**3GPP TSG-RAN WG2 Meeting #116 e R2-210xxxx**

**E-Meeting, Nov. 1 – 12, 2021**

**Source: Lenovo, Motorola Mobility**

**Title:****[Post115-e][244][Slicing] Resolving FFSs for solution 4 (Lenovo)**

**Document for:** **Discussion and Decision**

# **Introduction**

RAN2 has initiated the following long email discussion.

* [Post115-e][244][Slicing] Resolving FFSs for solution 4 (Lenovo)

Scope: Attempt to resolve solution 4 FFSs, including understanding if there are any impacts to RAN4 requirements. Can draft LS to RAN4 in case any potential impacts are identified.

Intended outcome: report + draft LS to RAN4 (if needed)

Deadline: Long

Following are the relevant agreements from the RAN2#115e:

Agreements

* 1: Solution Option 4 is selected for further work i.e., resolve the FFSs, send any required LSs and consequently start to draft specification CRs
* 2 Following is taken as the baseline for Solution Option 4:

The “slice info” (for a single slice or slice group) agreed to be provided to the UE in the last RAN2 meeting using both broadcast and dedicated signaling are provided for the serving as well as neighboring frequencies. The following steps are used for slice based cell (re)selection in AS:

Step 0: NAS layer at UE provides slice information to AS layer at UE, including slice priorities.

Step 1: AS sorts slices in priority order starting with highest priority slice.

Step 2: Select slices in priority order starting with the highest priority slice.

Step 3: For the selected slice assign priority to frequencies received from network.

Step 4: Starting with the highest priority frequency, perform measurements (same as legacy).

Step 5: If the highest ranked cell is suitable (as defined in 38.304) and supports the selected slice in step 2 then camp on the cell and exit this sequence of operation;

FFS: How the UE determines whether the highest ranked cell supports the selected slice.

Step 6: If there are remaining frequencies then go back to step 4.

Step 7: FFS: If the end of the slice list has not been reached go back to step 2.

Step 8: Perform legacy cell reselection.

Following are the relevant agreements from the RAN2#114e:

|  |
| --- |
| * 1: Frequency priority mapping for each slice (slice -> frequency(ies) -> absolute priority of each of the frequency) is provided to a UE.   Note: Signaling optimizations are not excluded.  Note: "slice may also mean "slice group"   * 1b: Frequency priority mapping for each of the slice (slice -> frequency(ies) -> absolute priority of each of the frequency) is part of the “slice info” agreed to be provided to the UE using both broadcast and dedicated signaling. * 3: RAN2 consider a scenario in its work for slice specific cell (re)selection where it is possible that (Suitable) cells on the same frequency belonging to different TAs support different Slice(s). * 4: Working assumption: The Best cell principle according to absolute priority reselection criteria specified in clause 5.2.4.5 of TS38.304 needs to be met also for slice specific cell (re)selection. |

This email discussion will be carried in 2 phases; currently we are in the second phase:

Phase 1: Understanding company position and shortlisting main options/ issues

Phase 2: Resolving the issues, next-step (e.g., RAN4 LS) and converging

# **Discussion**

## 1st FFS: How the UE determines whether the highest ranked cell supports the selected slice?

This email discussion distinguishes two terms:

1. Slice Info: “Slice info” is defined as frequency priority mapping for each of the slice (slice -> frequency(ies) -> absolute priority of each of the frequency) and therefore consists of these 3 elements (slice, frequency and an absolute frequency priority). The slice info (for a slice or slice group) was agreed to be provided to the UE in the RAN2 meeting#113e using both broadcast and dedicated signaling. Slice info is provided for the serving as well as neighboring frequencies.
2. Slice support: This term is used in this document to signify only the slice(s)/ slice group(s) supported in a particular cell (serving cell or for neighbor cell).

Further, in RAN2#114e meeting, the following was also agreed:

*“RAN2 consider a scenario in its work for slice specific cell (re)selection where it is possible that*

*(Suitable) cells on the same frequency belonging to different TAs support different Slice(s).”*

This practically means that for cell reselections the UE can’t blindly assume that slice support on a frequency is uniform. Therefore, serving cell broadcasting slice support just for neighbouring frequencies may not be sufficient and UE needs to ensure if the highest ranked cell supports the selected slice (i.e., the slice from Step 2). One way to achieve this is to measure the corresponding frequency and check system information of the highest ranked neighbour cell to see if the selected slice is supported.

Further, it is difficult to assume that slice(s)/ slice-group(s) support of a cell can be broadcasted in SIB1 (to not increase SIB1 size too much), the UE may need to acquire other SIB of the neighbour to determine the same. If the highest ranked cell indeed does not support the selected slice, the UE will need to repeat the procedure until the highest ranked cell on some other frequency supports one of the selected slices (down the order). This may/ may not be an optimal solution and some companies may rather prefer that the serving cell provides slice support of neighbour cell. It can be argued that serving cell can’t provide (slice support) information on all neighbour cells; however, network implementation with/ without SON like features can help. If the answer to the following question is “Yes”, it can be assumed that both dedicated and broadcast signalling needs to be supported to provide slice support of neighbour cells.

**Q1: Do companies prefer that a (serving) cell provides slice support of neighbour cells?**

|  |  |  |
| --- | --- | --- |
| Company Name | Yes/ No | Comments (e.g., feasible/ not-feasible since…) |
| Qualcomm | Yes | First, we see no technique issue for serving cell to provide neighbour cell’s slice support info in SIB. Please note that neighbour cells’ cell reselection info have been included in NR SIB3/4/5. We think it is a natural extension for slice availability info of neighbour cell.  Secondly, we agree with rapporteur that it is difficulty to include slice(s)/ slice-group(s) support of a cell in SIB1 because of the payload size limitation of SIB1. Please note that maximum SIB1 size is only 2927bit according to 38.331 and thereby RAN2 carefully evaluated whether to increase even 1 more bit in SIB1 since Rel-16.  Finally, for the issue that serving cell may not provide all neighbour cells’ slice info, we don’t think it is important issue. In this release, the concerned scenario (different cells in same frequency support different slices) can only happen in TA boundary, which is infrequent event. Even when it happens, it can be resolved. Besides SON features mentioned by Rapporteur, we think serving cell’s own slice support info can also be included in its SIB for UE to optionally check. |
| Xiaomi | Yes | We agree with rapporteur and QC that it seems difficult to include slice info in SIB1, thus to avoid acquiring other SIBs of neighbouring cells which can introduce extra access delay, we think it is a straightforward solution to provide neighbour cell’s slice support in SIB of serving cell.  Besides, we agree with QC that support slice info of serving cell can also be included in its own SIB used for slice checking which is met the requirement of WID that the the supported slice info of the current cell and neighbour cells needs to be broadcast. |
| CMCC | Yes | We think it is necessary to provide slice support of neighbor cells to avoid the latency for checking system information of the highest ranked neighbor cell to see if the selected slice is supported.  We also agree that it is difficult to broadcast the slice(s)/ slice-group(s) support of a cell in SIB1 due to the payload size limitation. |
| Nokia | Yes, but see comments | Our view is that the solution can work without checking the slice support before selecting the cell as in most of cases the neighbouring cells on a band support the same slices (we can assume that most of the cells are not at the border of tracking areas). If slice support of neighbouring cells is broadcast the signalling should be optimized (see our comments to Option A below). |
| OPPO | Yes | We agree with the rapporteur’s analysis on the payload size issue of SIB1 carries slice info and the complexity/latency issue of UE performs the camping check for the highest ranked cell on the potential frequency. Thus, we think it is necessary that a (serving) cell provides slice support of neighbour cells.  From a similar perspective, we agree with QC that the serving cell can also indicate its supported slice info, which is helpful for UE to check the supported slice of the serving cell in cell reselection. |
| Spreadtrum | Yes | The serving cell should provide slice support of neighbour cells.  The UE has to know the slice support of the highest ranked cell before it determines whether the cell supports the selected slice.  However, compared with provide slice support in serving cell, listen SI from the highest ranked cell may cause extra delay to cell reselection and not beneficial to UE energy saving. |
| Lenovo, MotM | Yes | Like other companies, we also believe that the serving cell should provide slice support of neighbour cells. However, we definitely need to optimize the signalling since it is unclear how many slice groups may need to be indicated by any cell, how many bits per slice group will be required and so on – but it should be clear that repeating frequency information (22 bits) across many different slices with same PCIs listed must be avoided. E.g., since there will be still 8 possible neighbouring frequencies for a cell, there’s no use repeating the frequency indication again and again for so many slices supported by the cell. Also, generally the slice deployment can be assumed to be reasonable uniform across neighbour cells, rather than listing each PCIs, only exceptional PCIs need to be listed that have different slice support compared with the rest of the cells on a frequency (or compared with the serving cell). |
| KDDI | Yes | If allowed NSSAI will not able to be broadcasted directly on system information, we need to revisit. |
| CATT | Yes | We think the UE obtains the slice information of the neighbour cells by reading the SIB of serving cell without checking the SIBs of neighbour cells is efficient. Besides, we agree that, slice info of the neighbour cells are rather too heavy to be included in SIB1 due to the limitation of payload size. |

Based on the above discussion, and irrespective of your answer to Q1, we can possibly have following solutions:

Option A: Serving cell broadcast slice support of neighbor cells

This option seems extremely signaling heavy e.g., multiple S-NSSAI for many neighboring frequency/ cells may need to be broadcasted. Some signaling optimizations may be used including use of slice groups (instead of individual S-NSSAIs), use of on-demand SIB – if a new SIB is used, etc.

Option B: Serving cell does **not** broadcast slice support of neighbor cells. TAC is used as a proxy for slice support and therefore neighbour cell’s SIB1 needs to be read

A UE needs to read the SIB1 (of the highest ranked cell) to find out the TAC. Based on this, the UE can determine if the selected slice is supported or not. In Rel. 17 this works due to homogeneous slice support principle. But in future releases if the homogeneous slice support principle is changed, this may become a problem.

Option C: Serving cell does **not** broadcast slice support of neighbor cells. Every cell broadcasts its slice support; therefore, a neighbour cell’s slice support needs to be acquired from that cell’s System Information

A cell should anyway indicate which slice(s)/ slice group(s) are supported to help a camped UE (e.g., a UE that selected this cell or a UE that reselected this cell for a certain slice but may need to check if a different slice, as indicated by NAS, is also supported in the serving cell). For reselection purpose, a neighbor cell’s SI needs to be read. However, as discussed earlier, for reselection this means reading of possibly SI of many neighbour cells, consuming time and battery.

Option D: Any other options??

Option D-Nokia: Broadcast neighbouring cells PCI and their associated TACs

This solution is a variant of option A using TACs. In this solution the cell broadcasts the neighbouring cells PCI and associated TAC for the cells that belong to TAC different from the cell. As UEs can assume that all cells that are not listed support the current TAC, and thus have the same slice support as the current cell, the signalling load is not very high. Based on this information UEs know the supported slices by reading MIB, and thus no extra SIB1 checking is needed. Slices corresponding to a TAC can be learnt via NAS or via RRC signalling during registration update (in dedicated RRC message).

**Q2: Which option do you prefer to acquire the slice support of a neighbor cell?**

|  |  |  |
| --- | --- | --- |
| Company Name | Option | Comments (benefits/ shortcoming of option(s)) |
| Qualcomm | Option A with comments | For Option B, we agree with Rapporteur that it can only work under homogeneous slice support assumption. Furthermore, it has chicken-egg issue: the UE can’t derive supported slices info in target cell of different TA only by its TAC until it performs registration in this target cell.  For Option C, we don’t prefer because it requires UE to read target cell’s SIB before camping, which can work only when slice info is included in SIB1. As we commented in Q1, including such info in SIB1 is quite difficulty.  For Option A, we disagree with Rapporteur that signalling optimization is required. The only spec impact is to decide whether include neighbour cell info in a new SIB or SIB3/4/5, which anyway needs a conclusion. Please note whether to have on-demand SIB is up to NW implementation.  In addition, as we mentioned in Q1, we think Option A can be modified to also include serving cell’s own slice support info. For example, below change:  **Option A: Serving cell broadcast slice support of neighbor cells and serving cell** |
| Xiaomi | Option A | We share the same view with QC and think QC’s modification is necessary to make it clear that the serving cell also needs to broadcast its own supported slices. |
| CMCC | Option A | We agree with QC’s suggestion that it’s natural to broadcast its own supported slices for serving cell. |
| Nokia | A, or B or D-Nokia are acceptable | Comment on Option A: If it is decided to advertise slice support information about neighbouring cells, then optimizing the broadcast information is important to make the solution feasible (the SIB size limit is valid for all SIBs). E.g., it could be limited to the bands where the slice support is not uniform in the area (if no cell level information is advertised on a band, then UE can assume that all cells of that band supports the same slices in that area); or no slice group support information is needed when a slice (or slice group) supported in all neighbouring cells.  Comment on Option B: It is a simple solution and the UE can learn the supported slices in the different TAs (e.g. via NAS or RRC signalling during registration procedure). Checking of SIB1 before selecting a cell introduces some additional latency of cell reselection and increases the UE power consumption.  Comment on Option C: Adding slice information to SIB1 is not feasible. Reading additional SIB before selecting the cell will significantly increases the latency of cell reselection and UE power consumption. Therefore, we think that Option C is not acceptable.  Option D-Nokia: This is a variant of option A using TACs that enables good signalling optimization.  Note also that *RRCRelease* can also be used to provide information to the UE. In *RRCRelease* the UE can get more information (the size limit is not so strict) and the information may be tailored to the slices (e.g. configured slices) that the UE may use. |
| OPPO | Option A | For B, we agree with the Rapporteur that it can work only under the homogeneous slice support assumption. In the future, the slice deployment requirement may change. In addition, there should be a way to let the UE know the association between TAC and the supported slice in the TA, which may introduce more inter-WG discussion and normative work. Finally, TAC based solution can not provide a finer granularity checking/comparison for slice support info.  For D-Nokia, the benefit may be reduced if heterogeneous slice deployment is supported. Similar to B, there should be a way to let the UE know the association between TAC and the supported slice in the TA, which may introduce more inter-WG discussion and normative work.  For C, it may require more complexity/latency since currently the UE only needs to read SIB1 for camping check, but now the UE needs to further read SIBs other than SIB1, since SIB1 may not include all required bit sizes for the support slice.  For A, it is a better solution as also mentioned by other companies above. |
| Spreadtrum | Option A | From our perspective, the Option A is preferred to acquire the slice support of a neighbor cell. As for signaling overhead issue, it definitely needs to be optimized which also needs further discussion. |
| Lenovo, MotM | Option A |  |
| KDDI | Option A | But in the case which many candidate cells having the same TAC, it causes many duplicated slice information are provided to UE. And we want to discuss some optimizations to reduce the duplicated slice information. |
| CATT | Option A | We support option A. The concern of option A is that the payload may be very heavy. But we think we can introduce one new SIB rather than using the current SIBs. If we reuse the existing SIBs which are also used by the legacy UEs, it will eventually bring extra resource/power consumption for the legacy UEs. |

**Q3: If Serving cell broadcasts slice support of neighbor cells, which SIB can be used?**

|  |  |  |
| --- | --- | --- |
| Company Name | SIB# | Comments |
| Qualcomm | New SIB | Compared with the solution to include slice info in SIB3/4/5, new SIB can reduce impacts to legacy UE. In addition, the payload size will be large as Rapporteur mentioned. Thus, we don’t prefer to include them in SIB3/4/5. |
| Xiaomi | SIB3/4 | As in the current spec, the cell reselection info of intra-frequency and NR inter-frequency neighbour cell are respectively included in SIB3 and SIB4 of serving cell, thus we think SIB3/4 should be considered firstly to carry the support slice info of neighbour cell, so that UE does not need to acquiring other SIBs to get the slice info.  Considering slice is only supported in NR frequencies, we think there is no need to include the slice info in SIB5 which carries the reselection info of LTE frequencies.  For the payload size concern as QC mentioned, as RAN2 has raised slice group mechanism to resolve this issue, we think it is not a issue to include slice info in SIB3/4. |
| CMCC | SIB ¾ | We think it is simple and natural to extend SIB3/4 to contain slice support of neighbor cells. |
| Nokia | SIB3/4 | Defining a new SIB would create some system level overhead (e.g. there are only a couple of spare values for new SIBs in *SI-SchedulingInfo* information element) that could be avoided with re-using SIB3 and SIB4. Our view is that RAN2 should specify a solution where the additional information could fit in legacy SIBs. |
| OPPO | SIB ¾ | Extend SIB3/4 is sufficient unless there is some critical issue. |
| Spreadtrum | SIB ¾ | Legacy SIB 3/4 are used to provide neighbouring cell information relevant for intra-frequency and inter-frequency cell re-selection. Thus, it is naturally and easily accept to introduce slice support of neighbour cells in SIB 3/4. As for payload size issue, it certainly needs to be optimized.  For SIB 5, it contains information relevant only for inter-RAT cell re-selection. There seems no need to include slice support in SIB 5.  As for new SIB, it is a possible way but has more impact on specification. Thus, it is not preferred. |
| Lenovo, MotM | SIB 3/ 4 | These SIBs will avoid re-listing cells/ frequencies for slice purpose. |
| KDDI | SIB 3/4 | We share the same view with CMCC. |
| CATT | New SIB | As explained in Q2, reusing existing SIBs brings impacts to legacy UEs. And we think slice which is one kind of optimization feature should minimize impacts to legacy UEs. |

**Q4: Which SIB broadcasts slice support of a serving cell?**

|  |  |  |
| --- | --- | --- |
| Company Name | SIB# | Comments |
| Qualcomm | New SIB | We prefer to put the slice support info for neighbour and serving cells in the same new SIB. |
| Xiaomi | SIB2 | Similar to our comments on Q3, for the supported slice info of serving cell, SIB2 which carries cell reselection info of serving cell in normal cell reselection should be considered firstly to avoid extra SIBs acquiring. |
| CMCC | SIB2 | We think it is simple and natural to extend SIB2 to contain slice support of serving cell. |
| Nokia | None | Adding slice information to SIB1 is not feasible. Adding this information to a new SIB requires UEs to read the new SIB before selecting the cell. We think this is not acceptable as this additional SIB reading would significantly increase the latency of cell reselection and UE power consumption. |
| OPPO | SIB2 | Extend SIB2 is sufficient. |
| Spreadtrum | SIB 2 | SIB2 is used to provide cell re-selection information common for intra-frequency, inter-frequency and/or inter-RAT cell re-selection as well as intra-frequency cell re-selection information other than neighbouring cell related.  Thus, SIB2 is suitable to carry slice support of a serving cell. |
| Lenovo, MotM | SIB2 |  |
| KDDI | SIB2 |  |
| CATT | New SIB | Using new SIB can minimize impacts to legacy UEs. |

## 2nd FFS: If the end of the slice list has not been reached go back to step 2?

This FFS intends to keep or remove the step 7 completely.

Argument for keeping Step 7: Even though there’s no guarantee that an application that triggers RRC Connection Est. is really from the highest priority slice an effort can be made esp. since measured frequency needs not be measured again for the “next” slice. Reselecting for highest/ higher priority slice is the main motivation for the work. Further, if the serving cell provides the slice support information for the neighbour cells, it might be more likely that not many iteration of Step 7 may be required.

Arguments for removing Step 7: Too much battery consumption for uncertain gain since it is anyway not clear that an application that triggers RRC Connection Est. is really from the highest priority slice.

**Q5: Does your company want to keep Step 7 (i.e., attempt to reselect on one or more lower than the highest priority available slice)?**

|  |  |  |
| --- | --- | --- |
| Company Name | Yes/ No | Comments |
| Qualcomm | No | 1. UE access latency is main KPI of cell reselection. We think the looping in Step 7 will cause extra latency of cell reselection, especially if number of UE intended slice is large. This is conflicted with the intention to introduce “quick” slice specific cell reselection. 2. We understand slice specific cell reselection is just a best effort enhancement as UE may not have traffic of the highest slice during reselection. 3. Even if we keep Step 7, it can’t resolve the issue raised online that UE may reselect to a cell that only supports 1st priority slice but not 2nd and 3rd priority slices. Assume a scenario with Cell A only support 1st priority slice while Cell B supports both 2nd and 3rd priority slices. According to agreed procedure, the UE will always reselect to Cell A as long as a suitable cell is found, irrespective of whether keeping step 7 or not. 4. We don’t agree with Rapporteur that measurement can be always reused in next iteration. Please note that existing IDLE inter-frequency measurements depends on frequency priority of serving cell and target cell:  * For a frequency with a reselection priority > serving frequency, the UE shall perform measurements for this frequency * For a frequency with a reselection priority <= serving frequency, the UE may choose not to perform measurements if the serving cell fulfils Srxlev > SnonIntraSearchP and Squal > SnonIntraSearchQ   Thus, when the UE changes frequency priority in next slice iteration, the above measurement condition may change, and so measurements in previous round can’t be reused in such cases.   1. This can be regarded as an optimization. Because we have only 3 meetings to finish this WI, we tend to keep it simple by focus on baseline solution and leave optimization to next release. |
| Xiaomi | Yes | It is reasonable for UE to find a cell supporting at least one of its intended slices rather than directly fallback to find a cell may not supporting any slice  Many companies have concerns on long latency and power consumption if it is kept.  A)The latency is not very critical issue for idle/inactive UE.  B)Power consumption is not a big issue as the measured frequency may needs not to be measured again. If companies still have concerns on it, it needs RAN4 to confirm. |
| CMCC | No | We understand that this step is only few benefits in TA boundary scenario where new TA does not support the highest priority slice in this release.   * **If we keep step 7**, the UE may attempt to reselect to a cell which supports the second or the third or lower priority slice. But there may also be possible that after multiple iterations, the UE does not reselect to a cell that supports some slice. This step will increase the power consumption and access latency. * **If we remove step 7**, the UE will perform the legacy cell reselection. * Whether there is step 7 or not, when the UE reselects to a cell in new TA, TAU procedure or registration update procedure will be performed and the UE will receive a new allowed slice list (which may include different slices with new or same priorities from the allowed list of last TA). Then the UE can perform cell reselection procedure based on the new slice priority order and the UE can be able to find a suitable cell to support the new highest priority slice.   Based on the analysis above, we prefer to remove step 7 considering the limited benefits and greater impacts on power consumption and access delay for the UEs.  On the other hand, we think that the measurement results of last iteration cannot be always reused in next iteration. For example, if the UE is moving, then the measurement results may be different and the highest ranked cell may also be changed. |
| Nokia | See comment | We think that it depends on whether re-using the measurements is possible (see Q6).  If new measurements on the same band are needed in all iterations (for all slice groups) then step 7 would create a significant overhead (delay, power consumption) in cell reselection. Thus, we think it is better to remove step 7 in this case.  If no new measurements are needed (measurements from earlier slice group can be re-used), then the iteration would not create significant extra overhead, but can lead better results in considering slices in cell reselection. Thus, we think step 7 can be kept in this case. |
| OPPO | No | Re-using the measurements may not be possible in all cases. If the number of the UE intended slice is large, the access latency and UE power consumption will be largely extended, which may contradict our SI/WI intention of fast accessing. |
| Spreadtrum | No | We agree with QC generally, the multiple iterations may consume pretty much time which violates the intention of fast access. And we already have a loop in step 6 which is enough. UE does best effort to find a cell but excessive optimization should also be avoided |
| Lenovo, MotM | Yes | Even though there’s no guarantee which slice will trigger a RRC connection setup next, the intention should be to attempt to camp on the cell supporting a higher prioritized slice. |
| KDDI | No | We share the same vies with QC #4. |
| CATT | No | We think it is not necessary to go through each slice in the list. This may be time and power consuming. |

**Q6: Since Q5 might affect a UE’s performance, would you prefer asking RAN4 if the measurements of frequencies can be assumed valid for the next iteration(s) and/ or if to keep step 7 or not?**

|  |  |  |
| --- | --- | --- |
| Company Name | Yes/ No | Comments |
| Qualcomm | No | We don’t think RAN2 can well formulate the question to RAN4 at this stage:   * For the question whether measurement is valid for next iteration, we have provided comments in Q5 that it is not always true. * If we request RAN4 to evaluate the impacts on UE’s performance (e.g., latency), RAN2 needs to first decide how many slices or slice groups can be configured. However, it is still not clear. |
| Xiaomi | See the comments | If RAN2 can not conclude on Q5 and one of concerns on keeping step 7 is whether the measurement can be reused, it is natural to ask RAN4 as it is not RAN2 scope. |
| CMCC | No | We think it is too early to ask RAN4 at this stage. |
| Nokia | Yes | Our view on Q5 depends on whether measurements result can be re-used or not. (If RAN2 cannot assume the re-use of measurements in iterations, then "step 7" should be removed.) |
| OPPO | No | It seems too early to ask RAN4 at this stage. |
| Spreadtrum | See comments | We have no strong views. An LS could be sent to RAN4 to check whether our assumption is valid which may help to solve the Q5. |
| Lenovo, MotM | Yes | This is mainly about UE performance Vs UE battery topic – and RAN4 are the expert on this topic. If we do not know for sure, which seems to be the case, we should better use their guidance. |
| KDDI | See the comments | We share the same view with Xiaomi. |
| CATT | No | We agree with QCOM that we have not figured out the how many slice (groups) can be configured. So it is difficult for RAN4 to decide. On the other hand, we can also figure out that not all the slices in the slice list are always to be supported as the highest priority. |

# **Conclusion**

In this email discussion

# **Contact list**

|  |  |  |
| --- | --- | --- |
| Company Name | Delegate Name | Email Address |
| Qualcomm | Peng Cheng | [chengp@qti.qualcomm.com](mailto:chengp@qti.qualcomm.com) |
| Xiaomi | Xiaofei Liu | [liuxiaofei@xiaomi.com](mailto:liuxiaofei@xiaomi.com) |
| CMCC | Ningyu Chen | [chenningyu@chinamobile.com](mailto:chenningyu@chinamobile.com) |
| Nokia | Gyuri Wolfner | [gyorgy.wolfner@nokia.com](mailto:gyorgy.wolfner@nokia.com) |
| OPPO | Zhe Fu | [fuzhe@OPPO.com](mailto:fuzhe@OPPO.com) |
| Spreadtrum | Xiaoyu Chen | [xiaoyu.chen@unisoc.com](mailto:xiaoyu.chen@unisoc.com) |
| Lenovo, MotM | Prateek | [pmallick@lenovo.com](mailto:pmallick@lenovo.com) |
| KDDI | Hiroki Suezaki | hi-suezaki@kddi.com |
| CATT | Chunlin Ni | nichunlin@catt.cn |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |