**3GPP TSG-RAN WG2 Meeting #112-e R2-20xxxxx**

**Online, 2-13 November 2020**

**Agenda Item: 6.1.1**

**Source: MediaTek Inc.**

**Title: Summary of email discussion [AT112-e][045][NR16] Extension of ToAddMod lists (MediaTek)**

**Document for: Discussion, decision**

1 Introduction

This document summarises the following email discussion related to document R2-2009976:

* [AT112-e][045][NR16] Extension of ToAddMod lists (Mediatek)

Scope: Continue discussion on P10, P11, converge to agreements if possible. Review and agree CR.

Intended outcome: Report, agreed CR (if possible)

Deadline: EOM, intermediate deadlines by the Rapporteur.

To allow time for CR preparation, the discussion will take place in two phases.

- Phase 1: Collect comments on the proposals and summarise potential agreements; deadline Tuesday 2020-11-10 1200 UTC

- Phase 2: Draft and agree a CR capturing the conclusions; deadline EOM (Friday 2020-12-13 1100 UTC).

R2-2009982 is taken as an initial proposed CR for discussion.

2 Discussion

## 2.1 Background and examples

The proposals for discussion refer to “examples 1 and 3”, which originate from the proposed text in R2-2009982 for section A.4.3 of 38.331.

Example 1 addresses the case where only the size of a list is extended, i.e. there are no changes to the contained structure (not even to extend an ID field):

-- /example 1/ ASN1START

ContainingStructure ::= SEQUENCE {

listElementToAddModList SEQUENCE (SIZE (1..maxNrofListElements)) OF ListElement OPTIONAL, -- Need N

listElementToReleaseList SEQUENCE (SIZE (1..maxNrofListElements)) OF ListElementId OPTIONAL, -- Need N

...,

[[

-- Non-critical extension lists

listElementToAddModListSizeExt-rN SEQUENCE (SIZE (1..maxNrofListElementsDiff-rN)) OF ListElement OPTIONAL, -- Need N

listElementToReleaseListSizeExt-rN SEQUENCE (SIZE (1..maxNrofListElements-rN)) OF ListElementId OPTIONAL -- Need N

]]

}

-- ASN1STOP

Example 3 addresses the case where the list size is extended and fields are added to the list element structure:

-- /example 3/ ASN1START

ContainingStructure ::= SEQUENCE {

listElementToAddModList SEQUENCE (SIZE (1..maxNrofListElements)) OF ListElement OPTIONAL, -- Need N

listElementToReleaseList SEQUENCE (SIZE (1..maxNrofListElements)) OF ListElementId OPTIONAL, -- Need N

...,

[[

-- Non-critical extension lists

listElementToAddModListSizeExt-rN SEQUENCE (SIZE (1..maxNrofListElementsDiff-rN)) OF ListElement OPTIONAL, -- Need N

listElementToReleaseListSizeExt-rN SEQUENCE (SIZE (1..maxNrofListElements-rN)) OF ListElementId-rN OPTIONAL, -- Need N

-- Parallel lists with maxNrofListElements-rN = maxNrofListElements + maxNrofListElementsDiff

listElementToAddModListExt-rN SEQUENCE (SIZE (1..maxNrofListElements-rN)) OF ListElementExt-rN OPTIONAL, -- Need N

listElementToReleaseListExt-rN SEQUENCE (SIZE (1..maxNrofListElements-rN)) OF ListElementId-rN OPTIONAL -- Need N

]]

}

ListElement ::= SEQUENCE {

elementId ListElementId,

field1 INTEGER (0..3),

field2 ENUMERATED { value1, value2, value3 }

}

ListElementExt-rN ::= SEQUENCE {

-- Field description should indicate that if the elementId-vNxy is present, the elementId (without suffix) is ignored

elementId-vNxy ListElementId-vNxy OPTIONAL, -- Need S

field3-rN BIT STRING (SIZE (8)) OPTIONAL -- Need R

}

ListElementId ::= INTEGER (0..maxNrofListElements-1)

ListElementId-rN ::= INTEGER (0..maxNrofListElements-rN-1)

ListElementId-vNxy ::= INTEGER (maxNrofListElements..maxNrofListElements-rN-1)

-- ASN1STOP

The remaining proposals from the previous email discussion are on the size of the extended ToRelease lists (proposal 10) and the possibility of omitting one of the extended ToRelease lists in example 3 (proposal 11).

## 2.2 ToRelease list size (Proposal 10)

During the discussion, two approaches to dimensioning the listElementToReleaseSizeExt-rN were considered, summarised as “short” and “long”. In the “short” approach, the SizeExt list contains only the “new” entries, i.e. its size is maxNrofListElementsDiff-rN, and in example 3 its entry type is ListElementId-vNxy (so that it can release only the entries that were added to the list by the listElementToAddModListSizeExt-rN). In the “long” approach, the SizeExt list has size maxNrofListElements-rN (as shown in the current examples) and it can release any of the list entries.

Both approaches were suggested in the email discussion; it appears that some companies feel there is a convenience in allowing a single ToRelease list to release any entry of the combined list, while others think there should be a structure for releasing only the extended entries.

**Q1: Should the size-extended ToRelease list use the “short” or “long” structure?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Short/Long** | **Comment** |
| Intel | Small preference for short (see also response to next question) | This should be seen related to how list index ID is used here - whether we use ListElementId-rN (which covers the full ID space) or just the new (difference) ID space. If ListElementId-rN is used, then it can be long to release the whole list. If we want to keep the structure, the ID space should also only cover the difference.  We seem to have used both options in RRC so far (and a possible error – see below). |

**Rapporteur’s summary:** Only one comment was received, with a slight preference for the short structure.

**Proposal 1:** Use the short-list approach to the size-extended ToRelease list as a baseline in the CR for examples 1 and 3 of R2-2009982.

## 2.3 Omission of a ToRelease list in example 3 (Proposal 11)

It was observed in the email discussion that in example 3, one of the ToRelease lists could be seen as redundant. There are a total of three ToRelease lists:

- listElementToReleaseList, the original non-extended list that can release only entries from the original portion of the list;

- listElementToReleaseListSizeExt-rN, the size-extended list that can release any list entry (in case of the “long” structure) or only the extended entries (in case of the “short” structure);

- listElementToReleaseListExt-rN, the full-length list that can release any list entry.

Functionally, either of the second and third lists could be omitted, while still making it possible to release any entry of the list. It could be desirable to eliminate one of the lists to avoid redundancy. There does not seem to be a functional reason to choose one list over the other—it is a question of preferred style.

**Q2: In the “example 3” scenario, should one of the extended ToRelease lists be omitted? If so, which one?**

|  |  |  |  |
| --- | --- | --- | --- |
| **Company** | **Yes/No** | **Which list** | **Comment** |
| Intel | Yes | Small preference to omit 3 (listElementToReleaseListExt-rN) | If size extension is done first (in one release), then we will have 2 (listElementToReleaseListSizeExt-rN) and there doesn’t seem to be a reason to have 3 as well in a later release when a parallel list is created.  If on the other hand the parallel list was created first, then there is no need for 3.  So we can be consistent by using only 2. |

**Rapporteur’s summary:** One comment was received, with a preference for omitting the third list (listElementToReleaseListExt-rN).

**Proposal 2:** Omit the third ToRelease list (listElementToReleaseListExt-rN in example 3 of R2-2009982) as a baseline in the CR.

## 2.4 Additional comments on R2-2009982

Any additional comments on R2-2009982 are invited.

|  |  |
| --- | --- |
| **Company** | **Comment** |
| Intel | 1) It could be useful to have a consistent field description for the release lists. Currently, the ***controlResourceSetToReleaseList*** doesn’t say anything about the ext list, ***spatialRelationInfoToAddModList*** mentions release in the addModList field descrition (“or deleted using *spatialRelationInfoToReleaseListSizeExt”)* and ***pathlossReferenceRS*** has no field description for the releaseList.  2) The recommended relationship (“The field description table should indicate …”) between the original list and sizeExt list seems to be missing in ***pathlossReferenceRSToAddModList***  3) In the end of the field description of ***spatialRelationInfoToAddModList,*** spatialRelationInfoToAddModList2 should also be updated to *spatialRelationInfoToAddModListSizeExt*  4) On example 1 regarding:  listElementToReleaseListSizeExt-rN SEQUENCE (SIZE (1..maxNrofListElements-rN)) OF ListElementId OPTIONAL -- Need N  Shouldn’t this have been ListElementId-vx as a more general case to cover the increase in index ID space? There could be rare cases where the original IE did allow larger ID space but often.  5) In the changes on PUCCH-Config:  spatialRelationInfoToReleaseListSizeExt-r16 SEQUENCE (SIZE (1..maxNrofSpatialRelationInfosDiff-r16)) OF PUCCH-SpatialRelationInfoId OPTIONAL, -- Need N  It looks to me there is an error in the spec – why did we do this without also updating the PUCCH-SpatialRelationInfoId? If this is indeed an error, I don’t know if we can correct it as part of this change, but at least we should not update it to align with the recommendation? |

**Rapporteur’s summary:** A set of suggestions was received in the following directions:

1. Make the existing field descriptions consistent in terms of the relations between the different lists
2. Align the field description of pathlossReferenceRSToAddModList with the recommended practices
3. Correct the field naming in the field description of spatialRelationInfoToAddModList
4. In example 1, change the ToRelease list entries to ListElementId-vNxy to reflect the increase in index space
5. Note the error in the list entry type for spatialRelationInfoToReleaseListSizeExt-r16

Rapporteur agrees with points 1-3. On point 4, it seems that updating the ID type would normally mean that the ListElement structure also needed to be extended with the new ID type, changing the scenario into example 3; this means that example 1 is really only applicable to cases where the ID type is not a simple index into the list (e.g. if there were an extension to the cellsToAddModList in MeasObjectNR, where the ID type is a PCI). On point 5, it seems clear that this was a mistake in Rel-16, but no backward-compatible fix is evident; fortunately the problem does not break the function of the IE, since the spatialRelationInfoToReleaseListExt-r16 can release any entry of the combined lists.

**Proposal 3:** As a baseline in the CR, adopt the following changes:

1. Make the existing field descriptions consistent in terms of the relations between the different lists
2. Align the field description of pathlossReferenceRSToAddModList with the recommended practices
3. Correct the field naming in the field description of spatialRelationInfoToAddModList

3 Conclusion

Since only one set of comments was received, it seems not possible to declare consensus, and the rapporteur’s impression is that further discussion is needed.

Based on the comments and the rapporteur’s understanding, the following proposals are made in respect of a baseline CR for continued discussion:

**Proposal 1:** Use the short-list approach to the size-extended ToRelease list as a baseline in the CR for examples 1 and 3 of R2-2009982.

**Proposal 2:** Omit the third ToRelease list (listElementToReleaseListExt-rN in example 3 of R2-2009982) as a baseline in the CR.

**Proposal 3:** As a baseline in the CR, adopt the following changes:

1. Make the existing field descriptions consistent in terms of the relations between the different lists
2. Align the field description of pathlossReferenceRSToAddModList with the recommended practices
3. Correct the field naming in the field description of spatialRelationInfoToAddModList

This issue is not time-critical to resolve at RAN2#112-e, and thus it seems possible to continue by email.

**Proposal 4:** Discuss and refine the baseline CR in a post-meeting email discussion.