3GPP TSG-RAN WG2 Meeting #115-e R2-21xxxxx

Electronic Meeting, xxxx, 2021

**Agenda item: 8.8.3**

**Source: CMCC**

**Title: Report for [Post114-e][252][Slicing] RACH partitioning details for slicing**

**WID/SID: NR\_slice**

**Document for: Discussion and Decision**

# Introduction

In RAN2#114-e[1], the following email discussion was allocated for RACH partitioning:

* [Post114-e][252][Slicing] RACH partitioning details for slicing (CMCC)

      Scope: Discuss the configuration details RACH partitioning: What is the configuration needed for slice-specific RACH? Which parameters need to be separated for slices (or slice groups)? How does the RACH prioritization work with existing RACH prioritization (e.g. MPS/MCS)? What information is needed to help design the "common" Rel-17 RACH prioritization scheme?

      Intended outcome: Discussion report (may include also draft CRs if there is enough convergence)

      Deadline: Long (August 6th, 0900 UTC)

Company contact details:

|  |  |  |
| --- | --- | --- |
| Company | Name | Email Address |
| Qualcomm | Peng Cheng | chengp@qti.qualcomm.com |
| Fujitsu | Ohta Yoshiaki | ohta.yoshiaki@fujitsu.com |
| Intel Corporation | Seau Sian Lim | seau.s.lim@intel.com |
| Samsung | Hyunjeong Kang | hyunjeong.kang@samsung.com |
| Xiaomi | Xiaofei Liu | liuxiaofei@xiaomi.com |
| Apple | Yuqin Chen | yuqin\_chen@apple.com |
| Huawei, HiSilicon | Jun Chen | jun.chen@huawei.com |
| Nokia, Nokia Shanghai Bell | Malgorzata Tomala | malgorzata.tomala@nokia.com |
| CMCC | Ningyu Chen | chenningyu@chinamobile.com |
| ZTE | Yuan Gao | gao.yuan66@zte.com.cn |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Discussion

## Issue 1: slice info for RACH configuration

RAN2#113b-e has agreed below agreement with FFS on slice group details:

**=>Slice specific RACH is only applicable if there is slice information (e.g., slice group or slice related operator defined access category) available for AS layer when access. FFS on details of slice group.**

At RAN2#113b-e meeting, [51] summarized slice info for cell reselection, and slice group (17 companies supported) and SST (7 companies) had more supports than other candidates. In the contributions of RAN2#114-e meeting, potential solutions (e.g., Slice access category, SST and Slice group) were raised.

Option 1: New slice grouping [2], [45]

Option 2: SST [6]

Option 3: UAC category [8]

Companies are invited to share views on which slice info should be used for RACH configuration.

**Q1.1: Which option do you prefer?**

|  |  |  |
| --- | --- | --- |
| Company | Option | Comments |
| Qualcomm | Option 1 | Because WI phase just started, we prefer a clean solution from beginning, i.e. Option 1.  For Option 2 (SST), it is not flexible. And we think it can’t support one important scenario for slicing enhancement: operator has requirement to prioritize some dedicated eMBB slice (i.e. eMBB slice paid by OTT for higher QoS) over common eMBB slice.  For Option 3 (UAC), we think it is not a clean solution due to below impacts to UAC:   * Access category was not designed to indicate slice info. So, there is not 1:1 mapping. Then, some slice info may not be derived if they belong to same access category (e.g. some paid/dedicated eMBB slices on top of common eMBB slices) * Not all the S-NSSAIs belonging to one access category can be supported by gNB, which may cause misunderstanding between UE and gNB on the supported slice. * UAC is a PLMN concept used for access control. According to TS 38.331, if a slice is not supported by a certain cell, the relevant access category has to be included in SIB message of all cells in the PLMN, which will bring significant SIB overhead and impacts on the network side. |
| Fujitsu | Option 2 | It would be good reuse existing parameter as much as possible. SST is composed of 8 bits (256 values) and only 2 bits (4 blues) are used. Slice group can be coded into the remaining bit with which S-NSSAIs can be mapped. The mapping can be configured by NAS. |
| Intel | Option 1 | SST on its own is not sufficient to differentiate the different slices and will need to be combined in some form with the SD. UAC category is not suitable since it is for access control purpose and does not provide the control needed for frequency prioritisation without sacrificing the access control behaviour. |
| Samsung | Option 1 | This option is simple. Others like SST, UAC category require additional clarification. For example a SST can be further partitioned into one or more SDs. So it need clarification whether all SDs of the same SST are classified into the same slice group or different slice groups. Since UAC and slice specific RACH are independently operated, it is better not to mix the usage of UAC category. |
| Xiaomi | Option2 | For option3 (UAC), we agree with QC that we should not have impacts on current UAC mechanism.  For option1 and option2, we think option2 (SST) should be considered first as it has minor impacts on spec and no need for extra signalling to provide the mapping of slice and slice group. For the slices with the same SST but different SDs, as they can provide similar service, we think it is a reasonable assumption that all these slices can be supported by the same cell/frequency under well deployment. |
| Apple | Option 1 | We think Option 1 is the cleanest one.  For Option 2, we don’t feel SST itself can flexibly manage the slice groups demanded by operators.  For Option 3, we recall that RAN2/SA2 has discussed to use the operator defined access categories to achieve slice specific UAC configuration. If that is possible, we believe it makes sense to use UAC scheme for slice grouping. Nevertheless, we also see the difficulty in terms of signaling design, i.e, how to associate the access categories with slice and slice group. |
| Huawei, HiSilicon | Option 1 | At RAN2#113b-e meeting, 17 companies supported option 1. In addition, slice group is applicable for both slice specific RACH and slide specific reselection solutions. |
| Nokia | Option 3 | In our understanding the agreement:  **Slice specific RACH is only applicable if there is slice information (e.g., slice group or slice related operator defined access category) available for AS layer when access.**  should hold even for a single slice.  We believe slice info should be used for RACH configuration can refer to a single slice. A single slice can be realized by Option 2 or Option3.  As per TS24.501 (section 4.5.3) Operator Defined Access Categories can be already associated with S-NSSAI. Since UAC provides Access Category to AS this information can be easily used for slice specific RACH parameters selection for that particular Access Category. The other advantage of using UAC is that it can be used in other cases, not just for slice specific RACH.  As per Rel-16, pure SST that is not available to AS. Providing slice RACH specific per SST would require additional exchange with NAS to make the UE’s AS aware of SST.  For any of the abovementioned Options, grouping can be basically achieved by listing which slice reference ids (Access Category range or list if SSTs).  Option 1 introduces a new concept of grouping that will require impacts to AS-NAS exchange and pure RRC procedures, while it seems to complicate slice specific RACH configuration for a single slice. |
| CMCC | Option 1 | Introducing new slice grouping is a clean way.  SST is not preferred, since SST may be not sufficient to differentiate slices.  As for UAC, we think the grouping rule may be difference. Slices or APPs that share similar AC barring parameters will probably classified into the same access category. But for slice based cell reselection or RACH configuration, the slices that sharing similar reselection rule or RACH configuration should be classified into same group, which is different from the UAC. |
| ZTE | Option3 | * For option 1, we understand the slice grouping should be negotiated between UE and the CN via NAS signaling thus SA2 and CT1 impact is foreseen. * For option 2, broadcasting SST would be too broad for UE to decide whether to access this cell as there would be at most 16777 216 slices with different Slice Differentiators for each SST. * Option 3 is preferred from our perspective to address the SIB payload size concern and security concern while minimizing the impact in SA2/CT1 specs. Regarding QC’s comments, we would like to further clarify:   + Access category was not designed to indicate slice info does not mean it cannot be used to indicate slice info. The existing NAS signalling allowed 1:N and 1:1 mapping between AC and S-NSSAI with full flexibility. Using the existing procedures as much as possible to minimize the impact is what we are always trying to do.   + Using AC to indicate the slice info does not mean NW has to broadcast the access control information (e.g. the barring factor and timer) associated with this AC. NW has full flexibility to decide whether to broadcast the access control information or not as it is broadcast per AC.   + The NAS signalling to configure the mapping between AC and slices is reused in option3. When configuring slice specific RACH resources and prioritization, the AC would be used in a similar way as the slice group ID. In other words, the AC is a type of slice group ID with an existing NAS signalling to configure the mapping information. |

If option 1 slice grouping is preferred, a following up question is how to configure the mapping information (i.e., mapping between S-NSSAIs and slice groups) to UE.

**Q1.2: Do you agree the mapping between S-NSSAIs and slice groups should be configured to the UE? Does the configuration go with AS signalling or NAS signalling?**

|  |  |  |  |
| --- | --- | --- | --- |
| Company | Yes/No | AS or NAS | Comments |
| Qualcomm | Yes | NAS | * Slice grouping should be UE dedicated. NAS signalling is more suitable to support UE differentiation on slice supporting. * Following homogeneous slice assumption in TA, slice grouping configuration should remain unchanged when not moving out of TA. Then, NAS signalling naturally makes UE to only update slice grouping in TAU. * For RACH, it needs to work for IDLE/INACTIVE UE. It implies that SIB needs to broadcast slicing grouping configuration if AS signalling is introduced. It will have significant overhead of SIB broadcasting. |
| Fujitsu | Yes | NAS | The configuration can be dealt with the same layer with the layer that deals with S-NSSAI. |
| Intel | Yes | NAS | With the assumption that slice availability is homogeneous within a TA, it will be easy for the slice group mapping to its S-NSSAI to be provided via NAS signalling (i.e. during initial registration and mobility registration/TA update) to the UE. |
| Samsung | Yes | NAS |  |
| Xiaomi | Yes | NAS |  |
| Apple | Yes | NAS |  |
| Huawei, HiSilicon | Yes | NAS |  |
| Nokia |  | NAS | In any case a grouping of slices for RACH prioritization should be based on available (group of) slices from NAS. |
| CMCC | Yes | NAS |  |
|  |  |  |  |

## Issue 2: RACH prioritization

The collision between slice based prioritization and the legacy MPS/MCS prioritization has been raised for several meetings.

In RAN2#113bis-e meeting, during the email discussion R2-2104322, companies shared views on the options.

* Option 1 (clearly specified): 14 companies
  + Option 1a (Slice override MPS): 7 companies
  + Option 1b (MPS override slice): 6 companies
  + Option 1c (select most beneficial parameters) , e.g. min {slice specific scalingFactorBI, MCS/MPS specific scalingFactorBI} and max {slice specific powerRampingStepHighPriority, MCS/MPS specific powerRampingStepHighPriority}: 1 company
* Option 2 (configurable by network): 13 companies

And the following conclusion was agreed in RAN2#113bis: *RAN2 confirms that the issue of prioritization parameter collision with MPS/MCS need to be resolved. There is UE based solution (option 1, fixed rule) or network based solution (option 2, configurable rule) or both. Discussion on pros and cons can be left to next meeting.*

**Option 1: UE based rule, i.e., Either slice override MPS, MPS override slice or UE select the most beneficial parameters**

**Option 2: Network based solution, i.e., Network indicates whether slice override MPS or MPS override slice.**

**Q2.1: Which option do you prefer?**

|  |  |  |
| --- | --- | --- |
| Company | Option | Comments |
| Qualcomm | Option 2 | Considering RAN2 is introducing RACH prioritization for different scenarios / cases ever from Rel-15 to Rel-17 (BFR/HO → MPS/MCS → Slice), we tend to think specifying a flexible / configurable way is more forward compatible way. This priority can be configured by gNB or be pre-configured via UE’s subscription |
| Fujitsu | Option 2 | NW configuration would be the baseline to control the priority. |
| Intel | Option 2 |  |
| Samsung | Option 2 | The RACH resource utilization and scheduling is up to NW. |
| Xiaomi | Option 2 |  |
| Apple | Option 2 |  |
| Huawei, HiSilicon | Option 2 |  |
| Nokia | Option 2, but | NW can handle proper RA prioritization by configuring MPS/MCS or slices accordingly – no need to indicate anything new. |
| CMCC | Option 2 |  |
| ZTE | Option 1,  UE select the most beneficial parameters | Having one always overrides the other does not seem to be fair as it is hard to tell which is more important (access from a MPS/MCS UE or access via a certain slice which requires low latency). Which one to prioritize would also be a headache for NW if we make it configurable.  Considering that a MPS/MCS UE access via a slice with low latency requirements can be treated as a supper VIP, we would prefer to let it select the most beneficial parameter and get access to NW as soon as possible. And the same rule can apply in the future with minimum spec impact if RACH prioritization for more scenarios and cases are introduced. |

**Q2.2: If you prefer Option 2, do you think UE based rule also needs to be specified when network indication is not available?**

|  |  |  |
| --- | --- | --- |
| Company | Yes/No | Comments |
| Qualcomm | Yes (slightly) | It is also possible that no priority value is (pre)configuration, especially for legacy system. In this case, we prefer to also specify a default rule.  If majority think network indication should be always available, we can also follow majority. |
| Fujitsu | No | Default configuration would be considered. |
| Intel | No | It can just be left to UE implementation since it just means that network has no preference. |
| Samsung | No | We prefer to define one mechanism. |
| Xiaomi | No |  |
| Apple | No | NW is expected to always provide the override indication together the slice prioritization configuration. |
| Huawei, HiSilicon | No | It can be left to UE implementation if no network indications is sent. |
| Nokia | No | Network based mechanism gives clear handling of any potential overload situation |
| CMCC | Yes | The default configuration should be specified. |
|  |  |  |

**Q2.3: If you prefer Option 1 or “Yes” for Q2.2, do you prefer [Option a] slice override MPS or [Option b] MPS override slice or [Option c] UE select the most beneficial parameters?**

|  |  |  |
| --- | --- | --- |
| Company | Option | Comments |
| Qualcomm | Option a | To guarantee the fairness among UEs initiating the same slice, slice specific RA prioritization parameter should override MPS/MCS specific RA prioritization parameter |
| CMCC | Option a | Same view as Qualcomm |
| ZTE | Option c | Having one always overrides the other does not seem to be fair as it is hard to tell which is more important (access from a MPS/MCS UE or access via a certain slice which requires low latency). Which one to prioritize would also be a headache for NW if we make it configurable.  We prefer UE to select the most beneficial parameters, e.g. min {slice specific scalingFactorBI, MCS/MPS specific scalingFactorBI} and max {slice specific powerRampingStepHighPriority, MCS/MPS specific powerRampingStepHighPriority}. |
|  |  |  |
|  |  |  |

*scalingFactorBI* and *powerRampingStepHighPriority* are the baseline parameters for slice based RACH prioritization. Companies are invited to share views whether there is any other parameter should be configured for slice based RACH prioritization.

**Q2.4: whether there is any other parameter should be configured for slice based RACH prioritization? If Yes, please list the proposed parameters. If No, it means we will stick to the baseline parameters.**

|  |  |  |
| --- | --- | --- |
| Company | Yes/No | Comments |
| Qualcomm | Deprioritize | Because legacy RACH prioritization only includes *scalingFactorBI* and *powerRampingStepHighPriority.* We are not sure whether any legacy impact if other parameters are introduced for slice RACH prioritization. Thus, we think RAN2 can first take the two parameters as baseline. We are open to discuss other parameters case by case after all issues of baseline are finalized. |
| Fujitsu | Yes/No | The baseline would be the parameters indicated above.  One new parameter which may worth considering would be “congestion” level of RAN slice. It can be used for load control of slice-specific RACH access. If this is worth considering, it may affect fallback mechanism of RACH. |
| Intel | No | For other parameters (e,g, making slice specific RSRP threshold for RACH type selection, reducing the number of attempts for 2-step RACH etc.), it is unclear how they can achieve faster access. |
| Samsung | No |  |
| Xiaomi | No |  |
| Apple | No |  |
| Huawei, HiSilicon | No |  |
| Nokia | No |  |
| CMCC | No |  |
| ZTE | No |  |

## Issue 3: RACH type selection

How to perform RACH type selection (e.g., slice-specific and common, 2-step and 4-step), if the 2-step and 4-step RA resources are configured?

Option 1: UE first selects between slice-specific and common RACH, then selects between 2-step and 4-step[45]

Option 2: UE first selects between 2-step and 4-step, then selects between slice-specific and common RACH[48]

**Q3.1: Which option do you prefer?**

|  |  |  |
| --- | --- | --- |
| Company | Option | Comments |
| Qualcomm | Option 1 | We think it doesn’t make sense that Network reserved isolated RACH resource for some slice traffic, but the UE selects common RACH just based on RSRP. It is conflicted with the intention to introduce slice specific RACH. Thus, we prefer the UE to first selection between slice specific RA and common RA. |
| Fujitsu | Option 1 | The intention of slice-specific RACH is fast access for the UE to the intended RAN slice. Option 1 can achieve this purpose. Note that 2-step RACH and 4-step RACH are methodology for the fast access to the Uu interface, and not the fast access to the service. |
| Intel | Option 1 | Our understanding of introducing slice specific RACH is to reduce the access latency for critical slice to achieve fast access. Hence if a slice/slice group is configured to use slice/slice group specific RACH, it should select slice specific RACH. |
| Samsung | Option 1 | If slice-specific RACH resources are configured, UE should use the configured slice specific RACH resources. |
| Xiaomi | Option2 | Option 1 seems more complexity to UE compared with option2.  For example, in the case of 2-step slice-specific RACH resource and 4-step common RACH resource configured, UE will first select 2-step slice-specific RACH resource without RSRP checking. However, in this case, it is quite possible that UE failed to access due to bad radio condition and fallback to 4-step common RACH, in this case, it can introduce extra procedure and access delay.  For option2, in the above case, UE firstly perform 2-step/4-step RA type selection and will select 2-step slice-specific RACH resource only when the RSRP threshold is met, otherwise, UE will select 4-step common RACH resource to initiate access to avoid extra fallback and access delay. |
| Apple | Option 1 |  |
| Huawei, HiSilicon | Option 1 | Option 1 is clear and simple. |
| Nokia | Option 1 | Selection between slice specific RA and common RA at first place, makes sense. |
| CMCC | Option 1 | If slice specific RACH resources are configured, network is expecting the UE to select only within the configured resources for that slice. |
| ZTE | Option 1 |  |

In the contributions of RAN2#114-e, companies discussed whether to introduce a new RSRP threshold or reuse the legacy threshold for the selection between 2-step and 4-step slice initiated RACH:

Option 1: A new threshold [10, 45]

Option 2: Reuse legacy threshold [2, 6]

Companies are invited to share views on the above two options.

**Q3.2: Which option do you prefer, regarding to whether to introduce a new threshold or not for slice initiated RACH?**

|  |  |  |
| --- | --- | --- |
| Company | Option | Comments |
| Qualcomm | Option 2 | We are not convinced to introduce a new threshold:   * The intention of RSRP threshold is to avoid UE with poor coverage performing 2-step RACH, especially for PUSCH decoding of MsgA. However, slice RACH isolation doesn’t change UE’s coverage status. * Legacy RSRP threshold is included in SIB1 because it is an essential IE for UE’s first RACH attempt. Then if we introduce slice specific threshold, it will increase payload size of SIB1, especially when multiple slice groups are configured |
| Fujitsu | Option 2 | 2-step RACH and 4-step RACH are methodology for the fast access to the Uu interface. The existing threshold is designed as such considering Un access performance. The intention of introduction of a new threshold is unclear and it may cause problem with Un access performance. |
| Intel | Option 1 (with comments) | A new threshold for the slice specific 2-step and 4-step RA resources selection may be a cleaner solution and provide more flexibility for the network for RACH congestion control purpose. However, the new threshold is a common threshold applied by different slices/slice groups that are configured to use slice specific RACH. |
| Samsung | Option 2 | It is not needed to use different RSRP values for slice specific RACH and non-slice specific RACH since 2-step RACH is not applicable for UE in poor coverage. |
| Xiaomi | Option 2 | As legacy RSRP threshold is not configured different value for different service type in legacy mechanism, thus we think for slice based RACH type selection rule, there is no need to introduce slice-specific RSRP threshold to reduce the impacts on current spec. |
| Apple | Option 2 | We don’t see the intention of introducing a new RSRP threshold as this is mainly to differentiate whether UE is in a good coverage, which does not change at all with or without slice specific RACH resource. |
| Huawei, HiSilicon | Option 2 | It is sufficient to re-use the legacy threshold |
| Nokia | Option 2 |  |
| CMCC | Either option 1 or 2 is ok | No strong preference. |
| ZTE | Neutral | We tend to agree that having slice specific threshold offers more flexibility for NW to control the access. Also there may be some slices with higher requirements on the RSRP when 2-step RACH is used.  We are also fine to reuse the existing threshold if the majority prefers to do so for simplicity. |

During the offline discussion in the last meeting, several issues are raised and need to be further discussed. And the table 1 from R2-2104322[2] can be used as a starting point for discussion.

Table 1. Fallback cases from R2-2104322[2]

|  |  |  |  |
| --- | --- | --- | --- |
| **Cases** | **RACH resource configuration in one BWP** | **RACH type selection for slice triggered access** | **Fallback after MSGA or MSG1 attempt number beyond threshold** |
| Case 1 | 2-step slice specific RACH  4-step common RACH | FFS Always perform 2-step slice specific RACH | Fallback to 4-step common RACH |
| Case 2 | 2-step slice specific RACH  4-step slice specific RACH  4-step common RACH | RACH type selection based on RSRP threshold | Fallback to 4-step slice specific RACH.  FFS Fallback from 4-step slice specific RACH to 4-step common RACH |
| FFS Case 3 is valid | 4-step slice specific RACH  2-step common RACH | FFS Always perform 4-step slice specific RACH | FFS:  No fallback vs. Fallback to common RACH |
| Case 4 | 4-step slice specific RACH  4-step common RACH | Always perform 4-step slice specific RACH | FFS:  No fallback vs. Fallback to common RACH |
| Case 5 | 2-step slice specific RACH  2-step common RACH  4-step slice specific RACH  4-step common RACH | RACH type selection based on RSRP threshold | Fallback to 4-step slice specific RACH.  FFS Fallback from 4-step slice specific RACH to 4-step common RACH. |
| FFS  Case 6 is valid | 2-step slice specific RACH  2-step common RACH | Always perform 2-step slice specific RACH | FFS:  No fallback vs. Fallback to common RACH |
| Case 7 | 2-step slice specific RACH  2-step common RACH  4-step common RACH | FFS Always perform 2-step slice specific RACH | Fallback to 4-step common RACH.  No fallback to 2-step common RACH. |
| FFS  Case 8 is valid | 4-step slice specific RACH  2-step common RACH  4-step common RACH | FFS Always perform 4-step slice specific RACH | FFS Fallback from 4-step slice specific RACH to 4-step common RACH. |

Regarding whether the FFS cases are valid, some companies have concern on the validity for case 3/6/8, while some other companies support to have some flexibility for network configuration.

**Q3.3: Do you have concern to support case 3/6/8 in specification?**

|  |  |  |
| --- | --- | --- |
| Company | Concern for case 3/6/8? | Comments |
| Qualcomm | No | We agree that some cases (e.g. Case 3) are strange and should not be preferred. However, these 3 cases are valid Network configuration. Generally, we don’t make restriction on Network configuration but leave the flexibility to Operators. We tend to keep this general principle. |
| Fujitsu | No | There is no need to restrict NW configuration. It would be up to NW implementation. |
| Intel | See comments | For Case 3 and 6, only 2-step common RACH is configured. If the discussion is for idle/inactive mode UEs, 4-step common RACH needs to always be supported in initial BWP for legacy UE. Hence such cases may not occur. If the discussion is also for connected mode, then Case 3 and 6 can be possible configuration for non-initial BWP.  For Case 8, from the fast access point of view which slice specific RACH is introduced, it seems a bit strange that 2-step slice specific RACH is not introduced while 2-step common RACH is available. |
| Samsung | No | We think that some configurations (no 4 step common RACH configuration as case 3, case 6) are strange but we can rely on NW implementation. |
| Xiaomi | No | Up to NW implementation. |
| Apple | In general No | It’s better to allow flexible NW configurations.  Nevertheless, we also share the concern raised by Intel on case 3 and case 6 that 4-step common RACH should be there for idle/inactive state. |
| Huawei, HiSilicon | No | Case 3/6/8 are not common cases. |
| Nokia | Yes | We think 4-step common RACH needs to always be supported in initial BWP for legacy UE, thus we don’t understand why to assume such restrictions. |
| CMCC | No | We would like to leave the flexibility for operator configuration. |
| ZTE | See comments | Agree with Intel that case 3 and case 6 applies to non-initial BWP for UE in connected. Case 8 is fine. |

For the fallback mechanism, fallback from 2-step slice specific RA to 4-step slice specific RA is naturally supported, similar to the legacy mechanism. The key issue is whether the UE can fallback from slice specific RACH to common RACH. According to the table above, there are 3 open cases. Companies are invited to share views on whether these cases need to be supported

Fallback case 1: Fallback from 4-step slice specific RACH to 4-step common RACH

Fallback case 2: Fallback from 2-step slice specific RACH to 4-step common RACH, if 4-step slice specific RACH is not configured.

Fallback case 3: Fallback from 2-step slice specific RACH to 2-step common RACH, if neither 4-step slice specific RACH nor 4-step common RACH is configured.

**Q3.4: Do you support any of the above fallback cases?**

|  |  |  |
| --- | --- | --- |
| Company | Which fallback case do you support? | Comments |
| Qualcomm | Fallback case 2 | For Fallback from specific RACH to common RACH with same RACH type (i.e. case 1 and 3), we are not convinced with its benefit. Unless UE can know heavier congestion on slice specific RACH resource than common RACH. However, the UE doesn’t know the load difference from common RACH. We tend to simplify the procedure.  We support Fallback case 2 because 2-step slice specific RACH may be failed because of radio condition or Network load. In such case, fallback to 4-step common RACH will help. Furthermore, considering 4-step slice specific RACH is not configured, we have nothing loss if trying 4-step common RACH. |
| Fujitsu | Yes | For now, we prefer to keep it as it is. First, we think that the fallback can be discussed later after many details are settled down. As discussed in this email discussion, slice info (which is also discussed in email discussion 251), RACH parameters and procedure including new threshold are still discussions. Once the details are settled down, fallback would be looked at. |
| Intel | Fallback case 2 | For Case 1 and 3, we do not see the further gain to fallback to 4-step common RACH after failing 4-step slice specific RACH. If it is due to RACH congestion, the chances of this will be quite low since the UE can perform multiple attempts on slice specific RACH and UAC can be used in such cases. |
| Samsung | Fallback case 2 |  |
| Xiaomi | All | For case1 and 3, as the fundamental intention of slice based RACH configuration is to guarantee UE fast access, thus if UE has failed to access on slice-specific RACH resource, it should be allowed to use common RACH resource to initiate access attempt other than just wait.  For case2, we have agreed that “Legacy 2-step RA fallback mechanism is supported.” thus, if UE failed to access based on 2-step RACH resource while 4-step RACH resource is configured, it can fallback to 4-step RACH. If there is no 4-step slice-specific RACH resource configured, it can try to initiate access based on 4-step common RACH resource. |
| Apple | Fallback case 2 |  |
| Huawei, HiSilicon | Fallback case 2 |  |
| Nokia | Fallback case 2 | It would follow the default approach that 2-step can fallback to 4-step.  Slice specific 2-step could fallback to 4-step common resources, it should be however further discussed, under what conditions. |
| CMCC | Fallback case 2 | Generally, we prefer to keep the solution simple. Fallback case 2 can address the issue of lack of coverage for 2-step RA. While fallback case 1 and 3 seems not so essential. |
| ZTE | All | 1. We can start from the most complex case when all kinds of RACH resources are configured:  * 2-step slice specific RACH * 4-step slice specific RACH * 2-step common RACH * 4-step common RACH   We understand it is also related to whether we will introduce slice specific RSRP threshold for 2-step RACH selection.  If the existing RSRP threshold is reused (i.e. one single RSRP threshold for RACH type selection), then the following fall back route applies:  2-step slice specific RACH -> 4-step slice specific RACH-> 2-step common RACH -> 4-step common RACH.  4-step slice specific RACH -> 4-step common RACH  If slice specific RSRP threshold is introduced, the following fall back route applies:  2-step slice specific RACH -> 4-step slice specific RACH -> 2-step common RACH -> 4-step common RACH  2-step slice specific RACH -> 4-step slice specific RACH -> 4-step common RACH  4-step slice specific RACH -> 2-step common RACH -> 4-step common RACH  4-step slice specific RACH -> 4-step common RACH   1. For other cases when part of the RACH resources types are configured, we can just delete the RACH resource type not configured from the above route.  * 2-step slice specific RACH * 2-step common RACH * 4-step common RACH   For example, if 4-step slice specific RACH resource is not configured, then the fall back routes turns into the following:  One single RSRP threshold applicable for all:  2-step slice specific RACH ~~-> 4-step slice specific RACH~~-> 2-step common RACH -> 4-step common RACH.  ~~4-step slice specific RACH -> 4-step common RACH~~  Slice specific RSRP threshold introduced:  2-step slice specific RACH ~~-> 4-step slice specific RACH~~ -> 2-step common RACH -> 4-step common RACH  2-step slice specific RACH ~~-> 4-step slice specific RACH~~ -> 4-step common RACH  ~~4-step slice specific RACH -> 2-step common RACH -> 4-step common RACH~~  ~~4-step slice specific RACH -> 4-step common RACH~~ |

# Summary

TBD**.**

# References

*RAN2#114-e*

1. R2-2104740 Further discussion on slice specific cell reselection Qualcomm Incorporated
2. R2-2104741 Further discussion on slice specific RACH Qualcomm Incorporated
3. R2-2104743 Draft LS to SA2 on slice grouping and slice priority Qualcomm Incorporated
4. R2-2104744 Discussion on SMBR enforcement Qualcomm Incorporated
5. R2-2104782 Considerations on slice based cell reselection Beijing Xiaomi Software Tech
6. R2-2104789 Considerations on slice based RACH configuration Beijing Xiaomi Software Tech
7. R2-2104791 Discussion on slice aware cell reselection ZTE corporation, Sanechips
8. R2-2104792 Slice specific RACH resources and RACH prioritization ZTE corporation, Sanechips
9. R2-2104873 Frequency prioritization for slice specific cell (re)selection Intel Corporation
10. R2-2104874 Further considerations of slice based RACH Intel Corporation
11. R2-2105109 Discussion on slice based cell reselection Apple
12. R2-2105110 Discussion on slice based RACH Apple
13. R2-2105203 Discussion on frequency priority for inter-frequency cell reselection China Telecommunication
14. R2-2105212 Further discussion on slice-based cell reselection Lenovo, Motorola Mobility
15. R2-2105213 Further discussion on slice-based PRACH configuration Lenovo, Motorola Mobility
16. R2-2105239 Discussion on Uplink SMBR enforcement Nokia, Nokia Shanghai Bell
17. R2-2105240 Slice specific cell reselection Nokia, Nokia Shanghai Bell
18. R2-2105331 Discussion on slice-based reselection vivo
19. R2-2105332 Discussion on slice-based RACH configuration vivo
20. R2-2105345 Slice specific RACH configuration Samsung
21. R2-2105438 Discussion on slice based cell reselection Samsung Electronics Co., Ltd
22. R2-2105475 Slice-specific RACH prioritisation Nokia, Nokia Shanghai Bell
23. R2-2105533 Discussion on slice based cell reselection Spreadtrum Communications
24. R2-2105534 Consideration on slice based RACH configuration Spreadtrum Communications
25. R2-2105568 Consideration on slice-specific cell reselection OPPO
26. R2-2105569 Consideration on slice-specific RACH OPPO
27. R2-2105630 Cell (re)selection for RAN slicing FGI
28. R2-2105631 Cell (re)selection for RAN slicing Asia Pacific Telecom, FGI
29. R2-2105697 Slice based Cell Reselection and intended slice Sony
30. R2-2105738 Considerations on contents of slice related cell selection info KDDI Corporation
31. R2-2105880 Discussion on slice aware cell reselection LG Electronics UK
32. R2-2105942 SMBR enforcement in RAN Ericsson
33. R2-2105943 Cell re-selection enhancements for slicing Ericsson
34. R2-2105944 RACH for RAN slicing enhancement Ericsson
35. R2-2106013 Slice-based cell/frequency prioritization NEC Telecom MODUS Ltd.
36. R2-2106014 RAN Slicing remaining RACH issues NEC Telecom MODUS Ltd.
37. R2-2106087 Consideration on slice-based cell reselection SHARP Corporation
38. R2-2106155 Discussion on SMBR enforcement Huawei, HiSilicon
39. R2-2106156 Discussion on slice based cell reselection under network control Huawei, HiSilicon
40. R2-2106157 Discussion on slice based RACH configuration Huawei, HiSilicon
41. R2-2106175 Discussion on Slice-based Cell Reselection CATT
42. R2-2106184 Analysis on slice based RACH configuration CATT
43. R2-2106223 Discussion on SMBR enforcement in RAN CMCC
44. R2-2106224 Discussion on slice based cell reselection CMCC
45. R2-2106225 Discussion on slice based RACH configuration CMCC
46. R2-2106373 UL SMBR enforcement Samsung
47. R2-2106374 UL SMBR enforcement Samsung
48. R2-2106375 Discussion on slice-specific RACH operation LG electronics
49. R2-2106418 SMBR enforcement in RAN Intel Corporation
50. R2-2106501 Summary of [AT114-e][250][Slicing] Usage of slice priorities for cell reselection (Lenovo) Lenovo

[51] R2-2104321 Summary of [AT113bis-e][251][NR] Slice-specific cell reselection (Intel) Intel

# Annex: Agreements for RACH in previous meetings

*RAN2#113bis-e*

Agreements

1 RAN2 aims to support both RO partition and preambles partition.

2 scalingFactorBI and powerRampingStepHighPriority can be configured at least in SIB (FFS for dedicated RRC signalling).

3 Network can configure slices with 4-step or 2-step (or both) RA resources.

4 Legacy 2-step RA fallback mechanism is supported.

2: RAN2 will prioritize the discussion for slice specific RACH for IDLE and INACTIVE mode. And CONNECTED mode is down prioritized and can be considered if time allows.

3: Slice specific RACH (including RACH isolation and RACH prioritization) is only applied for CBRA but not for CFRA.

4: To ensure the backward compatibility, it is RAN2’s common understanding that common RACH resource should be configured in initial BWP if the slice specific RACH resource is configured in initial BWP.

6: RAN2 confirms that the issue of prioritization parameter collision with MPS/MCS need to be resolved. There is UE based solution (option 1, fixed rule) or network based solution (option 2, configurable rule) or both. Discussion on pros and cons can be left to next meeting.

5.1: RACH type selection between 2-step slice specific RACH and 4-step slice specific RACH is based on a RSRP threshold.

FFS to introduce a slice specific threshold or reuse the legacy threshold.

FFS UE should first select between slice specific RA and common RA or UE should first select RA type between 2-step RA and 4-step RA

5.2: The table from [R2-2104322](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_113bis-e/Docs/R2-2104322.zip) can be used for further discussion.

Slice specific RACH is only applicable if there is slice information (e.g., slice group or slice related operator defined access category) available for AS layer when access. FFS on details of slice group.

*RAN2#114-e*

4: RAN2 confirm for a slice group, separated RO and/or separate preamble can be configured within the existing RACH-ConfigCommon and RACH-ConfigCommonTwoStepRA

5: Same as NR Rel-15 conclusion, RAN2 conclude that there is no RA-RNTI collision between slice specific RACH and legacy RACH in shared RO

6: Same as NR Rel-15 conclusion, RAN2 conclude that the RA-RNTI collision between slice specific RACH and legacy RACH may happen in separate RO.

Working assumption: this can be left to network implementation to resolve it (e.g. network configure RO in different time)

FFS how many slice groups we can have and how they are indicated.