3GPP TSG RAN WG2 Meeting #115-e R2-17xxxxx

**Electronic meeting, 16th-27th August 2021**

**Agenda item:** x.x.x

**Source:** Intel Corporation

**Title:** Report of email discussion [Post114-e][507][SData] Non-SDT data arrival handling

**Document for:**  Discussion and decision

# Introduction

The intention is to discuss the following topics as part of the email discussion “[Post114-e][507][SData] Non-SDT data arrival handling” taking into consideration the related proposals on RAN2#114e TDocs [1]-[22] and in preparation for responses to be received from SA3 and CT1 in relation to previous RAN2 LSs [23][24].

* [Post114-e][507][SData] Non-SDT data arrival handling (Intel)

**Scope:**

* + Phase 1 (identify the open issues/questions) – 5 days
  + Phase 2 (collect the company views on open issues/questions)
  + Phase 3 (collect companies view on preferred solution CCCH vs. DCCH with the aim to down-select)

**Email discussion to focus on:**

a. Develop details of both solutions (CCCH and DCCH) and identify any further impacts to other WGs (e.g. RAN3)

b. Develop details of how cell reselection could be handled (considering possible repetition of security material) and check if we could agree to support optimised handling of cell reselection

c. Can consider SA3/CT1 discussions into where appropriate .

**Intended outcome**: Report with agreeable proposals

**Email discussion deadline:** August 6th, 0900 UTC

**Note**: silent period is July 5-30 (may be updated during TSG RAN)

The **deadline for this 1st phase** of email discussion is **Friday June 18th, 0900 UTC.**

The **1st phase** of this email discussion provides an overview of the discussion points for the above scenarios that were identified in the contributions. Companies are invited to provide inputs on any missing topics/questions that need to also be included. Issues impacting other groups (e.g., RAN3) can also be listed. These issues are to be addressed in more detail with possible solutions during the 2nd phase.

For completion of the solutions and to acknowledge companies’ inputs, observations (with their corresponding description and references) are also included to capture points that seem straight forward (i.e. the suggestion is not to include a related discussion point for them in the 2nd phase).

# General topics: switch from SDT to CONNECTED

RAN2 has agreed to support transition from SDT session to RRC\_CONNECTED where that SDT session could be ongoing with and without UE AS Context relocation as shown in related agreements below.

* RAN2#113bis: “*UE switches from SDT to non-SDT in following cases: Case 1 (27/0): UE receive indication from network to switch to non-SDT procedure. Network can send RRCResume. FFS whether network can send indication in RAR/fallbackRAR/DCI to switch to non-SDT procedure.”*
* RAN2#112: “*RAN2 confirm that RACH based SDT is supported with and without UE context relocation*”
* RAN2#111: “*Context fetch and data forwarding with anchor re-location and without anchor re-location will be considered. FFS if there are problems with the scenario “without anchor relocation*”

This section addresses general topics for discussion that are therefore applicable to any scenario where the UE with an ongoing SDT session fallbacks into RRC\_CONNECTED regardless of the trigger condition. Some of those trigger conditions may be because network wants to continue the exchange of the SDT data while having the UE in RRC\_CONNECTED or network detects DL non-SDT data or network is informed that UL non-SDT data is available in UE side.

## Topic #1: Handling to switch from SDT to CONNECTED during an ongoing SDT session without UE AS context relocation

Rel-17 SDT WID captures that “*UL small data transmissions for RACH-based schemes (i.e. 2-step and 4-step RACH)*” will be enabled with “*Context fetch and data forwarding (with and without anchor relocation) in INACTIVE state for RACH-based solutions*”. Whether and how to address the scenario when UE transitions into RRC\_CONNECTED during an ongoing SDT session where UE AS context was not relocated is discussed by [2][5][8][19].



Figure 1. Transition into RRC\_CONNECTED during subsequent SDT without anchor relocation [8]

For the scenario explained above and also shown in the Figure 1, the following approaches are suggested:

1. Network release the UE back into RRC\_INACTIVE (potentially with updated suspend and SDT configurations) [5]. This may lead to additional delay (from the release and initiation of a follow up new attempt), however it may not be an scenario that occurs frequently.
2. A new mechanism is defined by RAN2/3 to update the security keys during the ongoing SDT session (due to the relocation of the UE context during an ongoing SDT session) [5][8][19]. The new procedure would be required to support the key change involving: to provide new NCC to the UE, suspend data transfer, resetting L2, re-establish PDCP, Resume data transfer. In addition, RAN2 will need to also discuss which node triggers the anchor change and which node generates the RRC message with the NCC considering the network architecture (including how the CU DU split is done for anchoring). Open questions are also identified by [19][8]:
3. Which node decides the content of *RRCResume* message (anchor gNB vs serving gNB)?
4. Which node performs ciphering and integrity protection for *RRCResume* message?
5. Handling of the security key update e.g.
   * 1. after receiving the second RRCResumeReq from the same UE, will the anchor gNB generate another new KgNB associated with the same target gNB?
     2. how to ensure security key separation if UL/DL data and *RRCResume* message are treated by anchor gNB with updated security key and then, *RRCResumeComplete* message and subsequent UL/DL are treated by the current serving gNB.
6. Which node decodes *RRCResumeComplete* message

Moreover solution details on the questions listed above and how the new mechanism may look like are also provided by [8][19].

1. How to address the scenario when switching from SDT to RRC\_CONECTED during an ongoing SDT session where the UE context was not relocated by the network. Including questions Q1-Q4 for above option 1.b).

## Topic #2: Radio bearer handling when switching from SDT to CONNECTED

RAN2 already agreed that SDT RBs are re-established at initiation of SDT procedure and the new security keys are applied then, i.e. there is no need to perform re-establishment second time when *RRCResume* is received as it is done Rel-15 NR. Therefore, when switching from SDT to CONNECTED, the PDCP of non-SDT RBs do not need to be re-established [12]. For this switch/fallback under network control, it is also explained in [6] that it is left up to network implementation that the data exchanged before triggering the fallback to resume is not lost (i.e. UE does not need to retransmit it) and SDT related data traffic can continue after UE gets RRC\_CONNECTED.

1. When UE receives *RRCResume* message during an ongoing SDT session or in response to *RRCResumeRequest* message sent for SDT (i.e. switch from SDT to CONNECTED), the PDCP entities for only the non-SDT RBs are re-established (i.e., SDT RBs are not re-established as were already resumed for the SDT session).
2. When switching from SDT to CONNECTED, it is left up to network implementation that the data exchanged before triggering the fallback to resume is not lost (i.e. UE does not need to retransmit it) and SDT related data traffic can continue after UE gets CONNECTED.

## New points to section 2

Please indicate if there are any other topics/questions that require to be addressed as part of this **section 2** on “General topics: switch from SDT to CONNECTED”.

|  |  |
| --- | --- |
| **Company’s name** | **Companies’ views** |
|  |  |
|  |  |
|  |  |

# Non-SDT data handling during ongoing SDT session

## General topics

### Topic #3: non-SDT Data available when “starting” an SDT session

Another scenario to address is the desirable behaviour is non-SDT Data is available when initiating an SDT session, i.e. before UE sends the 1st UL RRC message or before contention resolution which is discussed by [16][18][20] It might be important to discuss separately both scenarios:

* Scenario 1) When non-SDT Data becomes available after UE has initiated an SDT procedure, but 1st UL RRC message has not been sent yet. This could be when UE has already sent PRACH preamble when using 4-step RA-SDT.
* Scenario 2) When non-SDT Data becomes available after UE has initiated an SDT procedure and has sent the 1st UL RRC message, but contention resolution has not been received by UE in Msg.4/Msg.B.

1. The expected UE behaviour when non-SDT Data becomes available after UE has initiated an SDT procedure for: scenario 1) 1st UL RRC message has not been sent yet and scenario 2) contention resolution has not been done.

### New points to section 3.1

Please indicate if there are any other topics/questions that require to be addressed as part of this **section 3.1** on “General topics” for “Non-SDT data handling during ongoing SDT session”.

|  |  |
| --- | --- |
| **Company’s name** | **Companies’ views** |
|  |  |
|  |  |
|  |  |

## CCCH-based approach

This section aims to clarify how CCCH-based approach works to enable the switch from an ongoing SDT session to non-SDT operation considering the related inputs provided in [5][9][12][13][14][18][20]. Note that some of the topics are inter-related with the ones discussed as part of previous general sections 2 and 3.1.

### [CCCH point (1)] Detection of non-SDT data

Upon UE detects non-SDT data available during an ongoing SDT session, this section discusses initial UE’s actions associated with CCCH-based approach.

#### [CCCH point (1.1)] UE autonomous release

For CCCH-based approach, It is clarified in [5] that UE autonomously triggers the end or the release of the ongoing SDT session upon detecting the non-SDT data.

1. For CCCH-based approach, UE autonomously triggers the end or the release of ongoing SDT session upon detecting the non-SDT data.

### [CCCH point (2)] RACH, UAC associated with the 2nd resume proc.

It is explained that CCCH-based approach [5] may require additional signalling to the network (e.g. RACH) and applies again UAC same as any UE in RRC\_INACTIVE in order to send non-SDT data

1. When switching from SDT to non-SDT via CCCH-based approach, AS applies UAC and initiates random access procedure same as any legacy UE in RRC\_INACTIVE.

### [CCCH point (3)] Resume cause

This point (3) discuss the proposal that this 2nd *RRCResumeRequest* message uses a new resume cause value (for the network to differentiate it) [20].

1. When switching from SDT to non-SDT via CCCH-based approach, whether a new value of the resume case is defined to differentiate the UE that had an ongoing SDT session and is sending a 2nd *RRCResumeRequest*.

### [CCCH point (4)] PDCP COUNT and/or security key to be used

This section discusses the details of the 2nd *RRCResumeRequest* msg or the corresponding behaviour in UE and network side to support switching to non-SDT data during an ongoing SDT session for CCCH-based approach considering the inputs provided in [4][5][8][9][12][13][14][16][18][20].

References from legacy Resume/Suspend procedure are included below to show that the PDCP COUNT for all RBs are always reset. For example, this is done when *RRCRelease* is received including *suspendConfig* as shown below on the references taken from TS 38.331 § 5.3.8.3 and TS 38.331 § 5.1.4.

**5.3.8.3** **Reception of the RRCRelease by the UE**

The UE shall:

*\*\*\*text omitted\*\*\**

1> if the *RRCRelease* includes *suspendConfig*:

*\*\*\*text omitted\*\*\**

2> suspend all SRB(s) and DRB(s), except SRB0;

2> indicate PDCP suspend to lower layers of all DRBs;

*\*\*\*text omitted\*\*\**

***5.1.4 PDCP entity suspend***

*When upper layers request a PDCP entity suspend, the transmitting PDCP entity shall:*

*- set TX\_NEXT to the initial value;*

*- discard all stored PDCP PDUs;*

*When upper layers request a PDCP entity suspend, the receiving PDCP entity shall:*

*- if t-Reordering is running:*

*- stop and reset t-Reordering;*

*- deliver all stored PDCP SDUs to the upper layers in ascending order of associated COUNT values after performing header decompression;*

*- set RX\_NEXT and RX\_DELIV to the initial value.*

Current Release with suspend and Resume Request procedure resets the PDCP COUNT for all RBs as shown above. Since the CCCH method does not involve the RRC Release message, it is not clear if the COUNT for the RBs is reset for the second RRC Resume request.

On summary, the handling of the PDCP COUNT and the security key when switching from SDT to non-SDT should be clarified for CCCH-based approach as explained by [6][8][12][15][19]. Therefore, CCCH-based approach requires some level of update to resume procedure considering the points here discussed.

1. When switching from SDT to non-SDT via CCCH-based approach, understand whether the PDCP COUNT is (or not) reset.

TS 38.331 § 5.3.1.2 explains that same PDCP COUNT cannot be reused with the same security key as shown below:

“*It is not allowed to use the same COUNT value more than once for a given security key.*”

Note that further related details are discussed in the following sections of this email discussion.

1. The mechanism to be defined that enables the switch from SDT to non-SDT shall meet the following NR requirement: the same PDCP COUNT value is not used more than once for a given security key.

If the PDCP COUNT is reset for the CCCH based approach and as the UE has not received a new key, how to satisfy the above requirements for the RBs should be discussed [6][8][12][15][19].

1. When switching from SDT to non-SDT via CCCH-based approach and if the PDCP COUNT is reset, how to prevent the reuse of the same PDCP COUNT and the same security key for the RBs.

If PDCP count is reset upon detecting non-SDT data during an ongoing SDT session for CCCH-based approach, UE/network may not be able to detect data duplication and prevent data loss after transitioning into RRC\_CONNECTED.

1. When switching from SDT to non-SDT via CCCH-based approach, if the PDCP count is reset upon detecting non-SDT data during an ongoing SDT session, the UE/network may not be able to detect data duplication and prevent data loss during the transitioning into RRC\_CONNECTED.

### [CCCH point (5)] security associated resume MAC-I (dependent on SA3 outcome)

This sub-section discuss how *resumeMAC-I* is calculated for this 2nd *RRCResumeRequest* msg and some of it is dependent on SA3 outcome. For reference, it is shown below related actions from TS 38.331 on this:

*1> set the* ***resumeMAC-I*** *to the 16 least significant bits of the MAC-I calculated:*

*2> over the ASN.1 encoded as per clause 8 (i.e., a multiple of 8 bits) VarResumeMAC-Input;*

*2> with the* ***KRRCint key in the UE Inactive AS Context*** *and the previously configured integrity protection*

*algorithm; and*

*2> with all input bits for COUNT, BEARER and DIRECTION set to binary ones;*

Therefore, the suggested options are in relation to the NCC and the KRRCint key in used.

Irrespective of the security issue that SA3 is addressing, RAN2 can still discuss whether the key used for generating the ResumeMAC-I – for example, is it the one that is currently in use for the SDT data transfer or the one used for generating the ResumeMAC-I of the first ResumeRequest. Proposals were also made to address the security issue related to the reuse of key for generation of the ResumeMAC-I. The proposals for key to use for generating ResumeMAC-I include:

* + 1. NCC provided in last *RRCRelease* message i.e. same as for legacy *RRCResumeRequest* which was also used when the SDT session was started (before initiating ongoing switch to non-SDT) [12].
    2. Horizonal key derivation from current NCC [5] [14] [18].
    3. New NCC that was provided by the serving gNB in the 1st DL message after UE sends the 1st UL SDT msg (i.e. upon initiating the SDT session) [5].
    4. UE’s KRRCint key stored in UE Inactive AS Context i.e. same as for legacy *RRCResumeRequest* which was also used when the SDT session was started (before initiating ongoing switch to non-SDT) [12] (this may depend on the SA3 conclusion).
    5. UE’s new KRRCint key i.e. the one calculated when triggering SDT [12].

1. When switching from SDT to non-SDT via CCCH-based approach, which key is used for generating the *resumeMAC-I* for the 2nd *RRCResumeRequest* msg.

### [CCCH point (6)] Identification of UE AS context in the network

For legacy resume procedure, the release of the UE context from the anchor gNB0 is done upon completion of the path switch with the AMF and the serving gNB. However for SDT option, when to release the UE AS context from the anchor gNB may require further discussion considering inputs in [6][19][11][17]. The possible options are the following (as shown in Figure 2):

1. Same behaviour as legacy resume with the drawback that the I-RNTI stored in the UE during an SDT session points to the UE context in the anchor gNB when this has already been deleted.
2. UE AS Context is released from anchor gNB after serving gNB provides a new NCC and I-RNTI to the UE (i.e. at the end of the SDT session). The drawback is that serving gNB needs to inform anchor gNB when SDT session successfully ends in order to delete the corresponding UE AS Context.



Figure 2. SDT with anchor relocation from [19] but updated to include both options as explained in [6]

Upon initiation of the corresponding SDT session, the I-RNTI stored in the UE points to the UE context in the anchor/old gNB while the UE AS Context may be released from the anchor gNB after serving gNB completes the path switch with the AMF at the beginning of the SDT session.

On summary, during an ongoing SDT session, UE only has an stored/available I-RNTI that points to the anchor gNB. Therefore, if UE had an ongoing SDT session in a serving gNB that is different than the anchor gNB, the identifier to be included in this 2nd *RRCResumeRequest* message and how it can identify the UE context should be discussed. Some possible options also discussed in the contributions were:

1. I-RNTI i.e. same as for legacy *RRCResumeRequest* message which was also used when the SDT session was started (before initiating ongoing switch to non-SDT). This option would route the 2nd *RRCResumeRequest* message to the anchor/old gNB.
2. New I-RNTI that is provided by the serving gNB in the 1st DL message after UE sends the 1st UL SDT msg (i.e. upon initiating the SDT session) [6][14]. This option would route the 2nd *RRCResumeRequest* message to the serving gNB where the SDT session was ongoing.
3. When switching from SDT to non-SDT via CCCH-based approach and for the scenario where the ongoing SDT session is with UE AS context relocation, how serving gNB can locate the UE AS Context in the network for the 2nd *RRCResumeRequest* msg based on the I-RNTI available in the UE that may point to a released UE context in the anchor gNB.

### [CCCH point(7)] Network handling of the 2nd RRCResumeRequest and the RRCResume messages.

When the anchor/old gNB has/keeps stored the UE AS Context, after receiving the 2nd *RRCResumeRequest* msg from the same UE, it needs to be discussed whether anchor gNB generates or not another new KgNB associated with the same target gNB, as explained in Figure 3 below from [18].



Figure 3: Key generation after the second CCCH message [18]

1. When switching from SDT to non-SDT via CCCH-based approach with anchor gNB, after network receives the 2nd *RRCResumeRequest* msg, whether anchor gNB generates (or not) another new KgNB associated with the same target gNB. The following were raised in [18]:

* Will the second RRCResumeReq (i.e., in step 7) be routed to the old anchor gNB regardless of anchor relocation or not?
* After receiving the second *RRCResumeRequest* msg from the same UE, will the anchor gNB generate another new KgNB associated with the same target gNB?

### New points to section 3.2

Please indicate if there are any other topics/questions that require to be addressed as part of this **section 3.2** on “CCCH-based approach” for “Non-SDT data handling during ongoing SDT session”.

|  |  |
| --- | --- |
| **Company’s name** | **Companies’ views** |
|  |  |
|  |  |
|  |  |

## DCCH-based approach

This section aims to clarify how DCCH-based approach work to enable non-SDT switching from an ongoing SDT session considering the inputs provided by companies [3][5][12][18][20][22].

### [DCCH point (1)] Detection of non-SDT data

For DCCH-approach, it is assumed that the PDCP COUNT is maintained as the SDT session fallbacks into CONNECTED upon gNB sends *RRCResume* message during the ongoing SDT session.

1. When switching from SDT to non-SDT via DCCH-based approach, the PDCP COUNT is maintained for SDT DRBs (i.e. the SDT session fallbacks into CONNECTED upon gNB sends *RRCResume* message during the ongoing SDT session).

The sub-section below discusses how/what information the UE shares with the network when non-SDT data becomes available during an ongoing SDT session when using DCCH-based approach.

#### [DCCH point (1.1)] NAS/AS trigger

On new data arrival for the DCCH approach, the following options were proposed:

1. AS triggers the DCCH indication/request from UE to transition into RRC\_CONNECTED when non-SDT data becomes available during the SDT session [18].
2. NAS trigger new DCCH indication/request to AS for UE to move to RRC\_CONNECTED state when non-SDT data becomes available during the SDT session [18].
3. When detecting non-SDT data during ongoing SDT via DCCH-based approach, whether AS layer can initiate the indication/request to the network.

#### [DCCH point (1.2)] non-SDT notification to network

For DCCH-based approach, how UE sends the indication/request to switch into RRC\_CONNECTED when non-SDT becomes available during an ongoing SDT session, the following options were proposed to provide the indication to the network:

1. New UL RRC message [3].
2. Re-using legacy UL RRC message e.g. *UEAssistanceInformation* message [3].
3. For DCCH-based approach, how UE sends the indication/request to switch into RRC\_CONNECTED when non-SDT becomes available during an ongoing SDT session.

Additionally, it was also proposed to provide the following information to the network in the indication:

1. List of one or more RB IDs for which data is arrived [3].
2. Data volume per RB or cumulative can also be indicated [3].
3. For DCCH-based approach, which information is provided by UE to indicate/request the switch into RRC\_CONNECTED when non-SDT becomes available during an ongoing SDT session.

### [DCCH point (2)] switch from SDT to CONNECTED

For DCCH approach, the switch from SDT to CONNECTED is under network control, therefore after UE informs the network that non-SDT data is available, UE continues with the SDT session ongoing until network informs otherwise to UE. For example, network may respond at some point with *RRCResume* or *RRCRelease*. Upon UE receives *RRCResume* message, UE only re-establishes/resumes the non-SDT DRBs (as SDT RBs were already re-established/resumed upon initiating the SDT session) [3][5][12][18].

1. For DCCH approach, after UE informs the network that non-SDT data is available, UE continues with the SDT session ongoing until network informs otherwise to UE (i.e. by transitioning the UE into RRC\_CONNECTED or releasing the UE into legacy RRC\_INACTIVE or RRC\_IDLE). Upon UE receives *RRCResume* message, only the PDCP of non-SDT DRBs are re-established and resumed (as SDT RBs were already re-established/resumed upon initiating the SDT session).

For DCCH-based approach, the topic addressed in Discussion point 1) on how to handle the scenario when switching from SDT to CONNECTED during an ongoing SDT session where the UE context was not relocated by the network at the beginning of the SDT session is also applicable.

1. When switching from SDT to non-SDT (i.e. CONNECTED) via DCCH-based approach and for the scenario where the ongoing SDT session is without UE AS context relocation, previous Discussion point 1) is applicable (and would also need to be addressed).

### New points to section 3.3

Please indicate if there are any other topics/questions that require to be addressed as part of this section 3.3 on “DCCH-based approach” for “Non-SDT data handling during ongoing SDT session”.

|  |  |
| --- | --- |
| **Company’s name** | **Companies’ views** |
|  |  |
|  |  |
|  |  |

# Failure handling during ongoing SDT session

This section aims to understand how failure of an SDT connection could be handled. The main focus of this email discussion is in a failure due to cell reselection during a given SDT session, however companies are invited to provide their inputs in first sections aiming to understand whether a common UE behaviour could be defined regardless of the kind of trigger of the abrupt SDT failure.

## Triggers to an abrupt termination/failure of an SDT session

The following triggers were identified by companies as potential triggers that lead to an abrupt termination or failure of an ongoing SDT session:

1. Cell reselection [4][7][8][9][14][18]
2. Expiry of failure detection timer [4][8][18][20]
3. Lower layers indication [4][9]
4. Maximum number of retransmissions is reached in RLC [15][18]
5. Reject reception during SDT [9]
6. Abortion of connection establishment by upper layers (need FFS) [9]
7. Other events
8. Identify the trigger events that leads to an abrupt termination or failure of an ongoing SDT session.

## UE’s action upon detecting an abrupt termination/failure of an SDT session

It is also discussed by [4][18][8] whether a common UE behaviour when any of the applicable trigger events from previous Discussion point 13) that lead to an abrupt termination/failure of an SDT session.

1. Having a common UE behaviour when any of the applicable trigger events from previous Discussion point 13) lead to an abrupt termination/failure of an SDT session.

The following approaches are proposed when having to handle abrupt termination/failure of an SDT session:

1. UE transitions autonomously into RRC\_IDLE.
2. UE remains in RRC\_INACTIVE.

For example, for cell reselection trigger event, approach 1) is proposed by [16][14] and approach 2) by [7][12][13][18][20][21][4][10]. The following sub-sections aims to clarify how each approach may work taken into consideration the inputs from proposing companies. The following sub-sections aims to clarify how the approach 2) may work taken into consideration the inputs from proposing companies.

### Approach 2) UE remains in RRC\_INACTIVE

This section aims to clarify how approach 2) would work like, i.e. UE remains in RRC\_INACTIVE upon detecting an abrupt termination/failure of an SDT session. The motivation to enable this mechanism is to guarantee data continuity, and minimize/prevent data loss and duplication of the ongoing SDT session. Upon detecting this failure, the UE shall immediately initiate a sub-sequent access. It is important to keep in mind that the feasibility of this approach 2) depends on the inputs to be provided by SA3 in relation to the security related LS [23] sent by RAN2.

1. Upon UE detects an abrupt termination/failure of an SDT session and remains into legacy RRC\_INACTIVE, the UE shall immediately initiate a request to resume the suspended RRC connection or to (re)start the SDT session

This sub-sequent access after the failure (referred for this discussion as “recovery solution”) is explained by supporting companies from different angles and levels of details. For example:

* **Recovery solution 0)** UE suspends SDT DRBs and PDCP entities upon failure and performs SDT/RRC Resume procedure in the reselected new cell [12][20][7][13].
* **Recovery solution 1)** serving gNB provides a new NCC and I-RNTI upon initiating any SDT mechanism (i.e. 1st DL SDT msg) for future use after current SDT session ends or terminates [4][14].
* **Recovery solution 2)** gNB provides new NCC/RNTI immediately after an abrupt termination of the SDT session with UE’s behaviour aligned with RRC Reestablishment message operation continuing the SDT session in RRC\_INACTIVE [4].
* **Recovery solution 3)** UE uses horizonal key derivation for the recovery mechanism after an abrupt termination of an SDT session (where data uses this new key but short MAC-I may still be calculated based on the stored key) [4].
* **Recovery solution 4)** Assuming that SA3 informs that NCC and I-RNTI can be reused in a different cell, the recovery solution for cell reselection case works as follows [14][18]:
  + The *RRCResumeRequest* is routed to the old anchor gNB and the old anchor gNB shall be able to verify the UE and generate new keys irrespective of whether anchor relocation and path switch happens before in the SDT session – needs to be checked with RAN3
  + The new key is derived for the new cell (using the same NCC and the KgNB in the stored UE inactive context, but using new PCI/ARFCN)
  + PDCP based recovery mechanism is used to recover the lost/unacknowledged data whilst the UE Stays in RRC\_INACTIVE state

On summary, the recovery solutions 1)-4) explained above aim to define a mechanism that address the following concerns:

* Security concerns, such as, b.1) if new NCC is not provided after the failure of the SDT session, b.2) the NCC to be used for the follow up *RRCResumeRequest* in the recovery, and b.3) if the Data PDCP COUNT reset as the same key cannot be used with same count.
* Concerns of additional delay or even confusion when looking for the gNB where UE’s context was previously stored. I.e. I-RNTI stored in UE points to the anchor gNB when the new serving gNB has a copy of the UE AS context or is actually already relocated.

These concerns are common to some of the ones discussed in previous section sections, therefore the suggestion is to discuss them for the failure handling scenario keeping in mind the details/options provided in the corresponding previous points.

1. When a UE detects a failure of an ongoing SDT session and remains in RRC\_INACTIVE, UE shall initiate immediately a recovery mechanism (e.g. via SDT or resume) and the following sub-topics needs to also be addressed for this specific scenario (in relation to the 2nd resume procedure):
   1. Previous Discussion point 3) Resume Cause value for the 2nd *RRCResumeRequest* msg.
   2. Previous Discussion point 4) and Discussion point 5) on the PDCP COUNT and/or security key to be used.
   3. Previous Discussion point 6) on which key is used for the generation of the *resumeMAC-I*.
   4. Previous Discussion point 7) on the identification of UE AS context in the network when the ongoing SDT session is with UE AS context relocation.
   5. Previous Discussion point 8) on the network handling of the 2nd *RRCResumeRequest* and the *RRCResume* messages.

## New points to section 4

Please indicate if there are any other topics/questions that require to be addressed as part of this section 4 on “Failure handling during ongoing SDT session”.

|  |  |
| --- | --- |
| **Company’s name** | **Companies’ views** |
|  |  |
|  |  |
|  |  |

# Summary report and proposals

<Section to be updated by Rapporteur>

This report summarizes the views of xx companies ().

Aiming to help with the meeting discussion/progress, the proposals are categorized starting with:

* [To agree] when there is large support and hence proposed for easy agreement.
* [To discuss] when there is substantial level of support and agreement may be possible.
* [FFS] when there is low support or companies propose new solutions or options to possibly consider further e.g. if there is sufficient support (understanding that these topic have not been discussed by all companies when providing their views in the different discussion points).

The proposals also start with a number: for the format [x], ‘x’ represents the number of supportive companies (i.e. these solutions are marked as FFS as the proposed solutions were not discussed by all companies) and, for the format [x/y], ‘x’ represents the number of supportive companies, and (y-x) the number of companies with different view.

1. **[To agree]** xxx
2. **[To discuss]** xxx
3. **[FFS]** xxx
4. xxxx.

# Conclusion

The observations captured are the following:

**Observation 1.** xxxx.

The proposals captured are the following:

**Proposal 1.** xxx

The following list shows the proposals above organized based on the suggested priority aiming to help during its meeting discussion:

**Proposals for easy agreement**

xxx

**Proposals for discussion (1st priority) or to be captured as FFS**

xxx

**Proposals for discussion (2nd priority) or to be captured as FFS**

xxx

# Annex: companies’ point of contact

|  |  |  |
| --- | --- | --- |
| **Company** | **Point of contact** | **Email address** |
| Intel Corporation | Marta Martinez Tarradell | marta.m.tarradell@intel.com |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Reference

1. R2-2104771, Discussion on common control plane issues of SDT, OPPO
2. R2-2104772, on RACH-based SDT, OPPO
3. R2-2104785, Control Plane Common Aspects of RACH and CG based SDT, Samsung Electronics Co., Ltd
4. R2-2104881, Failure and successful handling for an SDT session, Intel Corporation
5. R2-2104882, CP-SDT remaining open issues, Intel Corporation
6. R2-2104883, RA-SDT remaining open issues, Intel Corporation
7. R2-2105101, Control plane aspects on the SDT procedure, Apple
8. R2-2105281, Consideration on CP issues, CATT
9. R2-2105448, Control plane aspects of SDT, NEC
10. R2-2105549 on RACH-based SDT, Spreadtrum Communications
11. R2-2105574, Small data transmission with RA-based schemes, Huawei, HiSilicon
12. R2-2105575, Control plane common aspects for SDT, Huawei, HiSilicon
13. R2-2105691, Discussion on subsequent SDT in NR, timer handling, and support for SRB1/2, Sony
14. R2-2105760, Common aspects for SDT, Ericsson
15. R2-2105810, Consideration on CP issues for small data transmission, Lenovo, Motorola Mobility
16. R2-2105885, Discussion on open issues of SDT, Qualcomm Incorporated
17. R2-2105886 on open issues for RACH based SDT, Qualcomm Incorporated, R2-2103433
18. R2-2105928, Control plane common aspects of SDT, ZTE Corporation, Sanechips
19. R2-2105929, Open issues for RACH based SDT, ZTE Corporation, Sanechips, Rel-17
20. R2-2106050, SDT CP and configuration aspects, InterDigital
21. R2-2106132, Discussion on CP aspects of SDT, China Telecomunication Corp.
22. R2-2106256, Anchor relocation and context fetch, CMCC
23. R2-2104401, LS to SA3 on Small data transmissions, Interdigital, April 2021.
24. R2-2104644, LS to CT1 on Small data transmissions (Ccing SA2), Intel Corporation, April 2021.