3GPP TSG-RAN WG2 Meeting #129 R2-250xxxx

Athens, Greece, 17 – 21 February 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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**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 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Summary 5**: TBD.

# 4 Conclusion

TBD.