3GPP TSG-RAN WG2 Meeting #129bis R2-2502764

Wuhan, China,7th – 11th April, 2025

**Agenda item: 8.0**

**Source: Nokia (Rapporteur)**

**Title: Report on [POST129][032][ASN.1] ASN.1 Review Process Improvements in Release 19**

**WID/SID: General - Release 19**

**Document for: Discussion and Decision**

# 1 Introduction

This document is the report of the following email discussion:

* [POST129][032][ASN.1] ASN.1 review process (Nokia)

Intended outcome: Proposals on how to improve the process (e.g. splitting the review files)

Deadline: long

An email discussion was held between RAN2#127 and RAN2#129 to discuss the existing end-of-release review process, including the ASN.1 and procedural reviews for the RRC specifications 36.331 and 38.331, and the LPP specification 37.355. The email discussion could be summarized as follows, the report of which can be found in [R2- 2501460](https://www.3gpp.org/ftp/tsg_ran/WG2_RL2/TSGR2_129/Docs/R2-2501460.zip) .

These items were noted as requirements. These do not all pertain to enhancements to the Release 19 process, as some of them capture forward-looking ideas for releases after Release 19 and for 6G. The requirements relevant to Release 19 are highlighted.

1. Use of a common tool available to all 3GPP member companies.

2. Ability to collaborate on the review file without creating conflicts.

3. Ability to provide comments on the review file.

4. Distribution of notifications when the review file is checked out or checked in.

5. Trackability of comments such that they can be referenced in discussion.

6. Merging of corrections into a CR for merging into the frozen specification.

7. The new review process should support all aspects of a specification document.

8. CR generation (semi-automatic or automatic) and change report generation is supported.

Additionally, it is an open question whether companies should be able to provide suggestions for direct, inline, modifications to the review file.

The goal of the present email discussion is to begin to explore ways to improve the end-of-release review process in a way that augments the existing tools, e.g., Microsoft Word and macros, to improve the issues identified in the email discussion described above. Additionally, an offline session was held to discuss enhancements to the format of the specification and implementing CRs. The relevant outcome was to initiate this email discussion to address Release 19 enhancements, the report of which can be found in [R2-2501518](https://www.3gpp.org/ftp/tsg_ran/WG2_RL2/TSGR2_129/Docs/R2-2501518.zip).

# 2 Contact Points

Respondents to the email discussion are kindly asked to fill in the following table.

|  |  |  |
| --- | --- | --- |
| Company | Name | Email Address |
| Nokia (Rapporteur) | Jerediah Fevold | jerediah.fevold@nokia.com |
| Ericsson | Håkan Palm | hakan.l.palm@ericsson.com |
| OPPO | Qianxi Lu | qianxi.lu@oppo.com |
| Huawei | David Lecompte | david.lecompte@huawei.com |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 3 Discussion

Depending on the feedback received during this email discussion, it could be expanded to further phases to hone in on a complete solution that we can all agree on.

## 3.1 Splitting the Review File

It has been suggested that splitting the review files into parts could help to address the performance issues with Microsoft Word when editing large documents. If we are to adopt such an approach, we should decide how the splits should be made and how the splitting would affect the procedure to check-out and check-in the review file, e.g., would there be a separate directory for each split, and does it make the check-out and check-in procedure too onerous given that we would mostly likely be working with more than 10 documents.

**Question 1**: Do you agree that splitting the review files into smaller parts would be feasible and would help to resolve the performance problems with Microsoft Word?

Please use the comments to add any details such as ideas on how to split the file, e.g., one file per major section (5), one file per minor section (5.1), or a custom split that focuses on creating approximately equally sized parts. It could also be discussed how we could structure the directories in the FTP server during the review.

|  |  |  |
| --- | --- | --- |
| Answers to Question 1 | | |
| Company | Yes/No | Technical Arguments |
| Ericsson | Yes | Splitting the review file (e.g. 38331) into a set of smaller sub-files would improve the performance. Assuming a delegate can work on each subfile at a time, this will anyway allow more delegates to work in parallel. Simplest is to split the spec into roughly equally sized subfiles and respect the existing section structure to give somewhat functional content of the smaller files. A split review file means of course more complex task to ensure the spec is consistent.  The FTP structure should reflect this split.  Here is one example of the split (sections with side numbers, based on 38331 caption list) into 13 smaller files (first section of the subfile listed):  Foreword 26  5.3 Connection control 73  5.3 Connection control 73  5.5 Measurements 227  5.8 Sidelink 374  6.3.1 System information blocks 610  – AdditionalPCIIndex 657  – LTE-NeighCellsCRS-AssistInfoList 815  – RACH-ConfigCommon 1000  6.3.3 UE capability information elements 1182  6.3.5 Sidelink information elements 1389  6.3.5 Sidelink information elements 1389  6.4 RRC multiplicity and type constraint values 1494 |
| OPPO | Yes | We also support the splitting, of which the benefit is more than ASN.1 review, considering nowadays, to open/read a 331 spec in WORD version is already time consuming and easy to collapse. |
| Huawei | Yes/No | But this would be not needed for per WI review (see Q4, where we suggest multiple per-WI review threads and one cross-WI review thread). |
| Nokia | Yes | As a starting point if the solution is to split the spec, Ericsson’s suggestion could work well. The sections would be sufficiently small and we don’t have to split inside of any sections. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Summary 1**: TBD.

It has been further suggested that the ASN.1 source could be reviewed separately from the procedural portions of the specifications we review. In this case, the procedure could be to extract the ASN.1 source to a single file and review it all together – the benefit is that changes could be implemented directly and tested against a syntax checker after each change.

**Question 2**: Do you agree that extracting and storing all of the ASN.1 in its own review file(s), per specification, would enhance the functionality of the review procedures?

|  |  |  |
| --- | --- | --- |
| Answers to Question 2 | | |
| Company | Yes/No | Technical Arguments |
| Ericsson | No | See response to Q1. |
| Huawei | See comment | We should keep IE/field descriptions together with ASN.1. |
| Nokia | No |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Summary 2**: TBD.

Two primary methods of reviewing the ASN.1 separately could be discussed.

1. Use Microsoft Word to store the review file for the ASN.1 and review based on existing procedures.
2. Extract the ASN.1 source to an “asn1” file and use Git to propose corrections.

The benefit of the first approach is that it requires fewer changes to the existing review procedure. The benefit of the second approach is to take an opportunity for delegates to become familiar with version control.

**Question 3**: If we decide to review the ASN.1 separately from the procedural part of the specification(s), is it preferable to use Microsoft Word and the existing review procedures or to use a plaintext ASN.1 file stored in Git to capture corrections?

|  |  |  |
| --- | --- | --- |
| Answers to Question 3 | | |
| Company | Word or ASN.1 File | Technical Arguments |
| Ericsson | Word | As you all know, we are (very) supportive of exploring possibilities to use new tools, e.g. Git. But it seems to have limited gain to move NR RRC into Git and review the ASN.1 with one set of tools (Git), while using another set of tools for reviewing the other parts of RRC.  Further, we propose that for 6G, RAN2 should write the field descriptions inline in ASN.1 using ASN.1-comments rather than having the field descriptions in tables like today. The ASN.1 and the associated field descriptions need in most cases be reviewed together for a proper review. The proposal above to have the ASN.1 in Git but the field descriptions in tables in Word seem not to make the ASN.1 review simpler.  Therefore, our view is that R19 ASN.1 review should use MS Word for both ASN1 and procedure text. |
| OPPO | Word | As stated by Ericsson, anyway R19 is not appropriate for Git. |
| Huawei | Word |  |
| Nokia | Word |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Summary 3**: TBD.

## 3.2 Providing Comments

Feedback from the email discussion after RAN2#127 and during the offline session on specification and specification review modernization held during RAN2#129 included that it is useful to be able to provide comments directly corresponding to the text. However, it was also noted that after a large number of comments, it isn’t possible to read everything directly anyway, and the large number of comments can slow down Microsoft Word.

Additionally, comments can only point to one instance of an identified issue, resulting in multiple of the same comment or additional work for the rapporteur to find all relevant instances of an issue.

The following excerpts show the scale of the comments.

**By the end of the Rel-18 review, we had accumulated 1428 comments.**

A close-up of a number

AI-generated content may be incorrect.

**Some pages have too many comments to read all at once. The example shown below isn’t the most extreme.**

Several comments discuss “*VarLTM-Config*”, which appears multiple times in the same page and has thus garnered a few comments. It is impossible to see anything but the selected comment unless a new one is selected, which would collapse the contents of that comment.

A screenshot of a computer

AI-generated content may be incorrect.

**An alternate view is provided for use in draft mode and is shown below.**

A screenshot of a computer program

AI-generated content may be incorrect.

However, when attempting to scroll to read the entire comment, it is highly likely that the document will automatically scroll to the location of the next comment causing a loss in context for the comment.

**Question 4**: Is the existing mechanism for providing comments in the review file sufficient? If not, please provide comments on potential solutions, keeping in mind that some of the performance issues could be reduced already if we agree to split the review file(s).

|  |  |  |
| --- | --- | --- |
| Answers to Question 4 | | |
| Company | Yes/No | Technical Arguments |
| Ericsson | Yes | Indeed, the total number of RILs is a problem to handle in Word. With clear guidance, delegates could be asked to limit the number of RILs and limit the text content in RILs. |
| OPPO | Yes and see comment | Besides the legacy method,  1/ all companies insert comment to the WORD file  2/ later a RIL list is generated based on the script on the WORD file  Is it feasible to consider an approach that  1/ all companies upload the comment using a table manner (Either a table using a separate WORD file, or a table using XLS file), with required fields as we did for RIL submission, and it ends up with a RIL list autonomously.  Hopefully, with the second approach, delegates do not have to worry about 331 spec collapse during RIL uploading step at least.  And in order for easy cross-check, the related WI-code for the uploaded RIL should be filled carefully (if we use XLS file, the filtering operation can help delegate to easily identify the related RILs). |
| Huawei | See comments | Possible improvements:  **Idea 1: only issues numbers in bubbles**  - the bubble comments in the specification file are only issue numbers  - the comments are in a separate file (excel or word)  Gains:  - no collapsing of comments  - can reference the same issue number in multiple places.  - can use revision marks in comments (if word is used)  Drawback: more files  This is similar to OPPO but we still can see all issues on the same piece of text.  For the separate file, if we use MS word, we can still extract it to excel for filtering (with loss in formatting).  **Idea 2: separate review files per WI (and for cross WI)**  - different files for WI-specific issues, one file for cross-WI/general issues  - WI files only cover running CR sections to decrease the file size  Gains:  - most issues are WI-specific, handled in WI-specific sessions  - we avoid merging all the comments to the common file just to extract them to WI-specific excel files and then use them to create WI-specific CRs anyway  - smaller files, easier to edit and review  Drawback: more files  Example for idea 1:  **Specification (NTN specific issues)** 5.2.2.4.21 Actions upon reception of *SIB19* Upon receiving *SIB19* in an NTN cell, the UE in RRC\_CONNECTED shall:  1> start or restart T430 for serving cell with the timer value set to *ntn-UlSyncValidityDuration* for the serving cell from the subframe indicated by *epochTime* for the serving cell;  1> if *SatSwitchWithReSync* and *t-Service* are included, and the UE supports hard satellite switch with resynchronization;  2> if *t-ServiceStart* is included and the UE supports soft satellite switch with resynchronization:  3> perform the satellite switch with resynchronization as specified in 5.7.19 between the time indicated by *t-ServiceStart* and the time indicated by *t-Service* for the serving cell.  2> else:  3> perform the satellite switch with resynchronization as specified in 5.7.19 at the time indicated by *t-Service* for the serving cell.  NOTE: UE should attempt to re-acquire *SIB19* before the end of the duration indicated by *ntn-UlSyncValidityDuration* and *epochTime* by UE implementation.  **Issue descriptions for NTN (ms word file)**:  [RIL]: H001 [Delegate]: Huawei (Lili) [WI]: NTN [Class]: 1 [Status]: ToDo [TDoc]: R2-24xxxxx [Proposed Conclusion]: v122  **[Description]**: UE should switch to new satellite upon t-Service, rather than leaving the NW unaware of the exact switching time  **[Proposed Change]**: If the network does not know when the UE switches to the target satellite, the target satellite does not know when to start scheduling the UE, either too early (wasting resources) or too late (degrading performance).  Before t-Service, the UE can remain connected with source satellite, there is no extra benefit if the UE switches to target satellite earlier (start synchronizing in advance is ok, because it may reduce switching time), so propose to change to  perform the satellite switch with resynchronization as specified in 5.7.19 ~~between the time indicated by~~ *~~t-ServiceStart~~* ~~and~~ at the time indicated by *t-Service*~~for the serving cell~~  **[Comments]**: [Ericsson - Ignacio] We agree that the start and end of the unchanged PCI switch should be better defined. It needs to be further discussed online.  [RIL]: H002 [Delegate]: Huawei (Lili) [WI]: NTN [Class]: 1 [Status]: PropReject [TDoc]: [Proposed Conclusion]: v122  **[Description]**: Redundant description, as there is already a conditional presence for *cg-NTN-RACH-Less-Configuration*  [Proposed Change]:  With the following conditional presence:  cg-NTN-RACH-Less-Configuration-r18 CG-NTN-RACH-Less-Configuration-r18 OPTIONAL -- Cond RACH-lessHO  ***RACHlessHO***  The field is optionally present, Need N, if rach-LessHO is present in reconfigurationWithSync. It is absent otherwise.  The sentence could be simplified to:  2> if rach-LessHO was included in reconfigurationWithSync and cg-NTN-RACH-Less-Configuration was configured:  **[Comments]**: [QC v084] the reason of presence of this part we believe was to make it more clear. Otherwise "cg-NTN-RACH-Less-Configuration was configured" would always be true as it says "was" after handover.  [Ericsson - Ignacio] We agree with Qualcomm's comment |
| Nokia | Comment | We see room for improvement as suggested by Ericsson. But we see some potential in using pointers to comments as suggested by OPPO and Huawei. This question could be the basis of further discussion. If we use a table, there is a potential to use another tool to capture the comments that can be used collaboratively, e.g., a form to submit a RIL which would end up in an online-viewable table or a downloadable XLSX.  An alternative based on Huawei’s proposal could be to simply have the table of comments at the end of the review file and write a simple script to extract the table from the Word file. This could be combined with the split and then no extra files would be generated. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Summary 4**: TBD.

## 3.3 File Management

The current procedure to manage check-out and check-in of the review file follows. A reviewer is often blocked because another reviewer has the file checked out, but it isn’t clear how to resolve that issue. What would possibly be resolved is how to check-out and check-in the review file(s) in a way that notifies other reviewers when the file is available again.

Navigate to Review Directory in FTP Server.

Version of Latest Review File

Version of Latest LOCK File

Equal?

Upload New LOCK file with version + 1

Upload Review File with version +1

Wait and try again later.

**Yes**

**No**

**Question 5**: Without the use of a version control system, please discuss alternatives to the manual uploading of a LOCK file to implement version control in our current end-of-release review procedure.

|  |  |
| --- | --- |
| Answers to Question 5 | |
| Company | Comments |
| Ericsson | For R19 ASN.1 review, we see no real alternative to the existing manual LOCK mechanism. If we split the review file in smaller subfiles (with LOCK per file) the risk that delegates has to wait a (long) time to get the token to upload decreases. It would of course be nice with an SVN or even SharePoint-like tool, but we think this is not realistic for R19. |
| OPPO | We are fine with the current WoW.  On the other hand, if we use the table based approach (See Q4), even if some collision happens, it maybe fine since Rapp can later copy/paste the related RILs into the final version (which is much easier, compared with copy/paste RILs from 331 WORD file..) |
| Huawei | We can use manual lock, but it should not be so difficult to make a tool that:  - shows the list of files  - whenever one clicks on one file, the tool  - does the check, and  - either  - uploads the lock file, downloads the file to edit and renames it with increased version number, or  - says ""locked by this-delegate@the-company since this-time". |
| Nokia | We sympathize with Ericsson and OPPO, but we are also open to developing a simple tool to handle the creation of the lock file as suggested by Huawei |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Summary 5**: TBD.

## 3.4 Additional comments (Ericsson)

We provided some additional comments/proposals, and a table for additional company comments.

1. Make the “RRC style” (e.g. naming conventions) more visible to WI editors and delegates.
   1. The RRC Rapp could give some presentation and provide slide mtrl.
2. Do not implement WI-CRs that are not functionally complete and not in good shape into the review file.
   1. The ASN. Review originally assumed high-quality WI CRs (i.e., formally respecting the RRC style and guidance, as well as functionally complete and stable.
   2. This is indeed controversial topic. Typically, all WI are considered complete at the time of creating the first spec version of the release, however this approach may make ASN.1 review more effective and smoother.
   3. Some WI CR may benefit from a delayed merge, allowing more time to resolve functional and ASN.1 issues outside the ASN.1 review.
3. The time for ASN.1 review is normally very short. To get more time, we could start the ASN.1 review on **draft** spec before official spec is available. We can discuss if this is feasible closer to the ASN.1 review.
4. To improve ASN.1 review by delegates, encourage the delegates to study and the latest WI CRs to more easily understand spec impact. In this way a reviewer who wants to review how feature X was implemented, can more easily understand feature X if it has reviewed the WI CR for feature X.
5. On adding and commenting RILs, advice delegates on the following:
   1. Avoid a lot of RILs overlapping same word/sentence/paragraph/section.
   2. Avoid too large text comments and text in a RIL
   3. Discussion among companies about a certain RIL should be outside review file (instead use e.g. email among most concerned/involved delegates). In principle only the problem and conclusion need to be captured in the RIL text itself.
6. Allow delegates to introduce changes directly in spec text of the review file
   1. Many minor changes can be corrected directly in the review file, simple typos can be corrected even without a RIL which reduces workload on the delegates
   2. This would possibly also reduce the need for additional tdocs, instead fix the problem inside the review file. So far, we have not allowed changes to the spec text in the review file
7. Can consider a Tdoc cap related to RILs, e.g. one Tdoc for RILs per company per WI.

|  |  |
| --- | --- |
| Company | Comments |
| OPPO | Steo-6 is a bit risky, considering sometime there is no clear boundary and some issues relies on the judgement / perspective of the RIL initiator. |
| Huawei | For the October 2025 meeting, it is unrealistic to have the merged specification available for enough time for ASN.1 review, so we should use agreed CRs and do per WI review as early as possible (see our idea 2 for Q4). Once the merged specification is ready, we can start the cross-WI review.  Problem 2 from Ericsson seems solved in this case.  Problem 5 from Ericsson can be solved by our idea 1 in Q4.  Problem 6 from Ericsson: at most only for typos, nothing else (many people proposed controversial things as "editorial"). |
| Nokia | 1 – This seems reasonable since education is always a good thing.  2 – This should probably be discussed to understand the implications, but if CRs which are not ready for implementation are held until they are, then we could avoid providing corrections for those, increasing efficiency.  3 – Let’s open this for discussion with the broader group.  4 – We can always encourage. Maybe this will become easier with the split or with WI-specific review files.  5 – We agree that avoiding overlapping RILs is desirable, and perhaps small changes to existing RILs can be discussed with the original author and updated – with attribution indicating updates were made based on other company inputs. Reducing the size of comments would also be useful. One suggestion could be to avoid reproducing similar text in the proposed change and description and rely more on the highlighted text for the comment to provide context.  6 – We think this could be quite difficult since changes on top of changes could easily occur, making the current state a bit confusing. We should try to understand what minor changes mean, and then discuss further.  7 – We should first discuss if the quantity of Tdocs for the RILs is a point of inefficiency seen by other companies. |
|  |  |
|  |  |
|  |  |
|  |  |

# 4 Conclusion

TBD.