-based with proactive trigger: UE triggers ST based on Sequence Number i.e. boosts reliability proactively every N packets [5][7]
3. UE-based with reactive trigger: UE triggers ST based on Tx-side timer or Tx-side counting of transmission failures, and autonomously applies a more reliable pre-configured UL transmission [5][6][8][9][11][12][15][16][17][18][21][22]

We will address each category in the following.

#### Survival Time handling left to gNB implementation only

Clearly this question needs to be sorted out first, because if ST handling can be fully addressed by gNB implementation, no further work is expected in RAN2 for this topic.

This proposal essentially relies on the observation that, even for the most stringent TSN flows of above Table 5.2-1 (ST = 0.5ms), [14] “*for UL traffic, the reception status is known at the gNB first and so there is no faster and more reliable way to indicate to the UL transmission reception status to the UE than gNB simply dynamically scheduling the UE*”. “*Upon observing this a gNB can schedule the subsequent packet with higher reliability to help ensure the survival time is not violated, such as, sending a (re)-activation command for UL CG or a dynamic uplink grant with a more robust MCS, or even activating PDCP duplication.*”

On the other hand, [6] specifically assesses this solution with a quantitative analysis of the reaction time for the most stringent usecase of Table 5.2-1 (ST = 0.5ms), which confirms that, as proposed in [14], gNB has enough time to provide either a (re)-activation command for UL CG or a dynamic uplink grant with a more robust MCS, but still, [6] lists the following drawbacks associated with such a solution for such usecase:

* this only leaves CG type 2 reconfiguration as solution for improving the reliability and rules out other solutions e.g. duplication activation via MAC CE
* CG type 2 reconfiguration can only play with MCS for improving the reliability, but that may not be sufficient to address deep link quality decrease due to e.g. beam blockage (which is why duplication was designed for NR in first place)
* it can only work with Type 2 CGs, not Type 1 CGs
* in this case gNB must schedule the retransmission grant *after* reconfiguring the CG, since the two commands are addressed to CS-RNTI with NDI = 1 and 0 respectively, hence must be serialized.
* this requires configuring at least 3 HARQ processes for this CG configuration, although only 2 are required in legacy implementation (see [6] Annex 2).

In addition, [15] brings the following concern with leaving ST handling all to gNB implementation: “*We feel that it is essential that a basic framework for ST handling is standardized. While entering the ST state is not expected to be a frequent event, when it does happen it is absolutely crucial that the procedure is executed in a standardized, verifiable manner due to requirements of applications for which it will most often be configured*”.

In summary, it is obvious that gNB has full freedom to use legacy scheduling and reconfiguration mechanisms to upgrade the radio resources targeted for an UL flow whenever it estimates it is needed. Therefore, if it can be considered that gNB implementation can be sufficient to address *some* of the usecases in the above Table 5.2-1, the key point to clarify is whether it is sufficient to handle all usecases of Table 5.2-1.

**Q2: Do you agree that handling of Survival Time in RAN can be all left to gNB implementation only so that specification enhancements for UE-based solutions are not needed?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | Yes | Comments to the drawbacks listed in the paper [14][6]   1. There is no requirement whatsoever to deliver the already “failed” message that triggering the survival time mode, and what it matters is the subsequent message. In other words, there is no need for the gNB to schedule two serialized DCI commands or use three HARQ processes.   [CATT] Correct, but precluding it is not quite satisfactory either. Our view is that the solution we select to handle ST should leave full freedom to NW to adopt one or the other strategy for the failed packet. Note that in Ericsson’s contribution [14] it reads: “*However, for the delay critical GBR resource type, packets delayed more than the PDB can be discarded or delivered depending on local decision [2]. For a QoS flow provided with survival time, a smart implementation should still deliver the packet up to the survival time even if after the PDB has been reached*”.  [Ericsson] The above yellow highlighted text, in Ericsson’s contribution, applies to the subsequent message, not the “failed” message. This is to illustrate the problem if start/stop of the survival time is not well defined. In my view, there is no requirement to salvage the “failed” message according to the stringent URLLC QoS, e.g., if needed, it can be retransmitted by the higher layer. Recall that the transmission resource of this “failed” message would compete for the transmission resource of the subsequent message, which would be more restrictive to meet the survival time requirement. We don’t think it is necessary to specify enhancements for un-identified requirements.   1. We acknowledge that network implementation cannot work with CG type 1, but CG type 1 was introduced in Rel-15 to reduce the latency when the network initially configures and activates the Configured Grant. If the latency in activation is an issue, network would obviously use CG type 2 in this case. In other words, it is quite strange to enhance a feature that was not intended for this use case and for which there is a clear better alternative (CG type 2).   [CATT] In our understanding, CG type 2 was introduced to address non-deterministic type of traffic, for example, a traffic with active/non-active periods e.g. VoIP. For such traffic, the possibility for dynamic activation/deactivation is indeed attractive. On the contrary ST is defined for periodic deterministic traffic which is expected to be active continuously for a long period of time, hence a better match seems CG Type 1. We agree though that configuring ST traffic with CG Type 2 is always possible and may not be considered as such a big issue.   1. There is a network implementation solution to activate PDCP duplication for survival time by sending a DCI command. Network can configure and activate PDCP duplication in a second cell, and the duplicate RLC entity on this cell is restricted by LCP restriction configuration to be sent on a configured but de-activated CG. The PDCP discard timer is set equal to the PDB. The network can send a DCI activation command to activate this CG if it sees the need. Since the PDCP discard timer is set equal to PDB, when the CG is activated, only the latest PDCP duplicate packet from the UE will be transmitted.   [CATT] This is a very strange way to activate duplication. Same as above, one first observation is that setting the PDCP discard timer to the PDB disallows retransmitting the failed packet which we still believe is a lack of flexibility. Moreover, this solution is inaccurate synchronization-wise. Indeed, NW cannot know when UE exactly started the discard timer so it is possible that NW activates the leg while the PDCP discard timer of the failed packet is still running, in which case the failed packet will fly first on the new leg and the next one will be delayed which has a high risk to violate ST. So we think this solution is both inflexible and not secure.  [Ericsson] We have the same comment as above that there is no need to specify enhancements for an un-identified requirement. At the reception of a PDCP SDU from upper layers, the transmitting PDCP entity shall, start the discardTimer associated with this PDCP SDU. gNB would learn by observation of UL transmissions or from the Burst Arrival Time that the time the TSC packet arrives at the UE side. In the example CATT mentioned, the network implementation would have already considered uncertainties in UEs (which should be very small for this tight TSC requirement), otherwise there is no way to configure CGs to match traffic arrival time. In the worst case, the network can slightly “delay“ transmission of the activation DCI to avoid this error situation.   1. For the comment in [15], the standardization is to provide essential features for inter-operability between different UE and network vendors. It is not to standardize the network behaviour.   [CATT] Agree that standardization mainly focuses on UE behavior, and that is precisely the point here: when a technical analysis shows that a system requirement cannot be handled by NW implementation alone, a UE behavior must be standardized accordingly.  The bottom line here is, for medium and long survival time, it can be up-to gNB implementation and what is contentious is the case for short survival time (e.g., 0.5 millisecond). In this case, our argument is, for short survival time, the UE-based reactive triggering is no better than gNB implementation, since the feedback is given by gNB. More importantly, we see there are some fundamental technical flaws in all those UE-based reactive triggering solutions, see more inputs below. For the UE-based proactive triggering, it can also be implemented by a network implementation solution, see details below.  Last but not the least, it is not required that the network must use the survival time mechanism to deal with this very stringent survival time. gNB is the first to know when an UL transmission expected within the context of a CG has failed and should be responsible for starting the ST timer, determining what to do (if anything) after starting ST timer and what to do (if anything) when ST timer expires. Any use case for which short periodicity precludes the ability of a gNB to dynamically react to detecting an UL transmission failure (by allocating additional UL resources for UE re-transmission before ST expiry) can also be left up to gNB implementation i.e. a gNB should be able to determine that there are CG type applications for which satisfying PER alone is sufficient (i.e. the gNB does nothing upon detecting ST expiry). |
| Samsung | No | We do agree with most of the assessment in [6] and reiterate our view that – if ST handling is left to network implementation – gNB may not be able to react fast enough in many cases where ST use is at its most critical, and/or gNB’s arsenal of techniques may be limited (as already pointed out in [6]). Additionally, leaving ST handling fully to NW implementation could significantly increase amount of network signaling for e.g. dynamic change of MCS level or reconfiguration, and it some cases dynamic control may not be possible at all. |
| LG | Yes | For medium and long ST, the network has enough time to control the UE’s UL transmission, and there should be no problem.  For short ST, the network can configure the radio bearer with high reliability, e.g. PDCP duplication, from the beginning. We think no method can guarantee very short ST, e.g. <= 1ms, other than configuring the radio bearer with high reliability. |
| Intel | See comment | Having both gNB implementation and UE based solution offers flexibility |
| MediaTek | Yes | Our base assumption should be to leave control of reliability to the NW, and we should only introduce changes when the NW implementation does not work.  On the drawbacks listed:  1. If the duplicated leg is mapped to a Type 2 CG as outlined by Ericsson, activation and deactivation of the CG can be used to control transmission of duplicate packets, i.e. reliability (both MCS and duplication) can be controlled by the NW.  [CATT] See answer to Ericsson  2. Type 1 CGs should not be considered here as they were not intended to address the case where reliability of the link needs to be dynamically controlled.  [CATT] See answer to Ericsson  3. Survival time is not meant to improve retransmission reliability, but rather subsequent transmission reliability. Therefore the issue of serialized DCI commands and the requirement of 3 HARQ processes do not apply here.  [CATT] It does if the gNB implementation chooses to both providing a retransmission grant AND acting with another DCI to update the CG type 2 resource. Key point is to allow a solution where the network has the freedom to retrieve or abandon the failed message while increasing the reliability for the next one.  Considering the above, we fail to see an issue with leaving the use of survival time to NW implementation. |
| Huawei, HiSilicon | No | For some IIoT scenarios, gNB implementation alone might be enough, e.g. for services with not very high service availability. However it is not always sufficient and efficient. It is tricky for the network to determine the “break point” between short ST and medium/long ST. If gNB implementation only based solution is adopted, the network has to proactively make configuration with high reliability for short ST and some medium ST cases, which will inevitably lead to waste of system resource, considering ST state would be low probability event.  PDCP duplication can achieve extremely high transmission reliability, then for the proactive gNB implementation based solution, it is to keep PDCP duplication being activated all the time, which is not resource efficient since most packets’ transmission via single link would be sufficiently reliable.  For uplink transmission, if the gNB is aware of the packet loss and that the application may enter survival time, it may react with activating PDCP duplication for the next packet transmission. However, as mentioned by companies, activating PDCP duplication via MAC CE when packet loss is detected by the gNB would not be timely. |
| CATT | No | See answers to Ericsson and MediaTek above. Essentially, we have a fundamental disagreement on “the UE-based reactive triggering is no better than gNB implementation, since the feedback is given by gNB”. As elaborated in [6] UE-based trigger offers more flexibility regarding the method for increasing the reliability, guarantees timely handling of the failed packet while increasing the reliability of the next packet, and does not consume any additional HARQ process compared with a legacy implementation of such usecase (periodic deterministic with 0.5ms period). |
| Nokia | No | We think that specification enhancements are needed especially for uplink as we have agreed earlier in RAN2, but we are not sure what “UE-based” solution means here in the question. From our point of view, specification change is needed mainly for use cases with very stringent requirements, but not necessarily a “UE-based” solution. |
| Lenovo | No | We agree with the analysis in [6] that for the most stringent use cases, there are limitations for the gNB-based solution, i.e. ST only handled by gNB scheduling/implementation. We think that a UE-based solution has some benefits over a gNB-based solution, as a UE-based solution is well specified and behaviour is predictable. Furthermore, UE could for example apply some smart pre-processing technique, e.g. pre-process the RLC PDUs in advance, in order to meet the processing timing constraints. |
| Qualcomm | No | We agree that for longer survival time constraints it can be left to gNB implementation to enhance the link reliability. For shorter survival time however, there is no sufficient time for the gNB to detect start of survival state AND take actions (e.g., turn on PDCP duplication via MAC CE signaled to the UE). The exact response time of gNB would depend on implementation for generating MAC CE and UE processing time, but our estimation is that the gNB-based solution would not be effective for survival time less than 5ms (which covers a lot of the required use cases). Furthermore, we disagree with the statement that “the UE-based reactive triggering is no better than gNB implementation”. An autonomous duplication solution would have a shorter response time than a gNB duplication via MAC CE, thus it is better for short ST use cases.  [Ericsson] The question is how the UE knows that the transmission was not successful. Isn’t it that gNB knows this information first, since it is the receiver?  The solution proposed by Ericsson to activate PDCP duplication via DCI to circumvent the MAC CE latency relies on a cumbersome configuration, we would much better prefer standardizing a proper solution.  [Ericsson] There is only “allowed” versus “not-allowed” configuration. No matter how cumbersome it might seem like, it is “allowed” and spec does not preclude any “allowed” configuration.  We also note that the autonomous UE reactive solutions would still be configured by the NW, e.g., gNB can configure the UE with the criteria to enter a survival state, or the NW can choose not to configure that UE behavior and rely on implementation, however, the presence of a standardized solution is crucial for a proper industrial application operation. |
| Xiaomi | No | According to the survival time definition provided by SA2 LS “R2-2010692 LS on Use of Survival Time for Deterministic Applications in 5GS”, the survival time is considered as “as a maximum time in units of “time” where each unit corresponds to the data burst periodicity defined in TSCAI in Rel-16”  The survival time requirements can be divided into the following two categories:   * Cat1 (Survival time = Transfer interval): No PDCP PDU transmission failure is allowed. Most stringent survival time requirement is 0.5ms. * Cat2 (Survival time = n \* Transfer interval, n>=2): Some PDCP PDU transmission failure are allowed. Most stringent survival requirement is 3ms with transfer interval=1ms.   Considering the processing delay (i.e. 5ms for MAC CE and 10ms for RRCReconfig) of the UE and the gNB, using the MAC CE or the RRC message to change the scheduling strategy (e.g. activate PDCP duplication or reconfigure CG type-1) would not work for both of the above two survival time Categories.  Using only the reactivation DCI of CG type-2 to fulfill the survival time requirement seems limiting the implementation flexibility of both the gNB and the UE. For example, PDCP duplication may not be always activated to avoid the redundant transmission, and the CG type -1 could be configured instead of CG type-2, to avoid the activation delay or to improve the spectrum efficiency by not using the extra activation DCI. |
| OPPO | No | In our understanding, UE-based solution is useful for the service with more stringent latency requirement or a shorter survival time value. For this case, there is no enough time left to gNB for react or modify resource transmission scheme, and UE-based solution can be a supplementary to improve link reliability and fulfil survival time requirement. |
| Futurewei |  | Yes for medium to long survival time. FFS for 0 to very short survival time. |
| ZTE | No | We generally agree with above comments that survival time is not meant to improve retransmission reliability, but rather subsequent transmission reliability.  If Survival Time trigging is left to gNB implementation only, we see two possible ways:   * The first way is that there would be “no” survival time state at all, e.g., the network may have to (always) make configuration with high reliability for the use cases provided with survival time requirements. For this way, we have sympathy with HW’s comments, e.g., firstly, it may be difficult for the network to determine the “break point” between short ST and medium/long ST. Therefore, the network may have to provide high reliability configuration for all ST use cases. Moreover, as entering ST state and requesting higher reliability transmission would be rare case, in most other cases the configuration with high reliability would be unnecessary and wasteful. * The second way is that gNB also detect survival time trigger and can only react after entering ST state. We see the following flaws in gNB side that may even make gNB unable to aware the need to enter the ST state:   + In uplink transmission, even it’s possible for gNB to detect loss or failure on packet transmission based on the knowledge of granted resources, as there may be several different types (periodic or aperiodic, deterministic or non-deterministic) services sent on UE side, it may still be not easy for gNB to know whether packet loss/failure on certain granted resources belong to a certain periodic deterministic service. Then the gNB may not be able to accurately trigger survival time for a certain periodic deterministic service.   + In uplink transmission, there might be another “empty packet” case, e.g., there is an expected UL packet but it’s not really received by the UE at the burst arrival time. The gNB cannot accurately identify when empty packet happens while the UE can.   In a summary, gNB implementation-based Survival Time handling has issue of resources waste or may be unable to accurately trigger Survival Time due to mismatch between packet loss on the granted resources and the service. So we don’t prefer such trigger.  Moreover, we agree with some above concerns for Link reliability increase scheme based on gNB implementation, e.g., activating PDCP duplication via MAC CE when packet loss is detected by the gNB would not be timely.  We also agree with CATT that it’s not true the UE-based reactive triggering is no better than gNB implementation as we don’t think UE-based reactive triggering always needs to rely on the feedback from gNB. We agree that UE-based trigger can guarantee timely handling of the failed packet while increasing the reliability of the next packet with some ways, e.g., a packet transmission failure can be confirmed if an AN PDB is exceeded. In the most stringent case (e.g., periodic deterministic with 0.5ms period), UE can also immediately trigger reliability enhancement of the subsequent link transmission when entering ST state. The details can be seen in the comments for Q4-a, Q8 etc. |
| vivo | No | For short ST, MAC CE and RRC reconfiguration cannot be used to adjust transmission scheme as their processing time is much longer than ST. Even if DCI based solution is used to adjust transmission scheme, gNB implementation only based solution is still not efficient. The reason is that gNB anyway needs to provide UE with scheduling information for (re)transmission, which can be used by UE to determine whether there is a need to enter survival time and improve transmission scheme autonomously. If gNB based signaling is adopted, the additional signaling overhead is inevitable. |
| III |  | Additional simulation to prove the gNB implementation-based solution can handle the most stringent survival time constrain is helpful for further decision. |

#### UE-based with proactive trigger

As described in [7], in this approach, UE “*does not rely on any kind of feedback (e.g. new dynamic grant, re-transmission grant, or ARQ NACK) from the receiver (gNB), but the UE proactively boost the reliability of at least one burst in every N-th incoming burst to make sure consecutive error of N burst does not occur. The PDCP layer may directly determine how to deal with a incoming packet based on its sequence number (SN)*”.

Specifically, [7] elaborates on the benefits of such approach compared with *reactive* triggers for which associate issues are listed in observations 1-3 as follows:

* Relying on feedback (e.g. HARQ feedback or uplink grant) for survival time support would end up a very restrictive solution which basically precludes possibility of flexible TDD, and any small jitter in the end-to-end path would make this solution useless due to extremely tight time budget.
* If PDCP duplication is to be used as the mean to boost reliability, relying on feedback (e.g. HARQ feedback or uplink grant) for survival time support would result in complicated cross-layer interaction that breaks the current way of RAN protocol operation, as PDCP needs to check status of HARQ processes.
* Relying on feedback (e.g. HARQ feedback or uplink grant) for survival time support may require RAN3 involvement as information exchange between UPF/SMF and gNB may be needed to enable cross link coordination for end-to-end survival time support in UE-to-UE communications.

And, regarding the lack of efficiency due to the resulting resource waste brought up in previous discussions as a drawback of this method, [7] brings the following argument:

*“One may argue that Option 4 is not efficient because reliability of some packets may be boosted unnecessarily. Nevertheless, for schemes based on feedback and autonomous PDCP duplication, the gNB would anyway have to pre-allocate configured grant (CG) resources for the duplication leg, in order to ensure that duplication leg can have radio resource immediately after it is activated. Hence, this is not more spectrally efficient as compared to proactive schemes. Moreover, it is worth noting that in an IIoT/TSC use case, it is much more important to ensure applications do not fail frequently than optimizing spectral efficiency, because failure of applications in such use cases can be quite costly. Thus, spectral efficiency should not be seen as the most prominent criteria when considering the mechanism for supporting survival time.”*

**Q3: Do you support Survival Time triggered proactively based on Sequence Number?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | In addition to that there is always a resource waste for the “proactively” allocated resources (only one out of 10^4 to 10^5 would be actually useful if one assumes PER of 10^-4 and 10^-5), this can be done by gNB implementation. For example, the network can configure a duplicated leg restricted on a CG with periodicity=N\*message periodicity and PDCP discard timer is set to PDB.  [Nokia] But it means duplication is always ON, isn’t this even more inefficient ? Even the CG is with periodicity = N\*message periodicity, you still need other resources to process other duplicated packets that do not match these CG timings.  [Ericsson] There is no radio resource waste. The duplicated packets that do not match these CG timings would be dropped due to PDCP discard timer. Note that the duplicated packets are by LCP restriction restricted to be sent on that CG (allowedCG-List-r16). UE might need to process the duplicated packets, but shouldn’t be a concern since this is an industrial IoT UE. |
| Samsung | No | Benefits are unclear, while the resource wastage appears higher than in other schemes. |
| LG | No/Yes | We think UE based solution is not needed. However, if needed, we think proactive solution is better than reactive solution because the reactive solution would not work for very short ST. |
| Intel | No | A reactive solution or gNB implementation which can meet the latency requirements of a given use-case will have no reason to cause application failure. And would be preferred since it would be more spectrally efficient than a proactive solution. |
| MediaTek | No | A proactive mechanism requires additional resources all the time, to address the case where an error seldom occurs (PER of 10-4/10-5). If still seen as useful, the alternative implementation solution suggested by Ericsson (CG with a periodicity of N\*msg periodicity) is available for use today. |
| Huawei, HiSilicon | Yes/No | The main concern of proactive method is the resource inefficiency. For example, for a service with survival time=one periodicity, and the packet error rate is 0.1%, the proactive method will adopt duplication transmission for every two packets. However, packet loss may occur every 1000 packets and duplication transmission is only needed for every 1000 packets. The service is overprotected by the proactive method with the significant waste of resource. |
| CATT | No | We agree with above companies that the main drawback is the resource waste associated with blindly boosting the transmission which can be up to every other packet for periodic deterministic with 0.5ms period and ST. Regarding the issues associated with reactive triggers raised by the proponent:   * Flexible TDD: It is our understanding that flexible TDD was designed to provide a high level of both granularity (symbol) and dynamicity (slot-level) in configuring the TDD slot format to *adapt* to the traffic, not the other way around. So we think that, also considering the tight latency requirement, when NW deploys a TSN UL flow on a TDD carrier, it should always make sure that the necessary DL symbols are available following the UL transmission to allow the UE timely receiving the NW feedback.   [Nokia] Yes but flexible TDD is also used to adapt DL/UL traffic load. If we force TDD to be configured in a way of matching HARQ timing, we are also sacrificing the flexibility of supporting timely transmission in either DL and UL.   * Cross-layer interaction (PDCP/MAC): such cross layer interaction already exists since MAC already controls PDCP duplication from MAC CE reception. It can just be extended to NDI toggle check in the UL grant reception procedure.   [Nokia] In this case we need additional mapping between DRB ID and HARQ process ID, which could be dynamically changed.   * ST support in UE-to-UE communications (possibly involving multiple gNBs): there are 2 different usecases brought up by SA1:   + Distributed automated switching for isolation and service restoration: for such usecase, our understanding is that the “steady-state” requirement is rather loose (<60s) and so can be handled by gNB. The stringent part is for the fault case (≥ 1ms), i.e. when a fault situation has been detected, triggering the fault location, isolation and service restoration procedure. However a fault situation is expected to be rare and last for a short period of time. Therefore, the best NW strategy in this particular case is to increase the reliability of all transmissions during the fault period to avoid any message failure, rather than focusing on those failed messages.   + Wired-2-wireless: the ST associated to such usecase is 3x the transfer interval, i.e. 3ms which leaves 1.5ms for each Uu which is therefore 3 times less stringent than the single link of usecase #1 in Table 5-2.1. We don’t see a problem with implementing the reactive trigger for the Uu used in UL. And as agreed in RAN2#113, the ST for the Uu used in DL is left to NW implementation.   [Nokia] Yes DL is based on implementation, but when reliability is boosted in DL part, shouldn’t the gNB for DL notify the gNB for UL as well in order to boost reliability end-to-end ? Currently we do not have any trigger for the UL to boost reliability based on what happened in DL the other end, so additional specification work is needed and it impacts RAN3 too. |
| Nokia | Yes | This is required for the most stringent use cases where both survival time and transfer intervals are as short as 0.5ms. In these use cases, relying on feedback is extremely risky and restrictive, as timely feedback is only feasible with very specific subcarrier spacing and TDD configuration, which is not always allowed depending on the available/operational carrier frequency. Besides, for UE-to-UE communications, it is not clear how the two links can be coordinated to ensure survival time can be protected in an end-to-end manner when we only rely on feedback in one link.  All in all, we think proactive method is the most appropriate to ensure that survival time requirement can be fulfilled in spectral efficient manner even in the most stringent use cases, because we can avoid application failure due to unreliable feedback caused by e.g. beam blockage.  From specification point of view, we only need a configuration message for the UE to identify the PDCP PDUs with SNs that need more attention, which is not complicated.  We must point out that, making sure applications do not fail due to 5G defect is of paramount importance. Here in 3GPP we are developing technologies to compete with cable-based solutions, and this is awkward if IIoT/TSC applications fail due to survival time violation caused by e.g. feedback failure over the air interface. |
| Lenovo | No | Our main concern of the proactive method is the resource wastage. |
| Qualcomm | No (See comment) | A reactive solution would be more appropriate in terms if resource usage and performance. For the most stringent use case (ST=0.5ms), if it is shown that no reactive solution can boost reliability in time then perhaps a proactive solution may be further assessed. At this point, we prefer on agreeing on a reactive solution first, then a proactive solution might be explored as a supplement if the reactive solution fails in the most stringent 0.5ms use case, i.e., if we can justify the huge cost associated with the proactive solution in terms of additional resource usage.  For the point made in [7] and by Ericsson and LG here that for some/all use cases the resources would be pre-allocated anyway for both reactive and proactive. We do not agree with this statement but even if this was the case, proactive would increase the UE power usage by proactively duplicating every N PDUs and create unnecessary interference to other UEs, whereas reactive would only duplicate once every 10^4 or 10^5 packets, so even if preallocation was to be accepted, we still think that reactive would be preferable and should be prioritized in further discussions. |
| Xiaomi | No if the reactive trigger can fulfill all stringent requirements | Compared with the reactive trigger, the proactive trigger seems causing more resource waste. However we are open to the proactive solution if the reactive trigger cannot fulfill some stringent requirement. |
| OPPO | No | Agree with companies that the main disadvantage of proactive method is the over-protection and resource wastage. |
| Futurewei | No | Agree with Qualcomm’s comments. |
| ZTE | No | Per our understanding, the UE-based with proactive trigger scheme means that, even there is no any packet transmission failure, the reliability for some packets would be proactively(or we can even say, blindly) boosted by UE. We agree with above companies that the main drawback of such UE-based proactive mechanism is resource waste. Considering the roughly calculation by HW, such resource waste may make this scheme almost unacceptable.  Also agree with Qualcomm’s comments. |
| vivo | No | Proactive method aims at preventing a QoS flow from entering survival time in advance, which is at the expensive of large resource cost. Considering that entering survival time is a rare case, this solution is not preferred. |

#### UE-based with reactive trigger

As mentioned in Section 3.1.1, this approach mainly includes two alternate options, either based on Tx-side timer or Tx-side counting of transmission failures that we elaborate further down. In short, with reactive triggers, UE autonomously applies a more reliable pre-configured UL transmission only when needed, i.e. when it *actually* experiences Survival Time, and according to the detailed latency analysis in [6], it meets the requirements for the most stringent TSN flows of TS22.104 reusing the legacy CG configuration for that flow (no need for additional HARQ process to be configured) and allows a wide range of transmission reliability increase methods (duplication, L1/L2 parameters adaptation, …) based on pre-configuration. Note it is Rapporteur’s understanding that the additional pre-configured resources potentially used during Survival Time (e.g. with PDCP Duplication) are not wasted in “normal” state if NW knows when they are used/not used by the UE i.e. if NW knows when UE triggers Survival Time, (e.g. based on NW feedback) in which case it can reuse them in “normal” state.

**Q4: Do you support UE-based reactive ST trigger?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | We are not sure if we understand the comment by the rapporteur that the additional pre-configured resources are not wasted. If there is no other traffic to be transmitted from the UE, then the resources are wasted. In addition, the allocated resources are configured for URLLC traffic and, thus, spectrum inefficient for eMBB traffic.  Similar to the response to Q3, this would always result in an unacceptable waste of UL resources, since the network has to pre-allocate all these physical resources so that the UE can use it without network indication in, e.g., one out of 10^4 to 10^5 cases. |
| Samsung | Yes | We share Rapporteur’s understanding. |
| LG | No | As explained in Q3, we think reactive solution would not work for very short ST because it requires feedback from the network. |
| Intel | Yes |  |
| MediaTek | No | As outlined in our response to Q1, we should not introduce new mechanisms when existing NW implementation options can suffice.  We do not yet see a use-case where UE-based reactive solution will work faster/better than NW implementation, given that any reaction from the UE is based on input from the NW. However, if a valid use-case can be identified, we are open to discuss such methods. |
| Huawei, HiSilicon | Yes | UE-based reactive ST trigger is much more resource efficient. Some argue that even if UE can autonomously activate PDCP duplication when packet loss is detected by the UE, there is no available resource for the activated leg(s). We think this might not be an issue, since the network can also be aware of packet loss, the network can then timely prepare transmission resource for the newly activated leg(s). From this perspective, UE-based reactive ST triggering combined with gNB implementation would be an effective way to handle survival time for services with extremely high reliability requirement.  [Ericsson] But, if the network can predict/be aware of the packet loss, then this is the network implementation-based solution. With this assumption, there is no additional gain in UE-based solution. |
| CATT | Yes | Answering Ericsson: sorry if we were unclear. Let me try again: taking, as an example, HARQ-NACK feedback (e.g. ReTx grant) as ST trigger in the UE, it is clear that NW is aware when UE exactly activates duplication or uses additional resources, since it is gNB who sends the HARQ-NACK. Thus, in normal state, NW knows as soon as upon correct decoding of the UL transmission that UE won’t use these resources “reserved” for the next message transmission in case it would have entered ST. And so NW can schedule other UEs on these additional but unused resources. Such scheduling can be e.g. a dynamic grant for a PUSCH overlapping with said additional resources, e.g. targeting eMBB traffic. Thus, such additional resources are never wasted.  [Ericsson] The resources are allocated in the form of UL grants (configured grants or dynamic grants). In my understanding, there is no guaranteed way to ensure that the UE would never use that UL grant, for example, the PUSCH may still contain MAC CE only, a padding MAC PDU might be generated due to UCI multiplexing (the recent Rel-16 correction) and etc |
| Nokia | No, except for… | There are some problems/concerns with reactive trigger based on feedback of transmission failure:   * In cases of triggering by transmission failure, how the UE can receive the NACK quickly/reliably in such a short period of time (e.g. 0.5ms)? As mentioned above, this is only possible with very restrictive configurations and hence severely limit the network implementation/deployment flexibility. Furthermore, even if feedback is possible, it is still error-prone due to e.g. beam blockage. We should not put survival time requirement of critical applications at risk as this is very costly for the technology stakeholders. * It requires further dynamic cross-link coordination when considering UE-to-UE communications, to make sure the reliability of both links can be increased to ensure survival time can be protected in an end-to-end manner. * The MAC may need to check the LCHs corresponding to the HARQ process that needs retransmission, and further indicate to the PDCP layer to activate duplication. Moreover, what if a message is actually conveyed by two TBs in the MAC layer? There are many issues that need to be resolved in specification if we are use reactive trigger for survival time support.   The only possible reactive trigger that can be considered form our point of view is “missing packet arrival in the upper layer”. As we are dealing with periodic traffics, if a message did not arrive at the upper layer of the transmitter at the expected timing (which means it is lost somewhere in the upstream), then the survival time state should be triggered. |
| Lenovo | Yes | Agree with the rapporteur that preconfigured resources may be reassigned to other UEs for cases when no transmission failure occurred. Since gNB knows first – as argued by Ericsson – whether transmission failure occurred or not, gNB can also assign addition pre-configured resources if they are not required.  [Ericsson] The resources are allocated in the form of UL grants (configured grants or dynamic grants). In my understanding, there is no guaranteed way to ensure that the UE would never use that UL grant, for example, the PUSCH may still contain MAC CE only, a padding MAC PDU might be generated due to UCI multiplexing (the recent Rel-16 correction) and etc. |
| Qualcomm | Yes, with comment | We think most if not all short survival time use cases are best handled by a UE reactive trigger. We do not however agree that this must be tied to a timer or a counter at the UE as the UE can use some feedback methods to trigger a survival state without the presence of a timer. Thus, we agree with the general question without the timer/counter addition. |
| CMCC | Yes | We think UE triggered PDCP duplication activation/deactivation, lower the MCS or boost the data transmission power should be studied as UE reactive trigger method. |
| Xiaomi | Yes | We think that the reactive solution is more resource efficient and can provide more flexibilities for the gNB configuration (e.g. not always activate PDCP duplication and not always use CG type-2). |
| OPPO | Yes | We agree with Rapporteur’s analysis. |
| Futurewei | No | A Tx-side timer based on the PDB may not trigger as fast as a feedback from the gNB can, hence incurring additional delay to start the rescuing efforts and therefore is not preferred. Tx-side counting of transmission failures is still based on a feedback from the gNB, hence should not be counted as UE-based reactive trigger. |
| ZTE | Yes | Compared with above UE-based with proactive trigger scheme, the main difference of UE-based reactive ST trigger is that only after detecting packet transmission failure, the reliability for subsequent packets would be boosted. It’s obvious that UE-based reactive ST trigger is much more resource efficient.  Moreover, compared with gNB implementation-based scheme, UE-based with proactive trigger scheme can handle the mapping between packet loss/failure on certain granted resources and a certain periodic deterministic service and “empty packet” case. |
| vivo | Yes | Agree with Rapporteur’s understanding |

#### Tx-side Timer

This solution is supported in contributions [8][10][11][12][15][17][18].

The triggering timer, e.g. equal to 5G-AN PDB [11][12][17], is (re)started either at the reception of the packet in PDCP [11][12] or at BAT [17], or upon receiving ACK for the previous packet [15][18] and stopped upon receiving ACK. Survival Time is triggered when such timer expires [11][12][15][17][18] and optionally also upon receiving NACK [11][12][17]. It requires gNB always sending either ACK or NACK for each packet. It ensures that a transmitter interprets a lack of feedback (PDCCH miss) as a possible failed transmission but may lead to uselessly adapting the transmission resource configuration to achieve higher reliability (e.g. in case missed PDCCH was carrying ACK).

**Q4-a: If you answered “Yes” to Q4, do you support Tx-side Timer method for triggering ST?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | Even though we haven’t answered “Yes”, we would like to point out the issues in this solution.   1. This solution requires gNB to send always a ACK and NACK, which is not feasible in licensed spectrum operation, as there is no PHICH channel in NR. For shared spectrum operation, it is agreed in the last meeting that the support in UCE is up-to network configuration. 2. As strongly argued by the rapporteur that gNB based implementation has issues when the survival time is short, the tx-side timer has similar issues since transmitting a PDCCH every 0.5 millisecond is a large resource waste, if feasible at all. |
| Samsung | Yes | We are one of the proponents of this approach. Regarding Ericsson’s concern about ‘large resource wastage’ – this is not how we see it. Entering ST state is not a frequent occurrence. While it may be comparable (overhead-wise) with some gNB-based implementation solutions (as noted by Ericsson), the advantage of the Tx-side timer over gNB-based implementation solutions is that it can actually get the job done in the sense of quicker reaction. |
| LG | No | We share the concerns from Ericsson. In addition, as explained in Q4, reactive solution would not work for very short ST because it requires feedback from the network. |
| Intel | Yes | We support Tx-side Timer however, we do not agree that it requires gNB always sending either ACK or NACK for each packet e.g the timer can be started after receiving NACK only in the case of failed transmission. |
| MediaTek | No | We share the concerns from Ericsson, that there is no PHICH like in LTE to provide HARQ ACK to the UE. Furthermore, as this method is based on ACK/NACK info from the gNB, it is obvious that the gNB is aware of a failure before the UE, and can react faster than the UE. |
| Huawei, HiSilicon | Yes with comments | For uplink transmission on licensed spectrum, there is no explicit HARQ feedback for each PUSCH transmission. It is not easy to direct apply such method using HARQ feedback, while relying on RLC feedback suits for services with no stringent survival time requirement.  We think Tx-side Timer method is inferior to the Tx-side Counter method discussed in the following section. |
| CATT | No | We agree with Ericsson that mandating gNB to always send ACK for each successful transmission is cumbersome (we are not clear how it could work without explicit ACK). In addition, there are different views on when to start the timer, but for the case where the timer is started at the reception of the packet in PDCP, gNB cannot exactly know when it was started and so when it expires. Thus, there is only loose synchronization between NW and UE regarding when ST is triggered. On the other hand the benefit of this approach is that it is robust against PDCCH miss. However, this is only an issue if the PDCCH was actually carrying NACK. Given the PDCCH BLER is typically in the range [10^-6 10^-5] and the PUSCH BLER associated with the TSN flow in the range [10^-5 10^-4] the probability to have both consecutive events is in the range of [10^-11 10^-9], so that may not be a showstopper. |
| Nokia | No | We are dealing with cyclic deterministic uplink traffic here, which will be typically conveyed by configured grant in practice. HARQ ACK/NACK from the gNB is not expected for configured grant and not supported in the current specification. Introducing explicit HARQ-ACK for each packet will bring significant work in RAN1 as well. And hence introducing such a timer is rather pointless. |
| Lenovo | Yes/No | Timer based solution could be considered. |
| Qualcomm | No | We do not see a need for a UE Tx timer. An ACK/NACK feedback is not available to UE in licensed band and mandating that type of feedback may not be feasible. For the sake of argument, assuming an ACK/NACK is available from gNB, then, the UE can just use the NACK as a trigger to enter survival state without a timer. In the baseline licensed operation when there is no ACK/NACK, the UE can infer the survival state from a scheduled retransmission on DG (we refer to that as an **implicit NACK**). In any case, as long as the UE relies on some well-defined trigger, a timer will not be needed.  We think that agreeing on the more fundamental two questions (triggers for survival state and how to increase link reliability) should precede discussions about specific implementations which may or may not utilize timers.  We further add that the current timer proposals complicate the UE implementation with no verifiable benefit, we would much rather prefer a simple operation with no timers. |
| CMCC | Yes | It could be considered as one solution. |
| Xiaomi | No | The transmitter should be allowed to tolerate some loss of PDCP PDUs according to the definition of the survival time, given that the survival time is considered as “as a maximum time in units of “time” where each unit corresponds to the data burst periodicity defined in TSCAI in Rel-16”. |
| OPPO | Yes | If we need to support short ST service, we need to accept the related work/price. Considering entering ST state may not be a frequent event, the scheme on Tx-side timer still has its own benefit on quick adjustment. |
| ZTE | Yes | Generally agree with Samsung.  We think that the value of Tx-side Timer can be the value of one AN PDB, specifically:   * The timer could be started at the reception of the packet in PDCP. * If the message is not sent successfully due to the Tx-side timer expires, UE can enter ST state. Such process doesn’t rely on feedback from network and therefore no delay issue. * If the message is sent successfully before the Tx-side timer expires, UE can stop the Tx-side timer without entering ST state. For example, during running of Tx-side timer, if UE receives initial Tx grant, the UE can assume implicit ACK for previous packet transmission and stop the timer; If neither initial Tx grant nor ReTx grant is received during the whole Tx-side timer, the UE can also assume ACK for packet transmission and stop the timer; But if one or more ReTx grant have been received, we think an explicit ACK would be needed from gNB to UE if gNB finally successfully receives the packet. Such ACK can be transmitted on PDCCH. As we assume such explicit ACK would be only for a few packets (not each packet), the resource waste would be limited. By above way, UE can correctly stop the timer when it needs. |
| vivo | Yes | In application layer, RX enters survival time when it does not receive a packet successfully within a predefined time. Similarly, in RAN domain, RX can enter survival time when it does not receive a packet successfully within AN PDB. In order to facilitate TX to monitor the status of survival time at RX side, introducing a TX Tx-side Timer (i.e. timer value is set to AN PDB) for triggering ST is reasonable. |

#### Tx-side Counter

This solution is supported in contributions [5][6][8][16][18][21][22].

Survival Time is triggered when UE experiences N consecutive UL transmission failures for the flow/DRB/LCH configured with Survival Time. Transmission failures are detected by receiving NACK from the NW. NACK can be HARQ-NACK, e.g. DCI retransmission grant [5][6][8][16][18], or RLC NACK [5][16]. In its simplest form, N=1 and Survival Time is triggered upon receiving a HARQ NACK [5][6][8][16][18].

**Q4-b: If you answered “Yes” to Q4, do you support Tx-side counter-based method for triggering Survival Time?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | Even though we haven’t answered “Yes”, we would like to point out the issues in this solution.   1. The network may dimension multiple retransmission grants to meet the (PDB, PER) target in the “normal” state. It is not clear from which retransmission grant the UE needs to re-act.   [CATT] We understand “multiple retransmission grants” as addressing consecutive transmission failures of the same application message (?). In such case, UE reacts (i.e. ST is triggered) on the earliest received retransmission grant for that message.  [Ericsson] Yes, the network may schedule multiple retransmission grants to meet the PDB of one message. If it is always the first retransmission grant, then the UE unnecessarily enters survival time mode.   1. This method mandates that the network to always schedule a retransmission for the “failed” message transmission, which is far less important than the message in the subsequent period.   [CATT] Not necessarily, as mentioned in [6], we could also reuse DFI mechanism for just sending NACK. All the mechanism is already available; it is just a matter of allowing configuring it for that purpose. We agree that it is less important than addressing the reliability of the message in the subsequent period, but that’s the key benefit of the Tx-side counter-based method: it allows addressing both without time conflict.  [Ericsson] RAN2 has agreed not to optimize for UCE environments and whether DFI is suitable for licensed spectrum with very short survival time needs further RAN1 confirmation.   1. As strongly argued by the rapporteur that gNB based implementation has issues when the survival time is short, the tx-side counter would not work with RLC NACK and a short survival time.   [CATT] We agree that RLC NACK is challenging for short ST. |
| Samsung | Yes | We could consider this. |
| LG | No | Same comments as Q4-a. We share the concerns from Ericsson. In addition, as explained in Q4, reactive solution would not work for very short ST because it requires feedback from the network. |
| Intel | See comment | This is a similar approach as the Tx-side timer. RLC NACK may be too slow for some use-cases. |
| MediaTek | No | Same comments as Q4-a. The NW is positioned to react faster than the UE here. |
| Huawei, HiSilicon | Yes | Tx-side counter-based method is a simple and easily-implemented solution for triggering survival time. For IIoT services with extremely stringent service availability requirement, the counter value can be set as one, which means each HARQ retransmission will trigger UE to improve transmission reliability. Here, HARQ retransmission works as an implicit NACK indication without introducing explicit HARQ feedback for uplink transmission. |
| CATT | Yes | This solution does not require any new timer and does not mandate gNB sending back HARQ-ACK for every successful packet. So we think it is attractive by its simplicity.  Answers to Ericsson’s comments are added above. |
| Nokia | No | A HARQ-NACK corresponds to a TB, rather than an application message where the survival time is concerned. RLC-level NACK is far-fetched considering the delay. |
| Lenovo | Yes | We prefer a UE-based solution where UE triggers based on reception of a NACK or retransmission DCI, some mechanism which increases the transmission reliability. |
| Qualcomm | No | We did not answer “yes” to the previous question however we would like to clarify that the scope of UE-based solutions considered here is extremely short survival times triggered by a single failure. In that sense, a counter will not be useful, as the UE has no time to count multiple failures (FFS how a failure is indicated) before triggering a survival state. We reiterate our view that the priority here is a fast UE reaction to a survival state which requires a simple trigger (such as an implicit NACK), rather than a counter operation |
| CMCC | Yes | We think RAN can detect message failure via the transmission state of previous packet, based on HARQ feedback or ARQ feedback during the survival time |
| Xiaomi | Yes | Regardless of whether we use a timer or a counter to detect the survival time, some PDCP PDU loss should be allowed. |
| OPPO | See comment | As we understood, Tx-side counting solution is similar as Tx-side timer solution. But, Tx-side counting solution might not cover the case, e.g. the UE needs to rely on the timer to trigger more reliable transmission in case that RTT is large. |
| ZTE | No | We assume such Tx-side counter would be incremented based on each reception of ReTx grant.  As the duration between retransmissions may be variable, e.g., depending on the gNB scheduling, it may be difficult to configure a suitable number for such counter that can be consistent with AN PDB. If the configured number is too small, it may cause that UE enters ST state too early, and may even cause increased reliability for the retransmission of current packet. This is unnecessary. Or if it wants to avoid such process, the gNB’s retransmission scheduling may be subject to some restrictions. This is also undesired. |
| vivo | See comments | We understand the difference of TX-timer based solution and TX-count based solution (when N is not equal to 1) is whether enhanced scheme needs to adopt in case the transmission of a packet fails. We also observe that these solutions are essentially same for the use case that survival time equals to transfer interval.  Even though we prefer to follow the design of survival time in application layer (i.e, entering ST once the failure of a packet is detected) , we are also open to consider this solution. |

#### Other triggers

#### Feedback enhancements

In [12][22], it is suggested that new feedback mechanisms, faster than current HARQ RTT should be studied. For example, in [12]: “*in order to be able to accurately determine whether the data packet has been successfully sent within the AN PDB time, the feedback corresponding to the data packet needs to be notified to the sender within the AN PDB time and as soon as possible. This part may still need to be enhanced*”. And in [22]: “*To meet the stringent delay requirement of the URLLC services, fast feedback should be introduced to reduce the HARQ feedback delay*”.

However, Rapporteur suggests, to narrow the scope of this discussion, that the above two enhancements can be considered as part the above Tx-side timer and/or counter based solutions, if selected, and can be discussed later.

#### Survival Time and UCE

In RAN2#113-e, it was agreed:

*-* Support for survival time in UCE is up to network configuration.

[10] asks RAN2 to consider if survival time can be applicable to UCE. However, it is Rapporteur’s understanding that one consequence of the above agreement it that configuring/scheduling UL transmissions of a flow/DRB/LCH configured with Survival Time on an unlicensed frequency cell under controlled environment (UCE) is not precluded and so, should be supported. Then, [5] suggests that Survival Time in UCE should be handled differently than in non-UCE, for example leveraging the *cg-RetransmissionTimer* and taking LBT failures into account. At this early stage, rather than going into the details, we propose first to answer the below basic question.

**Q5: Should we consider additional aspects for handling Survival Time in UCE?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | We are not sure why we re-discuss the same question, since the paper [10] is a resubmission whose topic was discussed and concluded in the last meeting. In our view, industrial IIoT would mostly and firstly be deployed in the licensed spectrum and we want to focus on enhancements for licensed spectrum first.  At this moment, with the assumption of UCE (i.e., LBT fails rarely), the operation LBT failures can increase the risk of ST expiration they are seen as occurring infrequently and therefore we can justify not considering their potential impact at least within the scope of Rel-17 ST operation. |
| Samsung | No |  |
| LG | No | RAN2 already made agreements on this issue. |
| Intel | No | Previous agreement in RAN2#113e seems sufficient for now “Support for survival time in UCE is up to network configuration” |
| MediaTek | No |  |
| Huawei, HiSilicon | No | We shall focus on discussion about handling survival time on licensed spectrum. |
| CATT | FFS | Agree with above companies that we may consider this at a later stage but we would prefer to still leave the door open to some UCE-specific adaptation. |
| Nokia | No | We have already agreed that no optimization for UCE will be examined. |
| Lenovo | No but | We agree with the [10] that LBT failure will increase the probability of ST expiry, therefore consideration of LBT failure makes actually sense. However as mentioned by other companies, RAN2 discussed this issue already and concluded already not to specifically address this in Rel-17 |
| Qualcomm | Yes | We agree with the rapporteur’s understanding regarding UCE. We note that the challenge in survival time is often detecting failures at the UE. UCE does offer ways to detect failures through LBT and DFI configuration. Furthermore, UCE has the unique possibility of an LBT which we do not account for in the licensed band. Thus, a UCE specific solution would actually be better than reusing whatever is specified for licensed. In our view, a complete solution should tackle both possible use cases and strive to offer the best operation separately. |
| CMCC | No |  |
| Xiaomi | No |  |
| OPPO | No | As we understood, it is already agreed that no extra enhancement on ST support in UCE in Rel-17, i.e. only the solution for CE is applied, if ST is configured by the gNB. |
| Futurewei | No |  |
| ZTE | No |  |
| vivo | No |  |
| III | No | At least in Rel-17. |

### Link reliability increase mechanisms

Consistent with the outcome of the RAN2#113-e meeting offline [2], the solutions proposed in the contributions to RAN2#113bis-e can be classified into 3 groups:

1. Duplication activation [5][6][7][8][9][13][15][16][17][22]
2. L1/L2 configuration adaptation [5][7][8][10][11][15][16]
3. gNB scheduling [14][17][19][20]

However, we think the gNB scheduling option is already addressed by Q1. Indeed, similar to what we mentioned in Section 3.1.1.1, it is obvious that gNB has full freedom to use legacy scheduling and reconfiguration mechanisms to upgrade the radio resources targeted for an UL flow whenever it estimates it is needed, and the question here is rather whether it is sufficient to handle all usecases of Table 5.2-1, which is already addressed by Q1. Therefore, we focus here on the link reliability increase mechanisms autonomously triggered by UE-based solutions.

#### Duplication activation

**Q6: Should it be possible for a UE, when configured accordingly, to autonomously activate duplication when Survival Time is triggered?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | Per Q2, we propose a gNB can and should determine when the ST timer is triggered and what to do thereafter (if anything). It would be possible if network has pre-allocated the resource for UE to transmit the duplicates. But it requires to have reserved UL resources all the time for the LCHs with PDCP duplication configured but not activated yet. Use of survival time is a very rare event which means that any reserved UL resources would ONLY be used for one out of 10^4 to 10^5 attempted message transmissions, making this approach extremely spectrum inefficient. In this aspect, we don’t consider it is “possible” since it does not make sense for the network to allow this possibility. |
| Samsung | Yes but… | Rapporteur’s intention here is unclear to us. Methods under 1 and 2 summarized in 3.1.2 above seem to cover reactive triggering discussed previously and as such we support them.  However the Rapporteur seems to imply that the present section covers ‘autonomous’ UE action? If this is the intention, then we could consider this approach since it leads to fast reaction and entry into ST state. However, new signaling may be required to inform the gNB the terminal has entered ST state. Additionally, new rules need to be agreed on when the terminal may enter ST. We therefore propose to study this once the details of the baseline reactive mechanisms are agreed. |
| LG | No | We share the concerns from Ericsson. If the UL resources (for sending duplicated packets) are pre-allocated, radio resouce would be wasted very much. Else, if the UL resources (for sending duplicated packets) are not pre-allocated, the UE has to send BSR, which is essentially same as network control of PDCP duplication, i.e. activate PDCP duplication based on the BSR. We don’t see any benefit of UE autonomous activation of PDCP duplication. |
| Intel | Yes | UE autonomous PDCP duplication is expected to have reduced latency |
| MediaTek | No | Agree with Ericsson and LG that pre-allocated UL resources are wasteful to address errors that take place with a probability between 10-4 and 10-5 |
| Huawei, HiSilicon | Yes | Duplication is an effective method to improve transmission reliability. Autonomous duplication activation according to triggered conditions, e.g. HARQ retransmission, together with gNB implementation to timely schedule transmission resource would be a promising solution to handle survival time. |
| CATT | Yes | We think this is the most efficient way to increase the link reliability, also considering beam failure in the main leg. As elaborated in our answer to Q4, the resources of the duplicated leg are NOT wasted in normal state since NW is aware that they are not used by the UE hence it can allocate them to other UEs. |
| Nokia | Yes | This is possible, but we think the UE should be pre-configured. For instance, when the DRB is configured with more than two RLC entities, a subset of RLC that the UE is allowed to activate should be indicated to the UE beforehand, in order to ensure the gNB and UE have a common understanding about what leg will be activated in survival time state. |
| Lenovo | Yes | The mentioned resource wastage is not so clear, since NW could reassign the resources to other UEs in case packet was correctly decoded. Essentially NW knows - as mentioned - first whether Packet transmission was successful or not. We think that UE autonomous activating PDCP duplication addresses the potential error cases, e.g. beam blockage, best. |
| Qualcomm | Yes | PDCP duplication is a very effective method to boost reliability which is exactly what is needed in survival state. Making duplication UE autonomous is intended for fast reaction before survival expiry. We think that this solution would be the most effective one. As for the concern about pre-allocation we do not think that this would necessarily be the case. For example, the UE can autonomously activate PDCP activation through a pre-configured trigger by the network. The gNB can then send a DCI for a CG or a DG on the other carrier (since it will also know that the UE has activated duplication) to provide the needed resources for duplication. This would circumvent the need to explicitly activate duplication from the gNB via a MAC CE (and taking a big hit in terms of latency that may cause survival expiry). This is just one possible solution. Second of all, the resources in IIoT use cases may not be the bottleneck. For example, the survival expiry causing the system to go to a “DOWN” state may be a very drastic outcome that the network would rather dissipate those extra resources to guarantee no survival expiry happens than saving resources and risking a DOWN state. On the other hand, the network may not be inclined to activate duplication all the time (every PDU) or proactively (every N PDUs) for UE power and cell interference considerations. We think in this use case reactive duplication may be the most appropriate, effective, and reliable solution. |
| CMCC | Yes | The UE triggered PDCP duplication activation/deactivation mechanism was discussed in R16 and could be reconsidered since it may fasten the PDCP duplication activation/deactivation procedure |
| Xiaomi | Yes | PDCP duplication is used to ensure URLLC transmission. It is more efficient to activate the PDCP duplication only when there is a need to meet the survival time requirement. |
| OPPO | Yes | It is an efficient way to improve link reliability and decrease reaction latency. We agree with companies that resource wastage is not a big issue, even if it exists, considering the pre-allocated resource can be occupied by other UEs or the risk on entering DOWN state is more unacceptable issue. |
| Futurewei | No | The UE may activate UL duplication in response to a feedback from the gNB. The feedback serves as an explicit or implicit activation command. There is no true UE autonomy in doing this. |
| ZTE | Yes | We understand Q6-Q7 mainly address how to increase reliability for the subsequent transmission after entering ST state. Generally, no matter which kind of trigger (UE-based or gNB-based) can be agreed according to discussion in Q2-Q4, PDCP duplication would be an effective method.  One issue in PDCP duplication is that additional delay may be caused by activating/deactivating PDCP duplication through MAC CE from gNB. Therefore, one possible solution would be autonomously activating PDCP duplication by UE self. Moreover, we agree with CATT that the resources of the duplicated leg are NOT wasted in normal state.  However, we disagree that PDCP duplication is just activated when entering ST state. Although we think in the most stringent case, entering ST state can activate PDCP duplication at the same time, it is obviously such process is not suitable in all scenarios. In scenarios where the value of survival time is equal to or greater than 2 transmission intervals, it’s no need to activate duplication just when entering ST state. In order to avoid resources waste, PDCP duplication can be activated a bit later, e.g., just before maximum tolerable number of packet losses or failures. This can be achieved via another timer, e.g., Survival Time timer. See our comments in Q8. |
| vivo | Yes | PDCP duplication is at the cost of resource overhead to improve reliability. It should be used only when entering survival time. |
| III | Yes | If the UE-based solution is adopted, autonomously activate duplication in the UE side can avoid the most stringent survival time constrain become violated. |

#### L1/L2 configuration adaptation

At this stage, this option can include, for example, flexible L2 configuration switching (e.g. RLC leg switching [7][8]), dynamic LCH configuration change (e.g. priority), UE autonomous LCH restriction relaxation, and LCP adjustment by pre-defined rule [10], adaptive L1 configuration [11] e.g. lower the MCS or boost the data transmission power [16].

**Q7: Should it be possible for a UE, when configured accordingly, to autonomously adapt its L1/L2 configuration when Survival Time is triggered?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | See response to Q6. In addition, there are no details for each solution. |
| Samsung | Yes but… | Please see our response to Q6. |
| LG | No | Same comment as Q6. |
| Intel | See comment | We support autonomous PDCP duplication by the UE. Other solutions, e.g lower MCS, may also have RAN1 impact and LCP adjustment may not guarantee that survival time is not violated. |
| MediaTek | No | Same comment as Q6 |
| Huawei, HiSilicon | Yes | We think flexible L1/L2 configuration switching can be combined with gNB implementation solution to handle survival time. |
| CATT | Yes but | we need to nail down the different options. |
| Nokia | Yes | Essentially, we think RLC leg switching can be deemed as a general framework to cover all possibilities, including adaption in LCH configuration, LCH mapping restriction, and L1 parameters etc.  So, we think RAN2 only needs to specify the framework of RLC switching, the rest of details can be leave to gNB implementation, which provides the best flexibility for the gNB. |
| Lenovo | No | We think PDCP duplication is best suited. |
| Qualcomm | No | More details are needed on the specifics to assess the efficacy of this solution. Our concern is that flows with ST constraints are already configured very conservatively in terms of MCS, LCP, etc. (expected to have the highest priority), we doubt there is any room to reconfigure L1/L2 to increase reliability. In any case, PDCP duplication should be prioritized. We also have a big concern that changing PHY parameters on the fly would cause significant UE implementation complications and would have large RAN1/RAN4 impacts that require lots of evaluation from these WGs. |
| Xiaomi | Yes | We see the benefit of more adaptive way to change the L1/L2 configuration. However as there are many L1/L2 parameters which can be changed by the UE, we should narrow down the scope of discussion in this aspect. |
| OPPO | Yes | We think the flexible L1/L2 configuration switching is another way to ensure ST requirement. |
| Futurewei | No | The UE may switch its L1/L2 configuration in response to a feedback from the gNB. The feedback serves as an explicit or implicit switch command. There is no true UE autonomy in doing this. |
| ZTE | Yes but | We agree that flexible L1/L2 configuration can also achieve the purpose of increasing reliability. In some scenarios where PDCP duplication is not supported or is unsuitable to be used, e.g., for the UE at the center of the cell, this scheme can be used. |
| vivo | Yes | Flexible L1/L2 configuration is an efficient way to handle survival time, e.g. autonomously adjust the priority is beneficial for obtaining grant quickly. |

### Survival Time management

Other than increasing the reliability, the UE behavior while in Survival Time is discussed in various contributions, specifically:

1. Survival Time timer
2. Return to normal state

#### Survival Time timer

A timer is discussed for monitoring the Survival Time itself [5][8][11][15][18]: the transmitter starts the Survival Time timer upon one of the triggers discussed in Section 3.1.1 (whichever is agreed). The transmitter would subsequently stop and reset the timer when indicated that one (or more) transmission(s) succeeded via e.g. explicit HARQ (or RLC [15]) ACK or upon CG timer expiry [5]. However this timer only makes sense if UE has an action upon its expiry, which is proposed to be studied by [18]. The proposed UE behaviors discussed in contributions depending on Survival Time timer status are:

* UE decreases / increases reliability if it receives ACK while timer is running / not running (incl. expired), respectively. Or the other way around. [15]
* UE returns to normal state when it leaves Survival Time (understand incl. timer expiry) [11]
* There is no specific behaviour upon Survival Time expiry, so such timer is not needed [5]

**Q8: Is a Survival Time timer needed for monitoring the Survival Time itself? If yes, please indicate the impact of this timer on the UE behavior (e.g. when timer expires).**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | Per Q2 response, the gNB is in control of what to do when ST timer is started and expires à the UE does not need a new timer. We are not sure if there is any difference from the “Tx-side Timer” discussed in the subsection 3.1.1.3.1.  [CATT] The “Tx-side Timer” discussed in the subsection 3.1.1.3.1. is for monitoring the “entering” into Survival “state”. Survival Time timer discussed here is for monitoring the “exit” out of Survival “state”. The “Tx-side Timer” starts the Survival Time timer and is proposed by some companies to be set to the PDB. The Survival Time timer should be set to the value provided in the TSCAI. |
| Samsung | Yes | We think that some action is always needed upon ST expiry, even if it is just to return to “normal” state. So yes we do need a timer. |
| LG | No | We share the concerns from Ericsson. |
| Intel | No | Once survival time has expired, the application shall go into down state. UE/gNB actions to ensure that survival time is not violated come before the expiry. We think that for such actions, the Tx side timer in 3.1.1.3.1 is sufficient. We don’t think this additional timer is needed. |
| MediaTek | No | We share the concerns from Ericsson. |
| Huawei, HiSilicon | Yes | During such ST timer is running, the UE improves its transmission reliability. When this timer expires, the UE returns to normal state. We prefer to have this timer to make UE behavior clear and predicable. |
| CATT | No | We fail to see how Survival Time timer expiry, meaning the application communication service is “down”, would impact the UE.  Also please find some above clarification to Ericsson. |
| Nokia | No | The motivation of such a timer is not clear. Eventually the UE can only use the resource allocated by the gNB anyway, so gNB can always ask the UE to fall back to normal state by e.g. deactivating some resources. Moreover, if the proactive trigger discussed in Q3 is applied, there is no need for such timer at all. |
| Lenovo |  | Could be further discussed. Seems to be related to the Tx-side Timer option. |
| Qualcomm | No | UE increases reliability anyway upon survival state trigger irrespective of the specific timer reading. There is no specified behavior of UE upon ST expiry, so a timer is not needed there either. Thus, no Survival time timer is needed at the UE.  Again, at this point we are not clear how the associated complexity increase at the UE is justified. |
| Xiaomi |  | We are open for this discussion. However, we would like to firstly understand the use cases of the survival time, e.g. why this timer is needed and how to start/stop the timer and what the expected UE behaviors could be when the timer expires. |
| OPPO | Yes | Since ST timer can be maintained by the UE side, we want to have a clear UE behavior. When ST timer is running, the UE choose a reliable transmission. When ST timer is stopped, UE choose a relaxed transmission. Note that ST timer may not be expired but stopped when e.g. one packet or PDCP SDU is successfully transmitted. |
| Futurewei | FFS | It may depend on eventually how the survival time operation is handled. We can postpone this until the big picture is clearer. |
| ZTE | Yes | Generally we have sympathy with CATT’s explanation on intention of “Tx-side Timer” and “Survival Time timer” as response to Ericsson. More specifically:   * The Survival Time timer is started upon expiration of the “Tx-side Timer”, e.g., when entering ST state. * In the less stringent cases other than the most stringent case, ONLY upon expiration of the Survival Time timer, UE can autonomously activates PDCP duplication. As mentioned in Q6, such process can avoid too early activation of PDCP duplication since a bit more transmission failures can still be tolerated. * In the most stringent case, the start of Survival Time timer can also immediately trigger reliability enhancement for the subsequent link transmission, e.g., activating PDCP duplication. In this case, the Survival Time timer can be killed just after activation of PDCP duplication (this is just to keep common process for all the cases as much as possible. Another possible way may be not to start Survival Time timer in this case). * In order to coordinate with higher layer requirement on survival time, we assume the Survival Time timer here should be less than the value provided in the TSCAI; * Survival Time timer needs to be reset upon successfully packet transmission. That means the UE can “exit” out of ST state.   Here a related scenario may need further consideration, e.g., even PDCP duplication is activated, it’s still possible that all the packet transmissions equal to the value in TSCAI are failed. We assume UE can have an overall statistic to identify this scenario, e.g., it may also rely on this Survival Time timer or some other way. Upon identification of such scenario, UE may be able to take some action, e.g., handover to other cell or report something to the network. |
| vivo | Yes | When ST timer expires, the application layer at RX side will enter communication unavailable status. Transmission with high reliability may be no longer needed. Thus, we prefer to maintain a ST timer to allow UE autonomously fallback to transmission scheme with lower reliability. |

#### Return to normal state

Once a UE is in Survival Time, [8][11][17][18] suggest that it autonomously returns to normal state (default reliability) when indicated that one (or more) transmission(s) succeeded. However, it is expected that those companies that consider that Survival Time trigger should all be left to gNB implementation would also assume it is under gNB responsibility to decrease the link reliability when it is appropriate to do so. Moreover, given it might be considered less time critical as for entering Survival Time, even with a UE-based Survival Time trigger, one could consider that the relaxing of the UL transmission reliability can be left under gNB control.

**Q9: Should UE autonomously return to normal state (default reliability) based on pre-configured criterion, or should it be left to network’s responsibility when and how to reconfigure the UL transmission back to a default configuration?**

|  |  |  |
| --- | --- | --- |
| **Company** | **UE/NW** | **Comments** |
| Ericsson | NW | It is indeed less time critical to exit the survival time and can be left for network control. |
| Samsung | Both | Both of the options could be considered – they are not mutually exclusive and can co-exist with appropriate configuration. E.g, assuming the Tx timer solution is adopted, ST state could be exited (in some scenarios) even before the timer has lapsed, for instance based on receiving signaling from the NW. |
| LG | NW |  |
| Intel | See comment | We think some statistical evaluation is required to ascertain what can qualify as a return to normal condition. For now, we can assume that return to normal state may be autonomously triggered by UE based on preconfiguration or be network controlled, however, this needs further discussion and analysis in RAN2 |
| MediaTek | NW | This should be left to NW implementation |
| Huawei, HiSilicon | UE | As in previous question, the ST timer can be maintained by the UE to control when the UE shall return to normal state. |
| CATT | NW | Unlike entering into Survival Time, the fall back to normal state is less critical and can be left under gNB control. |
| Nokia | NW | Besides, it is much more straightforward if proactive trigger in Q3 is applied, because the UE would only care about the SN of the incoming packets and it knows what to do based on pre-configuration. |
| Lenovo | UE | There could be some predefined criteria to control when the UE shall return to normal state. NW based solution would be also acceptable to us |
| Qualcomm | Both | Network would be able to instruct UE to exit survival state. FFS on whether there are benefits for the UE to be preconfigured and do it autonomously as well, thus we do not preclude it. |
| Xiaomi | See comments | We think this can be discussed later after the solution for survival time state is selected. Then we can analyze whether the gNB implementation is sufficient to send the UE back to “normal” state. |
| OPPO | UE | Since ST timer can be maintained by the UE side, it is a straightforward way for UE to return to normal state if the timer is stopped. |
| Futurewei | NW |  |
| ZTE | Both | After UE autonomouslyactivates PDCP duplication, if NW successfully receives the packet, NW can use the existing mechanism for deactivating PDCP duplication to indicate the UE to return to the normal state.  We also agree with Samsung and HW that it’s also possible for UE to control. |
| vivo | both | We think NW and UE can control the UL transmission back to a default configuration. UE based fallback can be configured by NW. |

## Burst Spread, Burst End Time

During SA2#143E, no conclusion could be reached regarding the inclusion of the Burst Spread and Burst End Time (BET) parameters in TSCAI parameters, hence they are not captured in TS 23.501 [4]. As a result, if RAN2 decides that either or both parameter is needed by RAN, RAN2 should explicitly ask SA2 to (re)consider introducing the(se) parameter(s) in TSCAI.

### Burst Spread

As a consequence of SA2 decision (or lack of), [5][7][14] suggest to not consider the Burst Spread parameter in RAN, [10][17] to further wait for SA2, while [8] think it should not impact RAN specifications anyways.

**Q10: Should RAN2 ask SA2 to consider Burst Spread in TSCAI?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | No | Not needed. |
| Samsung | No | This is an SA2 decision. If they need our input into their decision-making, they will send an LS. There is no need for RAN2 to trigger this. |
| LG | No |  |
| Intel | No | Similar to existing QoS parameters, while the new burst spread parameter (if introduced) may be useful to the gNB scheduler, its utilization is expected to be handled by the gNB implementation without any explicit RAN2 specification impact and this decision can be left up to SA2. |
| MediaTek | No | This is an SA2 discussion |
| Huawei, HiSilicon | No | SA2 has already decided to not include Burst Spread in TSCAI. Up to now, RAN2 has not found strong necessity to have such parameter in TSCAI, we shouldn’t ask SA2 to reconsider such parameter without further motivation. |
| CATT | No |  |
| Nokia | No | Burst spread is no longer in SA2’s scope, we don’t see why RAN2 should further ask for it. |
| Lenovo | No |  |
| Qualcomm | No | The solutions being discussed here have no use for Burst Spread. |
| CMCC | No | RAN2 may not consider studying the further mechanisms for the impact of burst spread, which will not be provided to RAN from SA2. |
| OPPO | No |  |
| Futurewei | No |  |
| ZTE | No |  |
| vivo | No | SA2 has discussed the burst spread and decide not to introduce in TSCAI. As RAN2 work is based on SA2 decision, then there is no need to ask SA2. |
| III | No |  |

### BET

BET is promoted in [14] as a key parameter supplementing the burst arrival time (BAT) thus allowing a gNB to determine the latest point in time where UE or gNB can receive a packet for inclusion in the next instance of a radio resource (uplink or downlink respectively) used to support transmission of packets associated with a given QoS flow. Further arguments in support of BET are given in [14] as follows:

* The key benefit of BET is that it provides the gNB with enhanced flexibility regarding where it can schedule the DRB resources needed for supporting a given QoS flow while satisfying the PDB requirement. Without BET a gNB will not know the point in time when the last packet will arrive which forces it to identify a latest point in time (i.e. a “drop dead time”) for accepting packets to be aggregated into the next MAC PDU to be sent, e.g., using a periodic DL SPS configuration or in a dynamic resource allocation used for the corresponding QoS flow. Assuming a “drop dead time” for BET can substantially reduce the flexibility regarding where a gNB can schedule the corresponding DRB resources and thereby reduce overall efficiency in managing radio resources. When there is only one packet in the burst, the value of the BET is the same as the value of BAT.
* The value of BET can be determined e.g. using the width of the TSN gate open cycle during which the UPF/NW-TT receives the set of packets and therefore enhancing TSCAI to include BET seems to be a feasible option.

In addition, [14] considers that BET is necessary for a correct Survival Time monitoring because the survival time should be counted from the PDB after the burst end time, see below Figure 1. In this way, it accounts for the possible latest arrival time in the subsequent bursts and thus the survival time can make sense for RAN.



Figure 1 Correct Survival Time Calculation

Opposite views are expressed in [7][8][17][20] arguing that BET can be deduced from other parameters e.g. burst size (or MDBV), while TSC flows are expected to experience minimum jitter, or that time difference between burst arrival time and burst end time is not expected to be large enough to be taken into account by the gNB for scheduling. And [5] considers that, following a reactive approach where RAN would start the survival state **reactively** upon a packet failure without considering burstiness of the application (Tx-side counter, Section 3.1.1.3.2) Burst Ending Time and Burst Spread knowledge are not needed at RAN for survival time monitoring.

**Q11: Should RAN2 ask SA2 to consider BET in TSCAI?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Ericsson | Yes | It is our understanding that MDBV (which is used for burst size in TSC AI) refers to the **maximum** data burst volume. If the actual message size is smaller than MDBV, then it is infeasible to deduce BET. Only if the actual message size is always equal the MDBV, then the network can deduce the BET. But the whole point of the TSC AI with BAT is for the network to pre-allocate the resources before traffic arrival. Otherwise, network can observe the arrival time and there is no need for BAT either.  We cannot assume that the time difference between burst arrival time and burst end time will never be large enough to be taken into account by the gNB for scheduling. In addition, if a gNB always assumes BET = BAT, then the gNB may allocate DRB resources that start prior to arrival of the last UL/DL packet to be included in a given CG/SPS resource and there are no resources for the transmission of these packets.  Regardless of the outcome of this discussion, we believe that there is a need to clarify that when the survival time starts, for example, if the BET were not agreed in this release, then at least the start of the survival time should be at some definite timing point, such as BAT + PDB. |
| Samsung | No |  |
| LG | No |  |
| Intel | No |  |
| MediaTek | No | This is an SA2 discussion |
| Huawei, HiSilicon | No | We understand that it was assumed that one application message is conveyed by one PDCP SDU and SA2 has concluded there is only one message per burst. The burst arrival time has already conveyed enough information about the arrival time of the packet corresponding to each burst. Accordingly, burst end time might not be needed. |
| CATT | No | We think BET is of no help in support of Survival Time in RAN if Survival Time is triggered based on monitoring transmission failure(s) rather than a PDB timer. BET could be somehow useful on top of BAT for allocating staggered SPS/CG resources in support of flows with some expected non-negligible jitter. But in general we think this should come from SA2 where latest discussion there showed that both Burst Spread and BET are not yet mature enough to make it in the TSCAI. So we could live without it so far and can re-discuss it in the next release. In SA2. |
| Nokia | No | According to the WID, RAN2 should work on RAN enhancement for new QoS parameters decided by SA2, rather than asking SA2 to provide specific information. Therefore, this is beyond the scope from our perspective.  Moreover, given that the gNB already knows BAT and burst size, we think BET can be derived by the gNB, and providing BET information from CN does not seem to be necessary. |
| Lenovo | No |  |
| Qualcomm | No | The scope being discussed here is for RAN to increase link reliability upon a certain failure to avoid an extended time without an application message conveyed in one PDCP SDU. We do not see how Burst Ending Time is relevant to that scope. |
| OPPO | No |  |
| Futurewei | No |  |
| ZTE | No | We don't think there is assumption that one burst contains more than one packets. Even if it’s the case, it can be assumed that they all arrived together. Moreover, the assumption that multiple packets arrive over a period of time may be inconsistent with the understanding for definition of TSCAI. Per our understanding for TSCAI in RAN3 specification, the (burst arrival time (BAT) + periodicity) can be used to deduce the arrival time of the next burst. If a new parameter burst end time (BET) is introduced, would there be any assumption that the end time of the next burst would also be (burst end time + periodicity)? Is it reasonable?  Anyway, we high level agree with above comment that this should be an SA2 discussion. |
| vivo | No |  |

## Other

The scope of this email discussion is intended to be kept rather reduced due to the small time period until the next e-meeting, however if Companies feel Rapporteur missed an important aspect to discuss, please add it below.

**Q12: Do you see another important aspect/topic to discuss in this email discussion?**

|  |  |
| --- | --- |
| **Company** | **Comments** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

3. Conclusion

TBD

4. Reference

1. RAN2#113-e meeting, meeting report;
2. R2-2102074, Offline on RAN enhancements QoS [AT113-e][506]; Nokia
3. 3GPP TS22.104, Service requirements for cyber-physical control applications in vertical domains, V17.4.0
4. TS23.501, System architecture for the 5G System (5GS), V17.0.0, 2021-03
5. R2-2102686, RAN Enhancement to support new QoS, Qualcomm Incorporated;
6. R2-2102726, Handling of Survival Time, CATT;
7. R2-2102993, RAN Enhancement for New QoS Parameters, Nokia, Nokia Shanghai Bell;
8. R2-2103060, RAN enhancements based on new QoS related parameters, Huawei, HiSilicon;
9. R2-2103125, Discussion on RAN enhancement to support survival time, vivo;
10. R2-2103196, Topics on new QoS handling, Fujitsu
11. R2-2103212, RAN enhancement based on new QoS, OPPO
12. R2-2103329, Further considerations on new QoS, ZTE Corporation, Sanechips, China Southern Power Grid Co., Ltd
13. R2-2103420, Discussion on RAN enhancements based on Survival Time, III
14. R2-2103429, RAN enhancements based on new QoS related parameters, Ericsson
15. R2-2103432, Entering and operating in the Survival Time state, Samsung Electronics GmbH
16. R2-2103689, Discussion on the RAN support for new QoS parameters, CMCC
17. R2-2103735, RAN2 Enhancements for Support of QoS Parameters, Intel Corporation
18. R2-2103798, Enhancements based on new QoS requirements, InterDigital
19. R2-2103896, Discussion on entering and exiting survival time state, Futurewei Technologies
20. R2-2104097, View on survival time mechanisms, LG Electronics UK
21. R2-2104225, Clarification on the survival time, Xiaomi Communications
22. R2-2104265, RAN enhancements based on new QoS, TCL Communication Ltd