**3GPP TSG RAN WG2 Meeting #116-e R2-21xxxxx**

**Electronic meeting, 1st – 12th Nov, 2021**

**Source: Huawei, HiSilicon**

**Title: [Post115-e][607][POS] Integrity assistance data (Huawei)**

**Agenda item: 8.11.5**

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

# Introduction

The following email discussion has been planned during RAN2#115 for the issues with integrity assistance data.

* [Post115-e][607][POS] Integrity assistance data (Huawei)

Scope: Discuss the supported assistance data for UE-based integrity determination, considering at least the following candidates that were proposed to RAN2#115-e:

* Quality indicators (standard deviation or variance) of the GNSS error sources
* Mean values of the GNSS error sources
* Information describing the time variation of the GNSS error sources
* Probability of satellite fault
* Probability of constellation fault
* “Do Not Use” assistance data alerts
* “Do Not Use” SV and/or GNSS constellation alerts

Assistance data can be considered in relation to the following categories of feared events from the TR:

* Feared events in the GNSS Assistance Data (category 1)
* GNSS feared events (category 3)
* LMF feared events (category 5)

Intended outcome: Report to next meeting

Deadline: Long

This questionnaire intends to handle the issues defined within the scope of the email discussion.

# Contact Information

|  |  |
| --- | --- |
| Company | Contact: Name (E-mail) |
| Swift Navigation | Grant Hausler (grant@swiftnav.com) |
| Huawei, HiSilicon | Yinghao Guo (yinghaoguo@huawei.com) |
| ZTE corporation | Yu Pan(pan.yu24@zte.com.cn) |
| Nokia | Ping-Heng Wallace Kuo (Ping-Heng.Kuo@nokia.com) |
| CATT | Jianxiang Li (lijianxiang@datangmobile.cn) |
| vivo | Annie Zhong(tingting.zhong@vivo.com) |
| OPPO | Liu yang (liuyangbj@oppo.com) |
| Xiaomi | lixiaolong (lixiaolong1@xiaomi.com) |
| InterDigital | Jaya Rao (jaya.rao@interdigital.com), Fumihiro Hasegawa (fumihiro.hasegawa@interdigital.com) |
| Fraunhofer | Birendra Ghimire ([birendra.ghimire@iis.fraunhofer.de](mailto:birendra.ghimire@iis.fraunhofer.de)) |
| ESA | Florin Grec (florin-catalin.grec@esa.int) |
| Ericsson | [Ritesh.shreevastav@ericsson.com](mailto:Ritesh.shreevastav@ericsson.com), [Fredrik.gunnarsson@ericsson.com](mailto:Fredrik.gunnarsson@ericsson.com) |
| Qualcomm | sfischer@qti.qualcomm.com |
| Samsung | June hwang (june77.hwang@samsung.com) |
| u-blox AG | David Bartlett (david.bartlett@u-blox.com) |
| Mitsubishi Electric Corporation | Akinori Taira (taira.akinori@cs.mitsubishielectric.co.jp) |

# Discussion Phase I

In the phase I of the email discussion, we will handle the following issues regarding the integrity assistance data:

* Error bounding techniques
* State space representation vs Observation space representation
* Identify the set of possible integrity assistance data for the 3 categories of feared events
* Identify the set of possible integrity assistance data for GNSS integrity service

Error bounding techniques

[6] has discussed on limitations with the zero-mean assumption for bounding the error distribution and proposed an alternative method of “paired overbounding”, which consists of bounding the true distribution by two non-zero mean Gaussians, shifted by and . In [3], it has also been mentioned that the error bound should at least include the variance/standard deviation of error sources. In [4], deviation has been proposed for bounding the error sources for satellite orbit, clock, bias, ionosphere, and troposphere.

The main difference between [6] and [3][4] is that [6] propose to add mean values of the error in the assistance data to further bound the error. Thus, companies are invited to answer the following question:

###### Question1-1: Do companies agree that we should adopt the “paired overbounding” technique for bounding the probability distribution of the errors for GNSS integrity?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Swift Navigation | Yes | Paired overbounding enables tighter bounding of the error distributions (to better reflect their real-world error properties), leading to smaller PLs. The paired overbounding method in [5] is also compatible with the proposals in [3] and [4] given the mean (µ) can be set to zero if the provider only chooses to report the sigma (σ). Hence, paired overbounding leads to greater flexibility in the choice of implementation by sending both µ and σ in the assistance data. |
| ZTE | Partially yes | One concern is that how to determine the μ value, another concern is that the μ value may have an impact on the original variance/standard deviation. That is to say μ and σ should be determined as a combination. If this can be well solved(although it may be out of RAN2’s scope), then it is beneficial to adopt the paired overbounding to better describe the error model. Moreover, transmitting the μ value additionally will not introduce larger spec impact in RAN2. |
| Nokia | Yes | It seems to be adopted in the GNSS ecosystem already, so it should be compatible for 3GPP to also take this into account. |
| CATT | Yes |  |
| vivo | Yes | We think paired overbounding is suitable because it is not only compatible with other methods but also achieves tighter bounding which reflects real-world error properties. |
| OPPO | Yes |  |
| InterDigital | Yes |  |
| ESA | Yes (partly) | We agree with Swift: [6] is building on top on a much simpler concept proposed in [3] and [4] while still keeping the simpler concept on the table. We disagree with where [5] suggest to put new fields such as mean and sigma values: new Ies as per [5] or extension of SSR Ies as per [3] and [4]. Generally speaking, we would like to keep the changes to LPP to a minimum in particular because new Ies translate to new posSIBs as well and there are already some problems with scheduling of posSIBs discussed in other email discussions. Regardless of this additional point we have in mind, we agree that paired overbouding is more complete solution. |
| Ericsson | Yes |  |
| Qualcomm | Partially No | We agree that both,  and  of the error sources should be provided in the assistance data. However, there seems no need to "adopt a paired overbounding technique" in the specifications. It was agreed that the "specific algorithms used for positioning integrity shall be up to implementation". We assume this applies to both, UE and LMF. In any case, a precise definition of "paired overbounding" technique needs to be provided before this question can be answered. |
| Samsung | Yes | We also see that real world error distribution can be reflected by overbounding with mean value addition. This is also not underestimating the error occurrence. Even this make some conservative estimate on the location but still good for the safety. |
| u-blox | Yes | An overbounding technique is essential because many errors are not Gaussian and can have longer tails. Using a single Gaussian assumption is not adequate given the wide range of TIRs identified and the fact that most lie far into the tails of the error distribution. |
| Intel | Yes |  |

###### Question1-1 Summary:

All the companies that have replied think that we can adopt the paired overbounding technique for bounding the error probability distribution for GNSS integrity, with the following understanding:

* Nokia thinks that the technique of paired overbounding is already adopted in the GNSS ecosystem.
* ZTE showed some concerns how to derive the mean and its impacts on the original variance/standard deviation.
* QC think the mean the variance/standard deviation of the error source are needed in the AD but not sure why “a paired overbounding technique” is needed and argue that we have agreed that the specific algorithm should be up to UE implementation
* SS thinks the paired overbounding technique provide good approximation to the error distribution in the real world.

Hence, we propose the following:

***Proposal1-1*: The paired overbounding technique is supported for bounding the error probability distribution for GNSS integrity as a baseline.**

## Representation of error sources

Positioning errors can be represented in the following different ways as introduced in different releases in 3GPP

* Observation state representation (OSR) in R15, e.g., RTK
* State space representation (SSR) in R16, e.g., PPP and PPP-RTK.

The main difference between OSR and SSR is that SSR has independent characterization of each error source while OSR aggregates all the error sources.

In [6], it has been pointed out that SSR has the following advantages:

|  |
| --- |
| * Alignment with the SSR representation of errors. * SSR representation can be converted to OSR but not vice versa, therefore SSR is more general. * Better integrity KPIs, from tighter bounding of the error distribution (see ‘error overbounding’ above). * OSR representation is specific to a single user location whereas SSR representation can cover any number of users within a geographic area, leading to much enhanced scalability. * As OSR uses an aggregate of all errors, a single feared event will make integrity unavailable. With SSR individual components are sent separately so if one component is impacted by a feared event, the other components may still be used to allow for graceful degradation. |

During RAN2#115e, the following agreement has been achieved for the supported GNSS methods for GNSS integrity:

Agreements:

Proposal 1: Agree that the GNSS feared events will be addressed in the WI.

Proposal 2 (modified): Agree that all for A-GNSS positioning methods, positioning integrity determination is supported in LPP.

Proposal 3: Agree that additional IEs are needed in LPP to support A-GNSS positioning integrity determination.

Proposal 4: The specific algorithms used for positioning integrity shall be up to implementation.

Proposal 5: For interoperability, the use of “hard-coded” parameters should be minimized and instead the needed parameters should be sent explicitly in the assistance data.

Proposal 6: RAN2 agrees that the PL will be reported in the Integrity Results. It is FFS whether Mode 2 and the TIR, AL, TTA that were used in the integrity calculation will also be reported in the integrity results.

Proposal 8: Agree that the UE feared events will be handled in the implementation for UE-based (network-assisted) methods of positioning integrity determination.

Proposal 10: Agree that the LMF feared events can be handled via implementation for the UE-based (network-assisted) and UE-assisted (LMF-based) methods of positioning integrity determination.

Proposal 11: RAN2 agrees to use Common Positioning IEs to transfer the KPIs and Integrity Results.

Proposal 12: RAN2 agrees that the LPP procedures can be used to transfer the KPIs and Integrity Results. For UE-assisted, the LCS procedures remain FFS in the case of MO-LR.

While in the summary for GNSS integrity in RAN2#115 [2], it has also been argued by the feature lead that we should not do prioritization between SSR and OSR since the objective defined in the WID is to support GNSS integrity for all the GNSS positioning methods up to now, a.k.a. all the GNSS positioning methods since R9.

While based on the understanding from the rapporteur, the intention from [6] is not to deprioritize OSR and prioritize SSR. It is argued in [6] that the assistance data for SSR is a superset of OSR, that if we support the error representation with SSR for GNSS integrity, the GNSS integrity for OSR is naturally supported.

Based on the above, companies are invited to answer the following question regarding the error representation of OSR and SSR for GNSS integrity.

###### Question1-2: What additional assistance data are needed for the feared event whose error source is represented by OSR and please provide the corresponding TP (including the field by ASN.1 and field description)?

|  |  |
| --- | --- |
| **Company** | **Comments** |
| Swift Navigation | None – SSR representation of integrity supports all the positioning methods including RTK/OSR.  Note however that the converse is not true, OSR representation would not satisfy the PPP-RTK and PPP positioning methods (i.e. SSR methods). |
| Huawei, HiSilicon | Agree with Swift that OSR representation can be constructed by the SSR but the reverse is not possible. So SSR is enough |
| ZTE | Agree with Swift and Huawei that the indication of error source(statistical model) in SSR representation is enough |
| Nokia | SSR seems better suited to large scale industrial applications as it requires only unidirectional transmissions resources and consumes less transmission resources. Also, as commented by other companies, it can already cover OSR. So we should only consider SSR. If any need of OSR is identified in the future, we can come back to this. |
| CATT | Agree with Swift. SSR sends the individual error components separately, while OSR send the aggregate of all errors. SSR representation can support both SSR and OSR methods. |
| vivo | No need. SSR is enough which covers OSR. |
| OPPO | Agree with Swift Navigation OSR could be covered by SSR representation of integrity. |
| Xiaomi | Agree with Swift that SSR is enough. |
| InterDigital | Same understanding with Swift and other companies that SSR is enough |
| Fraunhofer | Same understanding as all above. |
| ESA | Agree with the majority. |
| Ericsson | We also agree with the majority to start from SSR, but agree with QC (below) that additions for OSR integrity AD are not precluded in case the existing RTK residuals are considered insufficient. |
| Qualcomm | Integrity assistance data for OSR should be aligned with RTCM and may comprise:  - General parameters, which are common to any NRTK implementation; e.g.,  - integrity of cycle slips detection  - Differenced parameters, which are specific for differenced approaches; e.g.  - double difference covariance network matrix elements  - baseline dependent integrity parameters  - carrier phase double difference residuals  - Undifferenced parameters, which are specific for undifferenced approaches and may be similar to SSR approach (tropo, iono, SIS integrity)  Details should await a response LS from RTCM. |
| Samsung | We also have the same view with the majority that SSR representation can autonomously handle the OSR representation, so no need further information in AD. |
| u-blox | Additional assistance data is not essential for OSR but the use of SSR representation of integrity may be complex and not optimum for OSR. We should not preclude adding OSR specific integrity parameters in the future. |
| Intel | Agree with Swift that the Error sources represented by SSR can cover OSR, and no new information is needed. |

###### Question1-2 Summary:

Based on the feedback, all the companies think that the assistance data with SSR is enough, with the following understanding:

* Error representation by OSR can be constructed by SSR but the reverse is not possible
* QC thinks that the assistance for OSR should also be added and aligned with the RTCM spec, which may include general parameters, differentiated parameters and un-differentiated parameters
* U-blox also mentioned that the SSR representation is complex and not optimum for OSR, also it can be constructed by SSR.

Based on the above feedback, the rapporteur thinks that we can agree that error representation by SSR can be supported for GNSS integrity. For representation with OSR, companies can propose TP to the further meetings and it can be decided later if AD for OSR also needs to be added.

Hence, we propose the following:

***Proposal1-2*: Error representation by SSR is supported for GNSS integrity. FFS alignment with the assistance data for OSR in RTCM.**

In [7], it has been mentioned that in terms of SSR, the current spec already has basic support for GNSS integrity with the URA, specifically

|  |
| --- |
| * SSR Orbit corrections are provided as radial, along track and cross track corrections, with an optional velocity component in each of these dimensions. Corresponding uncertainty extensions per field is a natural extension * SSR clock corrections are provided as a parameterized polynomial where an uncertainty per parameter is natural, but also an uncertainty representing the correction as a whole * SSR code bias and phase bias are naturally extended with an uncertainty per field * SSR atmospheric delay models already come with quality indicators for the STEC and gridded corrections |

In [3], it has also been mentioned that current LPP already has basic system-level and user-level support for GNSS integrity

|  |
| --- |
| ***3GPP: Is there already a solution in LPP?***  Yes, the *GNSS-RealTimeIntegrity* IE. This is the most basic form of integrity capability.  ***3GPP: Is there already a solution in LPP?***  Yes, the *navURA* fields and *GNSS-SSR-URA* IE. |

Furthermore, in the TR [1] during the study item phase, we have defined integrity KPI for the key use cases of automotive, tail, and IIoT as follows:

Table 9.2.4: KPI examples for the Automotive, Rail and IIoT use cases [34][35][36][37].

NOTE: KPIs are defined by the service provider implementation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AUTOMOTIVE EXAMPLES** | | | | |
| **APPLICATION CATEGORIES** | **TIR** | **AL** | **TTA** | **Integrity Availability** |
| **Safety-Critical Applications**  - Warnings (red light, obstacle, queue, curve speed, blind spot lane change, pedestrians etc)  - Automated Driving (lane-level or better)  - Emergency Brake Assist  - Forward Collision Avoidance | Typical range:  ≥10-8/hr to ≤10-6/hr | Typical range: ≥1.5m to <5m | Typically ranges from 100s of milliseconds to <10 seconds | Typically ranges from 95% to 99.9% or greater |
| **Payment Critical Applications**  - Road User Charging (RUC)  - Pay Per Use Insurance  - Taxi Meter  - Parking Fee Calculation | Typical range:  ≥10-6/hr to ≤10-4/hr | Typical range: ≥1.5m to <25m | Typically ranges from 95% to 99.9% or greater |
| **Smart Mobility**  - Freight and Fleet Management  - Cargo/Asset Management  - Vehicle Access/Clearance  - Emergency Vehicle Priority  - Speed Limit Information  - In-Vehicle Signage  - Reduce Speed Warning  - Dynamic Ride Sharing |
| **RAIL EXAMPLES** | | | | |
| **APPLICATION CATEGORIES** | **TIR** | **AL** | **TTA** | **Integrity Availability** |
| **Safety-Critical Applications**  - Absolute Positioning  - Train Awakening  - Cold Movement Detector  - Track Identification  - Level Crossing Protection  - Train Integrity and Train Length Monitoring | Typical range:  ≥10-9/hr to ≤10-8/hr | Typical range: ≥2.5m to <25m | Typically  <7s | Typically ranges from 95% to 99.9% or greater |
| **Liability-Critical Applications**  - Trackside Personal Protection  - Management of Emergencies  - Train Warning Systems  - Infrastructure Charging  - Hazardous Cargo Monitoring  - On-Board Train Monitoring and Recording Unit  - Traffic Management Systems | TBD | Typical range: ≥25m to <62.5m | Typically ranges from seconds to <30s | Typically ranges from 95% to 99.9% or greater |
| **IIOT EXAMPLES** | | | | |
| **APPLICATION CATEGORIES** | **TIR** | **AL** | **TTA** | **Integrity Availability** |
| **AGV Applications**  - Mobile device tracking  - Asset tracking  - Process automation  - Inbound logistics | Typical range:  ≥10-8/hr to ≤10-1/hr | Typical range:  ≥0.5m to <30m (vertical/horizontal) | Typically ranges from 100s of milliseconds to <10 seconds | Typically ranges from 95% to 99.9% or greater |

With the above, companies are invited to answer the following question to help to give a justification for the what we are doing for GNSS integrity and why this enhancement is needed for the use cases in R17.

###### Question1-3: Which use case do companies think that the current support of GNSS integrity in R16 LPP is already sufficient for?

|  |  |  |
| --- | --- | --- |
| **Company** | **Use Case** | **Comments** |
| Swift Navigation | None | We do not believe any of the use cases in the table will be satisfied using the existing IEs alone. The existing IEs are not useful in assisting the UE in computing a Protection Level as they are incomplete and no statement is made in the R16 specification as to the statistical meaning of these indicators (e.g. residual risk). |
| Huawei, HiSilicon | None | The current support of integrity is quite primitive that it is not able to support the above cases of rail, automotive, and IIoT examples. |
| ZTE | None |  |
| Nokia |  | In order to support these use cases, Rel-17 LPP should at least provide mechanisms to convey integrity requirements (i.e. KPIs) and integrity results - the basic form of which is already agreed in the previous meeting so we are on track.  For the assistance data, however, we think the existing IEs can already be used to a certain extent, although they may not be perfect in some sense. Even if it is not ideal, we think it is more important to first observe what will be introduced by RTCM, before jumping to conclusions of adding new assistance data in 3GPP. |
| CATT | None | R16 LPP cannot sufficiently support the use cases listed above and more enhancements of integrity are needed. |
| vivo | None | The current support of GNSS integrity is not sufficient. |
| OPPO | None |  |
| Xiaomi | None | The R16 LPP needs to enhance to support the above use cases. |
| InterDigital | None | Enhancements are needed in IEs to derive integrity metrics to support the above use cases |
| Fraunhofer | None |  |
| ESA | None | The existing Ies are sufficient to inform UE if there is a problem with a number of GNSS satellites and/or signals while navURA is not fit for real-time operations of use cases we discuss.  At the same time, we feel the need to emphasise that by just adding few extra Ies as suggested in [5] will not be enough to satisfy the very stringent TIR levels (10^-6/h … 10^-9/h) from above. There are many reasons for this including but not limited to no progress on UE, LMF, communication channel feared events.  Based on all what we experienced until now, we are of the opinion that the work resumes to specifying enablers for GNSS positioning integrity without actually considering the use cases (and their needs) from above. To conclude, we agree with Nokia, it is best to wait for RTCM because the work there involves design of solution based on needs and testing in the field. |
| Ericsson | None |  |
| Qualcomm | Most of them… | …where code-phase based Differential-GNSS accuracies are sufficient (e.g., ~5m). Differential GNSS (as supported since Rel-9) also provides an integrity monitoring function that detects or ameliorates large satellite signal errors. This, together with RAIM, provides integrity sufficient for most applications.  However, high integrity integrated with high precision (HA-GNSS), additional assistance data are beneficial, e.g., as currently under development in RTCM. |
| Samsung | None | As I understand we RAN2 is specifying on this new concept in LPP domain in R17 WI. So, the definition of positioning integrity can only be available for the above cases only after R17 POS is specified. |
| u-blox | Few | It is important to remember that the TIR requirements listed in the above table are for the application. In a typical implementation the Positioning Function will accept inputs from several different navigation sources. These could include GNSS, IMU, Lidar, cameras, radar, map matching and of course RAT-dependent NR positioning sources. The Positioning Function combines inputs from multiple sources to arrive at a final navigation output. GNSS is one component of a high integrity positioning solution. Therefore it is necessary that each of the sub-systems contributing to the Positioning Function provides sufficient information about the quality of its output to allow meaningful decisions to be made. The GNSS integrity IEs in R16 do not provide sufficient information to allow different sources of position information to be combined optimally. |
| Intel | None | At least KPI and integrity results should be added. For assistance data, the question is whether we need to wait for RTCM or not considering they will only complete their work around the middle of next year. |

###### Question1-3 Summary:

General understanding is that the current (up until R16) support for GNSS integrity is in-efficient for the use cases listed in the TR for the key use cases for R17 GNSS integrity.

* Nokia mentioned and ESA agreed that they think the existing support can be used to a certain extent, but prefer to wait for the progress in RTCM before jumping to conclusions in 3GPP. The rapporteur would like make the observation that we have already sent an LS to RTCM, while still have not received the reply. We will come back to this issue with another question in the second phase of the discussion.
* QC thinks that most of the use case can already be satisfied for with the existing support for GNSS integrity up to release 16. But agree that for HA-GNSS, additional AD might be beneficial.
* Ublox thinks that there are few use cases can be achieved with the current support for GNSS integrity. The main issue with R16 support is that it does not provide sufficient information to allow different sources of position information to be combined optimally

But for the summary of this question now, the rapporteur would like to observe that there is no specification impacts for this if we think the current support for GNSS integrity is not enough; but there will be fundamental question on whether it is worthwhile to have R17 GNSS integrity at all if the current spec can already support these use cases. Nevertheless, based on the feedbacks, companies’ general thinking is still that the current support for GNSS integrity is still in-sufficient.

Hence, we propose the following:

***Proposal1-3*: The support for GNSS integrity in R16 is in-efficient for the use cases defined for GNSS integrity in TR 38.857 for R17**

Assistance data for different feared events

During the study item of the R17 positioning, the following categories for feared events have been defined in the TR [1], illustrated by the following Figure 1:



**Figure 1: Categorization of Integrity Feared Events**

During the RAN2 discussion in R2#115, the following agreements have been made in the feared events. Thus, we have agreed that (2) can be excluded with the following agreement.

cid:image002.jpg@01D79924.4046C090

We have almost agreed on the reprioritization of LMF-based integrity calculation and with the scope of the email discussion, the part (4) can also be excluded

In addition, for the LMF-feared events for UE-based integrity, the following has been agreed, and thus we don't need to discuss on category (5):

Proposal 10: Agree that the LMF feared events can be handled via implementation for the UE-based (network-assisted) and UE-assisted (LMF-based) methods of positioning integrity determination.

Then, the assistance data for feared events in the categories (1) and (3) are open for discussion. In the following, we further discuss what assistance data are needed for the remaining 2 types of feared events.

For the text proposals submitted to the last meeting and the discussion papers, the following parameters have been provided mainly under the following contributions:

|  |  |
| --- | --- |
| Swift Nav et al [5] | The assistance data for feared events are highlighted as follows  (a)Integrity Bounds (b) Residual Risks (c) Correlation Times (d) Alerts (e) Validity Times (f) Service Parameters   * *GNSS-CommonAssistData*   1. gnss-Integrity-ServiceParameters      + irMinimum      + irMaximum   2. gnss-Integrity-ServiceAlert      + serviceDoNotUse      + ionosphereDoNotUse      + troposphereDoNotUse   3. gnss-Integrity-TroposphereParameters      + epochTime      + iod-ssr      + validityPeriod      + pTroposphereFault      + tTroposphereFault      + tCorrelationTroposphere      + tCorrelationTroposphereRate   4. gnss-Integrity-TroposphereErrorBounds      + epochTime      + iod-ssr      + correctionPointSetID      + validityPeriod      + gridList SEQUENCE (SIZE(1..64)) OF Integrity-TroposphereGridElement        - meanTroposphereVerticalHydroStaticDelay        - stdDevTroposphereVerticalHydroStaticDelay        - meanTroposphereVerticalWetDelay        - stdDevTroposphereVerticalWetDelay        - meanTroposphereVerticalHydroStaticDelayRate        - stdDevTroposphereVerticalHydroStaticDelayRate        - meanTroposphereVerticalWetDelayRate        - stdDevTroposphereVerticalWetDelayRate * GNSS-PeriodicAssistData   1. gnss-Integrity-PeriodicServiceAlert   2. gnss-Integrity-PeriodicTroposphereErrorBounds   3. gnss-Integrity-PeriodicConstellationAlert   4. gnss-Integrity-PeriodicConstellationParameters   5. gnss-Integrity-PeriodicBiasErrorBounds   6. gnss-Integrity-PeriodicOrbitClockErrorBounds   7. gnss-Integrity-PeriodicIonosphereParameters   8. gnss-Integrity-PeriodicIonosphereErrorBounds * GNSS-GenericData   + - gnss-Integrity-ConstellationAlert     - constellationDoNotUse     - integrity-svAlertList SEQUENCE (SIZE(1..64)) OF Integrity-SVAlertElement       * svID       * svDoNotUse   1. gnss-Integrity-ConstellationParameters      + epochTime      + iod-ssr      + validityPeriod      + pConstellationFault      + tConstellationFault      + pSatelliteFault      + tSatelliteFault      + tCorrelationRangeOrbit      + tCorrelationRangeClock      + tCorrelationRangeRateOrbit      + tCorrelationRangeRateClock   2. gnss-Integrity-BiasErrorBounds      + epochTime      + iod-ssr      + validityPeriod      + integrity-biasErrorBoundsList SEQUENCE (SIZE(1..64)) OF Integrity-BiasErrorBoundsElement        - svID        - meanCodeBias        - stdDevCodeBias        - meanCodeBiasRate        - stdDevCodeBiasRate        - meanPhaseBias        - stdDevPhaseBias        - meanPhaseBiasRate        - stdDevPhaseBiasRate   3. gnss-Integrity-OrbitClockErrorBounds      + epochTime      + iod-ssr      + validityPeriod      + orbitClockErrorMeanShapeVector      + orbitClockErrorCovarianceShapeMatrix      + orbitClockRateErrorMeanShapeVector      + orbitClockRateErrorCovarianceShapeMatrix      + orbitClockErrorBounds List SEQUENCE (SIZE(1..64)) OF Integrity-OrbitClockErrorBoundsElement-r17        - svID        - orbitClockErrorScaleFactor        - orbitClockRateErrorScaleFactor   4. gnss-Integrity-IonosphereParameters      + epochTime      + iod-ssr      + validityPeriod      + pIonosphereFault      + tIonosphereFault      + tCorrelationIonosphere      + tCorrelationIonosphereRate   5. gnss-Integrity-IonosphereErrorBounds      + epochTime      + iod-ssr      + correctionPointSetID      + validityPeriod      + gridList SEQUENCE (SIZE(1..64)) OF Integrity-IonosphereGridElement-r17        - satList SEQUENCE (SIZE(1..64)) OF Integrity-IonosphereSatElement-r17          * svID          * meanIonosphere          * stdDevIonosphere          * meanIonosphereRate          * stdDevIonosphereRate |
| ESA et al [3] | Proposal 1. Add at least the quality indicator (standard deviation or variance) to each GNSS SSR IE in the Rel17 of LPP. Additional parameters are FFS at this moment. |
| Qualcomm [4] | Proposal 6: The assistance information that will be used to support integrity determination comprise quality indicators which can either be added to the SSR assistance data IEs or defined as a separate new IE:  - Uncertainty of the satellite orbit;  - Uncertainty of the satellite clocks;  - Uncertainty of the GNSS signal code bias;  - Uncertainty of the GNSS signal phase bias;  - Uncertainty of the ionosphere model;  - Uncertainty of the troposphere model; |

### Feared events in the GNSS Assistance Data

According to [1], the feared events in the GNSS assistance data is defined as incorrect computation of the GNSS Assistance Data and External feared event impacting the GNSS Assistance Data. While for the contributions submitted to the previous meeting, there are no proposals relating to this aspect. Hence, we propose the following question on whether assistance data for the feared event in GNSS assistance data should be defined, and if needed, what assistance data are needed.

###### Question1-4: Do we need to define assistance data for feared events in GNSS assistance data? If the answer is Yes, what parameters do companies think are needed for GNSS assistance data and please provide the TP for the parameters (including the field by ASN.1 and field description)? If the answer is No, please also clarify the reasons in the comments

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Swift Navigation | Yes (see comment) | The Alert parameters in [5] (e.g. Do Not Use (DNU) flags) are used to address both the ‘GNSS Feared Events’ and ‘Feared Events in the GNSS Assistance Data’. Although feared events may come from different sources, their net effect at the positioning function is the same, so they may share the same alert flag.  For example, a Satellite Vehicle (SV) alert means the system has detected a potential feared event on a given satellite which could impact integrity (meaning the error will likely exceed the corresponding Integrity Bound). This event may result from a satellite fault (e.g. a GNSS feared event) or from the inability of the service to validate the assistance data to a sufficient level for integrity (e.g. a Feared Event in the GNSS Assistance Data). Either way, the SV DNU flag is sent in the assistance data to avoid the corrections being used for the purpose of integrity.  NOTE: The DNU flag does not prohibit the user from continuing to use the corrections (SSR, RTK etc) to improve positioning accuracy (e.g. for applications which do not have an integrity requirement). DNU specifically refers to the usability for integrity. |
| Huawei, HiSIlicon | Yes | Some of the parameters can be used to indicate both GNSS feared events and GNSS assistance data feared events. |
| ZTE | Yes | At least the error bounds(orbit, clock, code bias, phase bias, ionosphere and troposphere) can be reused for feared events in GNSS assistance data. |
| Nokia |  | We would prefer to minimize the assistance data to be introduced. Error bounds such as the information in Q1-1 could be considered, but for others we prefer to first interact with RTCM before jumping to conclusions of adopting other types of assistance data. |
| CATT | Yes | Agree with Swift and ZTE. Some parameters defined in [5] can indicate both the external feared event impacting the GNSS Assistance Data and the GNSS feared events. The error bounds can be reused. |
| vivo | Yes | Parameters indicating both GNSS feared events and GNSS assistance data feared events can be used. |
| Xiaomi | Yes | Maybe a simple indication for both GNSS feared events and GNSS assistance feared events is sufficient. |
| InterDigital | Yes | Same view with Swift in that some flag(s) indicating whether the assistance data (e.g. corresponding to GNSS feared events and feared events in GNSS assistance data) is usable for integrity is beneficial. |
| Fraunhofer | Yes | The alerts can be global or local feared events. The GNSS feared events that are applicable locally needs to be signalled as having a local scope. Furthermore, in the alerts, the svDoNotUseFlag shall further specify the signals from this SV that should not be used. |
| ESA |  | Of course there will be need to introduce assistance data in LPP in support of GNSS positioning integrity. In doing this, we should aim at avoid unnecessary duplication in LPP: any item related to health of a GNSS Satellite and/or signal can already be signalled to UE by means of *GNSS-RealTimeIntegrity* IE. Therefore, we do not see the need to add new IEs achieving largely the same function.  We are also supportive on Nokia´s suggestion on waiting for RTCM to finish its work on this topic.  As a last note, [3] and [4] are proposing the same thing and should not be treated as different options. |
| Ericsson | Yes | We agree with Fraunhofer that NW ICE (Integrity Computing Entity) should be able to process the local feared event reported by UEs and convert these to do not use flags. |
| Qualcomm | No | …because the e.g., "Do Not Use" flag is already supported via IE *GNSS-RealTimeIntegrity* (which is mandatory):  "The location server shall always transmit the *GNSS-RealTimeIntegrity* with the current list of unhealthy signals (i.e., not only for signals/SVs currently visible at the reference location), for any GNSS positioning attempt and whenever GNSS assistance data are sent. If the number of bad signals is zero, then the *GNSS-RealTimeIntegrity* IE shall be omitted."  In addition, a UE does not expect "faulty" assistance data today; i.e., "incorrect assistance data" shall not be provided to the target device. |
| Ericsson |  | One comment to the last QC comment about UEs not expecting faulty AD. One key case is when data is available and seemingly current but not properly validated for integrity (e.g. if redundant observations for validation is temporary not available), then “do not use” for integrity assessments can still be relevant even if the provided data is used for positioning. |
| Samsung | See comment. | We basically agree with that at least the mean value for the error in Q1-1 can be inclucded in the AD. However, the basement on the mean value and error statistics were the fact that GNSS echo has made that value used. So it is better to interact with RTCM for further checking the parameters. |
| u-blox |  | This question is confusing. We understand it to be asking about error checks on the error check, and feel that it is not needed. What is important is to provide assistance data that includes quality measures (variance, validity time etc). It is important that this can be delivered with sufficiently reliability that we don’t need to add further checks on whether the quality measures were computed and delivered correctly. |
| Intel | Yes | Agree with Swift, DNU can be used to address both the ‘GNSS Feared Events’ and ‘Feared Events in the GNSS Assistance Data’. |

###### Question1-4 Summary:

The general opinion is that assistance data for feared events in GNSS assistance data is needed, with the following proposal

* Alert parameters can also be used for feared events in GNSS assistance data in addition to GNSS feared events
* QC thinks that there is no need to alert against feared events in GNSS assistance data because LMF would not send faulty assistance data to the UE.
* ESA and QC also mentioned the DNU already supported in the IE *GNSS-RealTimeIntegrity*  and mentioned that we should minimize the change to the LPP spec.
* Fraunhofer mentioned that there are global and local DNU and they need to be signaled independently.

Companies also think that we should minimize the assistance data to be introduced and prefer to wait for the RTCM conclusion. For this, as mentioned above, we will have another question in the phaseII on how to handle the discussion between 3GPP and RTCM.

With the above, we propose the following:

***Proposal1-4*: Alert parameters can also be used for feared events in GNSS assistance data in addition to GNSS feared events**

### GNSS feared events

At RAN2#115, there were proposals from several contributions [3][4][5][6] suggested to capture the following integrity assistance information and this was also captured in the summary document [2].

(1) Standard deviations, quality indicators, variances of the GNSS error sources.

(2) Mean values of the GNSS error sources.

(3) Information describing the time variation of the GNSS error sources.

(4) Probability of satellite fault.

(5) Probability of constellation fault.

(6) "Do Not Use" assistance data alerts

(7) "Do Not Use" SV and/or GNSS constellation alerts

In particular, the discussion in [4] has also proposed to categorize the integrity parameters as follows:

1. Integrity Bounds
2. Residual Risks
3. Correlation Times
4. Alerts
5. Validity Times

Then, from the rapporteur’s understanding, the integrity parameters listed in [2] can be mapped to the categories in [4] as follows:

* **Integrity bounds**

(1) Quality indicators (standard deviation or variance) of the GNSS error sources

(2) Mean values of the GNSS error sources.

* **Correlation Times**

(3) Information describing the time variation of the GNSS error sources.

* **Alerts**

(6) "Do Not Use" assistance data alerts

(7) "Do Not Use" SV and/or GNSS constellation alerts

* **Residual risk**

(4) Probability of satellite fault.

(5) Probability of constellation fault.

In addition, we think that the following assistance data have also been defined in [4] according to the categorization in [2]

* **Validity times**

*(8) “Validity Period” defined for Constellation, Ionosphere and Troposphere parameters and their error bounds.*

We would first like to ask the question whether companies think the above categorization is reasonable for the assistance date for GNSS feared event. From the rapporteur’s perspective, we think that understanding this question will help us having a better understanding of the structure of the assistance data for GNSS-feared event and can be helpful to capture in the stage2 spec.

###### Question1-5: Do companies agree that the assistance data for GNSS-feared event can be categorized into the five categories of (a)Integrity Bounds (b) Residual Risks (c) Correlation Times (d) Alerts (e) Validity Times?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Swift Navigation | Yes | ‘Applicability’ might be an alternative name for Validity Times, given some fields in this category are not only determined with respect to time (e.g. iod-ssr). However, these categories are only intended to aid interpretation of the concepts, whereas the actual naming and description of the fields will depend on the IEs we agree to define in the WI. Hence, we don’t think these specific categories need to be defined in the specification itself. |
| Huawei, HiSilicon | Yes | We agree with the categorization and whether it should be captured in the stage2 description can be decided by the spec editor |
| ZTE | Yes | Agree with the categorization, although we think only the Integrity Bounds is essential |
| Nokia | Yes | Such categorization would make our future discussions easier. |
| CATT | Yes | Agree with the five categories of the assistance data for GNSS-feared event. |
| vivo | Comments | These categories are only intended to aid the interpretation of the concepts and make future discussions clearer. Which categories will be selected and what are the actual namings of the selected categories depending on IEs we agree to define in the WI. |
| OPPO | Yes | Such catogrization is helpful for future discussion |
| Xiaomi | Yes | Agree with the categorization, but the detailed assistance information need to be further discussed. |
| InterDigital | Yes | We are ok with the proposed categorization of GNSS feared events |
| Fraunhofer | Yes | Categorisation is useful. |
| ESA | Yes | We like moderator´s suggestion, it will make discussions more structured in the future. |
| Ericsson | OK |  |
| Qualcomm | Yes | Agree with the categorization. However, the need/usage of each category requires more justification. |
| Samsung | Yes | Same view with QC. Categorization is ok but usage of each category requires more discussion . |
| u-blox |  | GNSS feared events can be categorized this way, The way in which assistance data are fitted into these criteria is for future discussion. |
| Intel | Yes |  |

###### Question1-5 Summary:

General opinions are that the assistance data can be categorized into the above categories. One company mentioned that the final categories of assistance data should depend on what are the final agreed assistance data for GNSS integrity. We hence propose the following for phase I. But if in Phase II, we manage to exclude certain categories, we can come back to this categorization.

Ublox ,QC and SS also think that which parameter for AD belongs to which category is up to further discussion.

***Proposal1-5*: the assistance data for GNSS-feared event can be categorized into the five categories of (a)Integrity Bounds (b) Residual Risks (c) Correlation Times (d) Alerts (e) Validity Times**

From the understanding of the rapporteur, with the parameters listed in the TP and the proposal, we think the proposal in [3] and [4] are already included in the current TP in [5].

Then, we would like to ask the following question:

###### Question1-6: Do companies agree that we need additional assistance data for GNSS feared event other than those defined in text proposal in [5]? If so, what additional parameters do companies think are needed for GNSS-feared events and please provide the TP for the parameters (including the field by ASN.1 and field description)?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Swift Navigation | No |  |
| Huawei, HiSilicon | No |  |
| ZTE | No |  |
| Nokia | No |  |
| CATT | No |  |
| vivo | No |  |
| OPPO | No |  |
| Xiaomi | No |  |
| InterDigital | No |  |
| Fraunhofer | No (comments) | Except that for svDoNotUseFlag, the following shall be added.  badSignalID GNSS-SignalIDs OPTIONAL, -- Need OP |
| ESA |  | We suggest to wait for RTCM to complete its work before moving further with definition of complicated IEs. |
| Ericsson | No | We are also fine with Fraunhofer’s suggestion above |
| Qualcomm | Yes | See our response to Question 1-2. |
| Samsung | No. | We think the list up parameters seems sufficient but which can be picked up still needs further discussion. |
| Intel | No |  |

###### Question1-6 Summary:

For the above question, general option is that no additional assistance data for GNSS feared event other than those defined in text proposal in [5]. QC mentioned about the comment to Question1-2. From the rapporteur’s understanding, AD for integrity service refers to AD that are not specific to certain feared events, but pertaining to the overall GNSS integrity feature, such as the AD for IR.

Fraunhofer mentioned the svNDU flag. However, the bad signal DNU has already been existing in the current spec. For these issues, we will ask further questions in the second phase of the discussion.

For this question, no proposal is formulated.

Assistance data for GNSS integrity service

In [5], Integrity Risk (IR) has been defined as assistance data for UE-based integrity calculation under the IE *GNSS-Integrity-ServiceParameters-r17* for the associated assistance data. This allows the UE to check if its TIR requirement (KPI) is within a range supported by the corrections service (based on the min/max IR).

We would like to ask the following question regarding assistance data for GNSS integrity service.

###### Question1-7: Do companies think other assistance data for GNSS integrity service is needed and please provide the corresponding TP (including the field by ASN.1 and field description)?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Swift Navigation | No | We believe *irMinimum* and *irMaximum* are the only service parameters needed currently. Note that the IR min/max are not actually used (directly) for the purpose of checking the TIR KPI. They are used for computing the bounds according to the formula from [5]:  [bound = ] *mean* + *K* \* *stdDev* where  *K* = *normInv*(*IRallocation* / 2)  *irMinimum < IRallocation < irMaximum* |
| Huawei, HiSilicon | No |  |
| ZTE | No |  |
| Nokia | No |  |
| CATT | No |  |
| vivo | No |  |
| OPPO | No |  |
| Xiaomi | No |  |
| InterDigital | No |  |
| Fraunhofer | No |  |
| ESA |  | We suggest to wait for RTCM to complete its work |
| Ericsson | No | We agree with Swift and if RTCM can provide timely input we can consider ESA’s suggestion, but we need to continue our work/progress in 3gpp. |
| Qualcomm |  | Similar to our response to Question 1-5, the need for any "Integrity Service" assistance data requires justification at first. |
| Samsung | No |  |
| Intel | No |  |

###### Question1-7 Summary:

For the above question, all the companies reply with No that they think assistance data for GNSS integrity service is not needed. Discussion on the issue with RTCM will continue in the second phase.

Hence, we propose the following:

***Proposal1-7*: The only needed assistance data for GNSS integrity service is Integrity Risk**

Any other issues

For the following open question, companies are invited to input any other issues relating to assistance data for GNSS integrity. We may be able to address the issue in phase II.

###### Question1-8: Do companies think there are other issues relating to assistance data of GNSS integrity?

|  |  |
| --- | --- |
| **Company** | **Comments** |
| ZTE | Since A-GNSS positioning supports UE-based and UE-assist(LMF-based) mode, we think LMF-based integrity should also be take into account. In LMF-based integrity, MT-LR based solution can be prioritized for discussion. |
| OPPO | Agree with ZTE. Besides, LMF-based integrity method may also reduce the power consumption of UE for integrity computation, compared with UE-based integrity method. Feared events transmission from UE towards LMF could be carried in new defined LPP msg or redefining current LPP Provide Assistance Data msg.  We think if online time is not enough, at least a email discussion on this topic containing pros and cons from each interesting company during next meeting is required. |
| Fraunhofer | Monitoring of interference (inter-system or jamming) or spoofing by capable UEs could provide valuable input for the system to determine the alert not to use a certain SV or a certain constellation or a certain GNSS signal. The network could utilise the report from capable Ues to generate such assistance data applicable locally. The mechanism for generating such assistance data can be left implementation specific, but the standard should support such optional reporting by capable Ues. |
| ESA | We agree with ZTE and OPPO. LMF-based integrity is most certainly possible: LMF has knowledge of GNSS errors and feared events (a requirement for UE-based) and all it needs is measurements from UE and indication of local feared events (e.g., multipath, etc.). From a conceptual point of view LPP already supports the transport of such information. New UE-feared events can also be reported to LMF as suggested by Fraunhofer. |
| Ericsson | We also agree that LMF-based integrity can be supported. Furthermore, we agree with Fraunhofer on interference monitoring by capable UEs for indications of local feared events that can be applied locally. |

###### Question1-8 Summary:

On the issue of LMF-based GNSS integrity calculation, please note that it is not under the scope of this email discussion. Companies who are in favour of this can submit contributions to the upcoming meetings to justify their reasons.

For the above feedbacks, no proposals are formulated.

Conclusion of Phase I

At the conclusion of the Phase I of the discussion, we propose the following

***Proposal1-1*: The paired overbounding technique is supported for bounding the error probability distribution for GNSS integrity as a baseline.**

***Proposal1-2*: Error representation by SSR is supported for GNSS integrity. FFS alignment with the assistance data for OSR in RTCM.**

***Proposal1-3*: The support for GNSS integrity in R16 is in-efficient for the use cases defined for GNSS integrity in TR 38.857 for R17**

***Proposal1-4*: Alert parameters can also be used for feared events in GNSS assistance data in addition to GNSS feared events**

***Proposal1-5*: Assistance data for GNSS-feared event can be categorized into the five categories of (a)Integrity Bounds (b) Residual Risks (c) Correlation Times (d) Alerts (e) Validity Times**

***Proposal1-7*: The only needed assistance data for GNSS integrity service is Integrity Risk**

# Discussion Phase II

Based on the results of the discussion in the first phase, in the second phase, we further discuss on following aspects of GNSS integrity assistance data

* Discuss on the signalling of the set of possible integrity assistance data for the 3 categories of GNSS-feared events
* Discuss on the signalling of the assistance data for GNSS integrity services
* Discuss the relationship between 3GPP and RTCM for the discussion on GNSS integrity assistance data

Placement of the GNSS integrity assistance data in LPP

In [5], it has been proposed to (a) put assistance date for *GNSS integrity service and troposphere* under ***common assistance data*** and (b) put those for *constellation and ionosphere* under ***generic assistance data***, with the following text proposal:

==========================TEXT PROPOSAL BEGIN=================================

#### – *GNSS-CommonAssistData*

The IE *GNSS-CommonAssistData* is used by the location server to provide assistance data which can be used for any GNSS.

-- ASN1START

GNSS-CommonAssistData ::= SEQUENCE {

gnss-ReferenceTime GNSS-ReferenceTime OPTIONAL, -- Need ON

gnss-ReferenceLocation GNSS-ReferenceLocation OPTIONAL, -- Need ON

gnss-IonosphericModel GNSS-IonosphericModel OPTIONAL, -- Need ON

gnss-EarthOrientationParameters GNSS-EarthOrientationParameters OPTIONAL, -- Need ON

...,

[[

gnss-RTK-ReferenceStationInfo-r15

GNSS-RTK-ReferenceStationInfo-r15 OPTIONAL, -- Need ON

gnss-RTK-CommonObservationInfo-r15

GNSS-RTK-CommonObservationInfo-r15 OPTIONAL, -- Cond RTK

gnss-RTK-AuxiliaryStationData-r15

GNSS-RTK-AuxiliaryStationData-r15 OPTIONAL -- Need ON

]],

[[

gnss-SSR-CorrectionPoints-r16

GNSS-SSR-CorrectionPoints-r16 OPTIONAL -- Need ON

]] ,

[[

gnss-Integrity-ServiceParameters-r17

GNSS-Integrity-ServiceParameters-r17 OPTIONAL, -- Need ON

gnss-Integrity-ServiceAlert-r17

GNSS-Integrity-ServiceAlert-r17 OPTIONAL, -- Need ON

gnss-Integrity-TroposphereParameters-r17

GNSS-Integrity-TroposphereParameters-r17 OPTIONAL, -- Need ON

gnss-Integrity-TroposphereErrorBounds-r17

GNSS-Integrity-TroposphereErrorBounds-r17 OPTIONAL -- Need ON

]]

}

-- ASN1STOP

| Conditional presence | Explanation |
| --- | --- |
| *RTK* | The field is mandatory present if the IE *GNSS-RTK-Observations* is included in IE *GNSS‑GenericAssistData*; otherwise it is not present. |

#### – *GNSS-GenericAssistData*

The IE *GNSS-GenericAssistData* is used by the location server to provide assistance data for a specific GNSS. The specific GNSS for which the provided assistance data are applicable is indicated by the IE *GNSS‑ID* and (if applicable) by the IE *SBAS‑ID*. Assistance for up to 16 GNSSs can be provided.

-- ASN1START

GNSS-GenericAssistData ::= SEQUENCE (SIZE (1..16)) OF GNSS-GenericAssistDataElement

GNSS-GenericAssistDataElement ::= SEQUENCE {

gnss-ID GNSS-ID,

sbas-ID SBAS-ID OPTIONAL, -- Cond GNSS-ID-SBAS

gnss-TimeModels GNSS-TimeModelList OPTIONAL, -- Need ON

gnss-DifferentialCorrections GNSS-DifferentialCorrections OPTIONAL, -- Need ON

gnss-NavigationModel GNSS-NavigationModel OPTIONAL, -- Need ON

gnss-RealTimeIntegrity GNSS-RealTimeIntegrity OPTIONAL, -- Need ON

gnss-DataBitAssistance GNSS-DataBitAssistance OPTIONAL, -- Need ON

gnss-AcquisitionAssistance GNSS-AcquisitionAssistance OPTIONAL, -- Need ON

gnss-Almanac GNSS-Almanac OPTIONAL, -- Need ON

gnss-UTC-Model GNSS-UTC-Model OPTIONAL, -- Need ON

gnss-AuxiliaryInformation GNSS-AuxiliaryInformation OPTIONAL, -- Need ON

...,

[[

bds-DifferentialCorrections-r12

BDS-DifferentialCorrections-r12 OPTIONAL, -- Cond GNSS-ID-BDS

bds-GridModel-r12 BDS-GridModelParameter-r12 OPTIONAL -- Cond GNSS-ID-BDS

]],

[[

gnss-RTK-Observations-r15 GNSS-RTK-Observations-r15 OPTIONAL, -- Need ON

glo-RTK-BiasInformation-r15 GLO-RTK-BiasInformation-r15 OPTIONAL, -- Cond GNSS-ID-GLO

gnss-RTK-MAC-CorrectionDifferences-r15

GNSS-RTK-MAC-CorrectionDifferences-r15

OPTIONAL, -- Need ON

gnss-RTK-Residuals-r15 GNSS-RTK-Residuals-r15 OPTIONAL, -- Need ON

gnss-RTK-FKP-Gradients-r15 GNSS-RTK-FKP-Gradients-r15 OPTIONAL, -- Need ON

gnss-SSR-OrbitCorrections-r15

GNSS-SSR-OrbitCorrections-r15 OPTIONAL, -- Need ON

gnss-SSR-ClockCorrections-r15

GNSS-SSR-ClockCorrections-r15 OPTIONAL, -- Need ON

gnss-SSR-CodeBias-r15 GNSS-SSR-CodeBias-r15 OPTIONAL -- Need ON

]],

[[

gnss-SSR-URA-r16 GNSS-SSR-URA-r16 OPTIONAL, -- Need ON

gnss-SSR-PhaseBias-r16 GNSS-SSR-PhaseBias-r16 OPTIONAL, -- Need ON

gnss-SSR-STEC-Correction-r16 GNSS-SSR-STEC-Correction-r16

OPTIONAL, -- Need ON

gnss-SSR-GriddedCorrection-r16 GNSS-SSR-GriddedCorrection-r16

OPTIONAL, -- Need ON

navic-DifferentialCorrections-r16 NavIC-DifferentialCorrections-r16

OPTIONAL, -- Cond GNSS-ID-NavIC

navic-GridModel-r16 NavIC-GridModelParameter-r16

OPTIONAL -- Cond GNSS-ID-NavIC

]] ,

[[

gnss-Integrity-ConstellationAlert-r17

GNSS-Integrity-ConstellationAlert-r17 OPTIONAL, -- Need ON

gnss-Integrity-ConstellationParameters-r17

GNSS-Integrity-ConstellationParameters-r17 OPTIONAL, -- Need ON

gnss-Integrity-BiasErrorBounds-r17

GNSS-Integrity-BiasErrorBounds-r17 OPTIONAL, -- Need ON

gnss-Integrity-OrbitClockErrorBounds-r17

GNSS-Integrity-OrbitClockErrorBounds-r17 OPTIONAL, -- Need ON

gnss-Integrity-IonosphereParameters-r17

GNSS-Integrity-IonosphereParameters-r17 OPTIONAL, -- Need ON

gnss-Integrity-IonosphereErrorBounds-r17

GNSS-Integrity-IonosphereErrorBounds-r17 OPTIONAL -- Need ON

]]

}

-- ASN1STOP

| Conditional presence | Explanation |
| --- | --- |
| *GNSS‑ID‑SBAS* | The field is mandatory present if the *GNSS‑ID* = *sbas*; otherwise it is not present. |
| *GNSS‑ID‑BDS* | The field may be present if the *GNSS‑ID* = *bds*; otherwise it is not present. |
| *GNSS-ID-GLO* | The field is optionally present, need ON, if the *GNSS ID* = *glonass*; otherwise it is not present. |
| *GNSS-ID-NAVIC* | The field is optionally present, need ON, if the *GNSS‑ID* = *navic*; otherwise it is not present |

==========================TEXT PROPOSAL END=================================

The rapporteur would like to ask the following question regarding the organization of GNSS integrity assistance data in LPP.

###### Question2-1: Regarding the organization of the assistance data for GNSS integrity, do companies agree that we should (a) put assistance date for GNSS integrity service and troposphere under common assistance data and (b) put those for constellation and ionosphere under generic assistance data?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Yes | But the ASN.1 details need double check, e.g. Dash is not needed after Integrity in the fields. |
| Qualcomm |  | First, the need, usage, etc. of all these additional assistance data elements needs to be clarified.  Generally, GNSS common assistance data (i.e., GNSS independent) should be under "common assistance data", and GNSS specific assistance data under "generic assistance data".  However, it is not quite clear why all these new assistance data are needed and whether they are all needed individually. E.g., a single new "common assistance data" and a single new "generic assistance data" element should be sufficient, which would require two new posSIBs only.  However, integration of the new assistance data into existing assistance data would be preferred, if possible. E.g., most new elements have an *iod-ssr* to specifiy the SSR data that the integrity values are applicable to. Why not put the new data directly into the SSR data instead of linking them together with an *iod­-ssr*?  With this proposal, we would need 10 new posSIBs just for GNSS integrity support! |
| CATT | No | 1. Suggestion on GNSS-CommonAssistData:  #1:  [[  gnss-IntegrityParameters-r17  GNSS-IntegrityParameters-r17 OPTIONAL, -- Need ON    L -- Need ON  ]]  #2:  Usually the service parameters are sent via RequestLocationInformation, especially in *CommonIEsRequestLocationInformation*. Why are the service parameters put in GNSS-CommonAssistData?  According to the wording in 3.4 Assistance data for GNSS integrity service above: “In [5], Integrity Risk (IR) has been defined as assistance data for UE-based integrity calculation under the IE *GNSS-Integrity-ServiceParameters-r17* for the associated assistance data. This allows the UE to check if its TIR requirement (KPI) is within a range supported by the corrections service (based on the min/max IR). ” it seems that this service data is only for UE-based integrity calculation. This kind of parameters should be put in *CommonIEsRequestLocationInformation.* Please refer to QoS for UE-based location calculation defined in *CommonIEsRequestLocationInformation.*  2. Question on GNSS-GenericAssistData:  Why IonosphereParameters are put in GNSS-GenericAssistData? Please refer to the gnss-IonosphericModel defined in GNSS-CommonAssistData. |
| Swift Navigation | Yes | As per the comments from QC, the messages are divided into CommonAssist (GNSS-independent) and GenericAssist (GNSS-specific). On the second point raised by CATT, the IonosphericModel under CommonAssist is for single-frequency coarse positioning. For high accuracy, the *GNSS-SSR-STEC-Correction* (GenericAssist) is used to map the slant path.  Also agree with QC that some consolidation with existing IEs could be helpful to reduce the number of posSIBs and to streamline. Some options and trade-offs are presented:  **New IEs required in LPP**  *gnss-Integrity-ServiceParameters-r17*  *gnss-Integrity-ServiceAlert-r17*  *gnss-Integrity-ConstellationAlert-r17*  > New IEs needed because the Service Parameters & Alerts are a specific feature of Integrity.  **IEs that could be consolidated with existing SSR messages:**  *gnss-Integrity-TroposphereErrorBounds-r17*  *gnss-Integrity-BiasErrorBounds-r17*  *gnss-Integrity-OrbitClockErrorBounds-r17*  *gnss-Integrity-IonosphereErrorBounds-r17*  > The *epochTime* and *iod-SSR* fields could be reused from the SSR messages  **TBD if these IEs should be consolidated with existing SSR messages:**  gnss-Integrity-TroposphereParameters-r17  gnss-Integrity-ConstellationParameters-r17  gnss-Integrity-IonosphereParameters-r17  > The parameters only need to be updated infrequently so we should discuss if the savings from reducing the number of posSIBs is worth the extra bandwidth of sending the same message on a regular basis.  > Another option is to consolidate these parameters into one message / IE meaning only one posSIB is required and we can send less frequently.  On the first point raised by CATT, we are open to moving the service parameters out of the assistance data if this is more appropriate, however we note that these parameters are not directly related to the TIR KPI. These are static parameters provided by the Network that are used in computing the bounds according to the formula from [5]:  [bound = ] *mean* + *K* \* *stdDev* where  *K* = *normInv*(*IRallocation* / 2)  *irMinimum < IRallocation < irMaximum* |
| ESA | Not before confirmation of alignment with RTCM | We believe we should coordinate with RTCM before agreeing on any TP for Stage 3.  Many of the IEs in this TP come without a justification and their use is not clear. |
| Nokia |  | Agree with ESA |
| ZTE | No | Firstly, we think gnss-Integrity-IonosphereParameters-r17 and gnss-Integrity-IonosphereErrorBounds-r17 should be in common assistance data.  Also, we agree with ESA that detailed ASN.1 description should be determined after coordination with RTCM. |
| vivo |  | Agree with QC. |
| u-blox |  | We think that the split is in general reasonable but we agree with the views put forward for consolidation into fewer posSIBs and/or inclusion of the integrity parameters with the corrections themselves.  We would like to see coordination with RTCM, but their timeline does not match ours and we would not want to see the work held up waiting for RTCM to complete their work first. |
| MELCO | Yes | Agree with Swift Navigation. |
| Ericsson | Yes, but | When it comes to posSIB; we need to deifne one or two common posSIBs which can then map to Integrity AD. So, the approach should not be to have one posSIB per AD but rather a common posSIB.  For SSR based Integrity, we think 3gpp should move forward but when it comes to OSR then we may have possibility to synch with OSR later.  We need to send our SSR Integrity work to RTCM so they can also align with ours. |

###### Question2-1 Summary

TBD

GNSS-feared event assistance data

### Constellation and its error bounding parameters

In [5], the following TP has been provided for the low update rate integrity parameters related to the satellite and constellation fault probabilities.

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-ConstellationParameters*

The IE *GNSS-Integrity-ConstellationParameters* is used by the location server to provide low update rate integrity parameters related to the satellite and constellation fault probabilities. Bounding parameters are not included in this message but in the *GNSS-Integrity-BiasErrorBounds* and *GNSS-Integrity-OrbitClockErrorBounds* IEs*.*

-- ASN1START

GNSS-Integrity-ConstellationParameters-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

validityPeriod-r17 CHOICE {

validityPeriodSeconds-r17 INTEGER (1..86400),

validityPeriodDays-r17 INTEGER (1..365)

},

pConstellationFault-r17 INTEGER (0..255),

tConstellationFault-r17 INTEGER (1..3600),

pSatelliteFault-r17 INTEGER (0..255),

tSatelliteFault-r17 INTEGER (1..3600),

tCorrelationRangeOrbit-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

tCorrelationRangeClock-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

tCorrelationRangeRateOrbit-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

tCorrelationRangeRateClock-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

...

}

-- ASN1STOP

|  |  |
| --- | --- |
| **Conditional presence** | **Explanation** |
| *seq* | The field may be present if the integrity service supports time-based estimation techniques such as Kalman filtering as the user must be provided with information about the time correlation of errors. |

|  |
| --- |
| ***GNSS-Integrity-ConstellationParameters* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***Iod-ssr***  This field specifies the Issue of Data number for the SSR data that the integrity values are applicable to. |
| ***validityPeriodSeconds***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***validityPeriodDays***  This field specifies the Validity Duration in days. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*. A day is defined to be 86,400 seconds.  Scale factor 1 day; range 1-365 days. |
| ***pConstellationFault***  This field specifies the Probability of Onset of Constellation Fault per Time Unit where a constellation fault is at least two satellites being faulty simultaneously due to the same event. This field specifies the onset probability that the residual range or range rate error exceeds a bound created using the minimum allowed inflation factor *Kmin*, and bounding parameters as *mean* + *Kmin* \* *stdDev* where *Kmin* = *normInv*(*irMaximum*/2).  The probability is calculated by where *n* is the value of *pConstellationFault* and the range is 10-10.2 to 1 per hour. |
| ***tConstellationFault***  This field specifies the Mean Constellation Fault Duration which is the mean duration between when a constellation fault occurs, and the user is alerted by the service through the DNU flags (or the integrity violation is over).  Scale factor 1 s; range 1-3,600 s. |
| ***pSatelliteFault***  This field specifies the Probability of Onset of Satellite Fault per Time Unit which is the probability of occurrence of satellite error to exceed the residual error bound for more than the Time to Alert (TTA).  This field specifies the onset probability that the residual range or range rate error exceeds a bound created using the minimum allowed inflation factor *Kmin*, and bounding parameters as *mean* + *Kmin* \* *stdDev* where *Kmin* = *normInv*(*irMaximum* / 2).  The probability is calculated by where *n* is the value of *pSatelliteFault* and the range is 10-10.2 to 1 per hour. |
| ***tSatelliteFault***  This field specifies the Mean Satellite Fault Duration which is the mean duration between when a satellite fault occurs, and the user is alerted by the service through the DNU flags (or the integrity violation is over).  Scale factor 1 s; range 1-3,600 s. |
| ***tCorrelationRangeOrbit***  This field specifies the Orbit Range Error Correlation Time which is the upper bound of the correlation time of the satellite residual range error due to orbit.  The time is calculated using:  Range is 1-28,200 s. |
| ***tCorrelationRangeClock***  This field specifies the Clock Range Error Correlation Time which is the upper bound of the correlation time of the satellite residual range error due to clock.  The time is calculated using:  Range is 1-28,200 s. |
| ***tCorrelationRangeRateOrbit***  This field specifies the Orbit Range Rate Error Correlation Time which is the upper bound of the correlation time of the satellite residual range rate error due to orbit.  The time is calculated using:  Range is 1-28,200 s. |
| ***tCorrelationRangeRateClock***  This field specifies the Clock Range Rate Error Correlation Time which is the upper bound of the correlation time of the satellite residual range rate error due to clock.  The time is calculated using:  Range is 1-28,200 s. |

==========================TEXT PROPOSAL END=================================

We would like the companies to check on the above text proposal and verify if it is correct.

###### Question2-2: Do companies agree with the above text proposal for the constellation parameters?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the Ies according to the previous discussion results at first, so we can figure out if these Ies are required. Too early to define the stage3 Ies. |
| Swift Navigation | Yes | We agree there is still a need for some additional justification in addition to the background and literature provided so far, e.g. [R2-2006541](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_111-e/Docs/R2-2006541.zip), [R2-2103954](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_113bis-e/Docs/R2-2103954.zip), [R2-2106105](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_114-e/Docs/R2-2106105.zip), [R2-2108340](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_115-e/Docs/R2-2108340.zip), [R2-2108385](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_115-e/Docs/R2-2108385.zip), [5], [6]. It would be helpful to the discussion if companies can provide more specific guidance on which topics they feel need further discussion or justification. In particular we think more detail is needed to expose how these Ies are to be used by the UE, beyond that which is already specified in the parameter descriptions. Swift also intends to suggest more detail on these points in upcoming contributions.  Generally speaking, the *Integrity-ConstellationParameters* are necessary for the UE to determine if its internal assumptions on the integrity risks associated with the Satellite and Constellation bounds are compatible with the assumptions that have been made when deriving these bounds at the Network (as indicated by the Residual Risk parameter). This is a very standard concept in the field of GNSS integrity including for ARAIM (Psat, Pconst). We can add these descriptions to the Stage 2 specifications (TS 38.305) (i.e. to further describe what the UE should do with this data).  For consistency with Question 1-5, we think it may be helpful to rename the pConstellationFault, tConstellationFault, pSatelliteFault, tSatelliteFault as follows, which is more consistent with industry terminology and represents that *p* corresponds to the probability (i.e. the Residual Risk):  *pConstellation, tConstellation, pSatellite, tSatellite* |
| ESA | Not before confirmation of alignment with RTCM | We believe we should coordinate with RTCM before agreeing on any TP for Stage 3.  P.S. dropping the word “Fault” from some of the fields is ok to us. |
| Nokia |  | Agree with Qualcomm and ESA |
| ZTE |  | Agree with QC, CATT and ESA |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think some clarification is required. For example a constellation comprises satellites from multiple generations; each generation is typically more advanced than the next with improved clocks etc. There does not appear to be a way of linking the coherence times to SVs, only to constellations. Or, possibly our understanding of how they would be used in the UE is incomplete. |
| MELCO | Partially Yes | pConstellationFault and pSatelliteFault are parameters well-known in GNSS community and used in an well-described algorithm as ARAIM. There should be no problem to standardize them.  epochTime, iod-ssr, validityPeriodSeconds, validityPeriodDays are supplimentaly parameters to let user to use the above parameters consistently. There should be no problem to standardize them together with the above parameters.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-2 Summary

TBD

Then, the bounding parameters for bias error are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-BiasErrorBounds*

The IE *GNSS-Integrity-BiasErrorBounds* is used by the location server to provide integrity bounding parameters relating to the satellite code bias, code bias rate, phase bias and phase bias rate residual errors after application of the SSR corrections.

-- ASN1START

GNSS-Integrity-BiasErrorBounds-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

validityPeriod-r17 CHOICE {

validityPeriodSeconds-r17 INTEGER (1..86400),

validityPeriodDays-r17 INTEGER (1..365)

},

integrity-biasErrorBoundsList-r17 Integrity-BiasErrorBoundsList-r17,

...

}

Integrity-BiasErrorBoundsList-r17 ::= SEQUENCE (SIZE(1..64)) OF

Integrity-BiasErrorBoundsElement-r17

Integrity-BiasErrorBoundsElement-r17 ::= SEQUENCE {

svID-r17 SV-ID,

meanCodeBias-r17 INTEGER (0..255),

stdDevCodeBias-r17 INTEGER (0..255),

meanCodeBiasRate-r17 INTEGER (0..255),

stdDevCodeBiasRate-r17 INTEGER (0..255),

meanPhaseBias-r17 INTEGER (0..255),

stdDevPhaseBias-r17 INTEGER (0..255),

meanPhaseBiasRate-r17 INTEGER (0..255),

stdDevPhaseBiasRate-r17 INTEGER (0..255),

...

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-BiasErrorBounds* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***validityPeriodSeconds***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***validityPeriodDays***  This field specifies the Validity Duration in days. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*. A day is defined to be 86,400 seconds.  Scale factor 1 day; range 1-365 days. |
| ***svID***  This field specifies the satellite for which bias error bounds are provided. |
| ***meanCodeBias***  This field specifies the Code Bias Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual code bias error.  The bound is *meanCodeBias* + *K* \* *stdDevCodeBias* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.005 m; range 0-1.275 m. |
| ***stdDevCodeBias***  This field specifies the Code Bias Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual code bias error.  Scale factor 0.005 m; range 0-1.275 m. |
| ***meanCodeBiasRate***  This field specifies the Code Bias Rate Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual code bias rate error.  The bound is *meanCodeBiasRate* + *K* \* *stdDevCodeBiasRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***stdDevCodeBiasRate***  This field specifies the Code Bias Rate Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual code bias rate error.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***meanPhaseBias***  This field specifies the Phase Bias Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual phase bias error.  The bound is *meanPhaseBias* + *K* \* *stdDevPhaseBias* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.005 m; range 0-1.275 m. |
| ***stdDevPhaseBias***  This field specifies the Phase Bias Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual phase bias error.  Scale factor 0.005 m; range 0-1.275 m. |
| ***meanPhaseBiasRate***  This field specifies the Phase Bias Rate Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual phase bias rate error.  The bound is *meanPhaseBiasRate* + *K* \* *stdDevPhaseBiasRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***stdDevPhaseBiasRate***  This field specifies the Phase Bias Rate Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual phase bias rate error.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |

==========================TEXT PROPOSAL END=================================

###### Question2-3: Do companies agree with the above text proposal for the bounding parameters for bias error?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the IEs according to the previous discussion results at first, so we can figure out if these IEs are required. Too early to define the stage3 IEs. |
| Swift Navigation | Yes | We also refer to the information provided in our response to Question 2-2. Generally speaking, the *Integrity-BiasErrorBounds* are used to statistically bound the residual Code and Phase Bias errors after the positioning corrections (e.g. RTK, SSR) have been applied. We can add these descriptions to the Stage 2 specifications (TS 38.305). |
| ESA | Not before confirmation of alignment with RTCM | Generally speaking we do not oppose to the bounding parameters for code and phase measurements bias errors by providing both mean value and standard deviation.  Since these are directly related to SSR data we would like to understand why not adding the mean and std values directly to the associated SSR IEs instead of creating new IEs? We think this TP leads to an increase in the number of posSIB (and scheduling issues).  Lastly, we wish to have formal confirmation from RTCM that there is alignment between our work and theirs. |
| Nokia |  | Agree with Qualcomm |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | A better understanding about how the corrections are to be supplied and consumed would be useful. For example the entire IE *GNSS-Integrity-BiasErrorBounds-r17* is OPTIONAL, but the data elements within it are not, therefore the service has to be capable of providing all fields or not using the IE at all. We also wonder whether paired overbounding is the most appropriate statistic for biases. It will be useful to gain a clearer understanding of how the fields are arrived at, how they will be generated, consumed and tested before finalizing.  Some more time to study the proposals is needed, but we feel that they are going in the right direction and and are generally supportive. |
| MELCO | Partially Yes | meanCodeBias and stdDevCodeBias are natural extension of parameters used in an well-described algorithm as ARAIM. There should be no problem to standardize them once how to map to those of well-described algorithm is explained.  epochTime, iod-ssr, validityPeriodSeconds, validityPeriodDays, svID are supplementary parameters to let user to use above parameters consistently. There should be no problem to standardize them together with the above parameters.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-3 Summary

TBD

Then, the bounding parameters for orbit clock error are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-OrbitClockErrorBounds*

The IE *GNSS-Integrity-OrbitClockErrorBounds* is used by the location server to provide integrity bounding parameters relating to the orbit, orbit rate, clock and clock rate residual errors after application of the SSR corrections.

-- ASN1START

GNSS-Integrity-OrbitClockErrorBounds-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

validityPeriod-r17 INTEGER (1..86400),

orbitClockErrorMeanShapeVector-r17 Integrity-MeanVector-r17,

orbitClockErrorCovarianceShapeMatrix-r17 Integrity-CovarianceMatrix-r17,

orbitClockRateErrorMeanShapeVector-r17 Integrity-MeanVector-r17,

orbitClockRateErrorCovarianceShapeMatrix-r17 Integrity-CovarianceMatrix-r17,

orbitClockErrorBoundsList-r17 Integrity-OrbitClockErrorBoundsList-r17,

…

}

Integrity-CovarianceMatrix-r17 ::= SEQUENCE (SIZE(10)) OF INTEGER (0..250)

Integrity-MeanVector-r17 ::= SEQUENCE (SIZE(4)) OF INTEGER (0..250)

Integrity-OrbitClockErrorBoundsList-r17 ::= SEQUENCE (SIZE(1..64)) OF

Integrity-OrbitClockErrorBoundsElement-r17

Integrity-OrbitClockErrorBoundsElement-r17 ::= SEQUENCE {

svID-r17 SV-ID,

orbitClockErrorScaleFactor-r17 INTEGER (1..255),

orbitClockRateErrorScaleFactor-r17 INTEGER (1..250),

…

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-OrbitClockErrorBounds* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***Iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***validityPeriod***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***orbitClockErrorMeanShapeVector***  This field specifies the Satellite Orbit and Clock Residual Error Bounds Mean Shape Vector which defines the mean parameter for a set of four paired overbounding models that bound the residual satellite orbit and clock error.  The 4 x 1 vector is normalised using the same normalisation factor as *orbitClockErrorCovarianceShapeMatrix*.  The 4 random variables are defined as:   * *A* – along track orbit error * *X* – across track orbit error * *R* – radial orbit error * *C* – clock error   The normalised values are transmitted in the following order:   * *orbitClockErrorMeanShapeVector*[0] = *mean*(*A*) * *orbitClockErrorMeanShapeVector*[1] = *mean*(*X*) * *orbitClockErrorMeanShapeVector*[2] = *mean*© * *orbitClockErrorMeanShapeVector*[3] = *mean*©   Scale factor 0.004; range 0-1. |
| ***orbitClockErrorCovarianceShapeMatrix***  This field specifies the Satellite Orbit and Clock Residual Error Bounds Covariance Shape Matrix which defines the covariance parameters for a set of four paired overbounding models that bound the residual satellite orbit and clock error.  The 4 x 4 covariance matrix is normalised using the same normalisation factor as *orbitClockErrorMeanShapeVector* and only the 10 upper right values are transmitted.  The 4 random variables are defined the same as *orbitClockErrorMeanShapeVector.*  The normalised values are transmitted in the following order:   * *orbitClockErrorCovarianceShapeMatrix*[0] = *cov*(*A*, *A*) (or *var*(*A*)) * *orbitClockErrorCovarianceShapeMatrix*[1] = *cov*(*A*, *X*) * *orbitClockErrorCovarianceShapeMatrix*[2] = *cov*(*A*, *R*) * *orbitClockErrorCovarianceShapeMatrix*[3] = *cov*(*A*, *C*) * *orbitClockErrorCovarianceShapeMatrix*[4] = *cov*(*X*, *X*) (or *var*(*X*)) * *orbitClockErrorCovarianceShapeMatrix*[5] = *cov*(*X*, *R*) * *orbitClockErrorCovarianceShapeMatrix*[6] = *cov*(*X*, *C*) * *orbitClockErrorCovarianceShapeMatrix*[7] = *cov*(*R*, *R*) (or *var*©) * *orbitClockErrorCovarianceShapeMatrix*[8] = *cov*(*R*, *C*) * *orbitClockErrorCovarianceShapeMatrix*[9] = *cov*(*C*, *C*) (or *var*©)   Scale factor 0.004; range 0-1. |
| ***orbitClockRateErrorMeanShapeVector***  This field specifies the Satellite Orbit and Clock Residual Rate Error Bounds Mean Shape Vector which defines the mean parameter for a set of four paired overbounding models that bound the residual satellite orbit and clock rate error.  The 4 x 1 vector is normalised using the same normalisation factor as *orbitClockRateErrorCovarianceShapeMatrix*.  The 4 random variables are defined as:   * *A’* – along track orbit rate error * *X’* – across track orbit rate error * *R’* – radial orbit rate error * *C’* – clock rate error   The normalised values are transmitted in the following order:   * *orbitClockRateErrorMeanShapeVector*[0] = *mean*(*A’*) * *orbitClockRateErrorMeanShapeVector*[1] = *mean*(*X’*) * *orbitClockRateErrorMeanShapeVector*[2] = *mean*(*R’*) * *orbitClockRateErrorMeanShapeVector*[3] = *mean*(*C’*)   Scale factor 0.004; range 0-1. |
| ***orbitClockRateErrorCovarianceShapeMatrix***  This field specifies the Satellite Orbit and Clock Rate Residual Error Bounds Covariance Shape Matrix which defines the covariance parameters for a set of four paired overbounding models that bound the residual satellite orbit and clock rate error.  The 4 random variables are defined the same as *orbitClockRateErrorMeanShapeVector.*  The 4 x 4 covariance matrix is normalised using the same normalisation factor as *orbitClockRateErrorMeanShapeVector* and only the 10 upper right values are transmitted.  The normalised values are transmitted in the following order:   * *orbitClockRateErrorCovarianceShapeMatrix*[0] = *cov*(*A’*, *A’*) (or *var*(*A’*)) * *orbitClockRateErrorCovarianceShapeMatrix*[1] = *cov*(*A’*, *X’*) * *orbitClockRateErrorCovarianceShapeMatrix*[2] = *cov*(*A’*, *R’*) * *orbitClockRateErrorCovarianceShapeMatrix*[3] = *cov*(*A’*, *C’*) * *orbitClockRateErrorCovarianceShapeMatrix*[4] = *cov*(*X’*, *X’*) (or *var*(*X’*)) * *orbitClockRateErrorCovarianceShapeMatrix*[5] = *cov*(*X’*, *R’*) * *orbitClockRateErrorCovarianceShapeMatrix*[6] = *cov*(*X’*, *C’*) * *orbitClockRateErrorCovarianceShapeMatrix*[7] = *cov*(*R’*, *R’*) (or *var*(*R’*)) * *orbitClockRateErrorCovarianceShapeMatrix*[8] = *cov*(*R’*, *C’*) * *orbitClockRateErrorCovarianceShapeMatrix*[9] = *cov*(*C’*, *C’*) (or *var*(*C’*))   Scale factor 0.004; range 0-1. |
| ***svID***  This field specifies the satellite for which error bounds scale factors are provided. |
| ***orbitClockErrorScaleFactor***  This field specifies the Satellite Orbit and Clock Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockErrorCovarianceShapeMatrix* and *orbitClockErrorMeanShapeVector* to restore the full values of the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClock* = *orbitClockErrorMeanShapeVector*[3] \* *orbitClockErrorScaleFactor* * *stdDevClock* = *sqrt*(*orbitClockErrorCovarianceShapeMatrix* [9] \* *orbitClockErrorScaleFactor)*   The bound is *meanClock* + *K* \* *stdDevClock* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  The scale factor is calculated using:  Range is 0.025-55 m. |
| ***orbitClockRateErrorScaleFactor***  This field specifies the Satellite Orbit and Clock Rate Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockRateErrorCovarianceShapeMatrix* and *orbitClockRateErrorBiasVector* to restore the full values of the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClockRate* = *orbitClockRateErrorMeanShapeVector*[3] \* *orbitClockRateErrorScaleFactor* * *stdDevClockRate* = *sqrt*(*orbitClockRateErrorCovarianceShapeMatrix* [9] \* *orbitClockRateErrorScaleFactor)*   The bound is *meanClockRate* + *K* \* *stdDevClockRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.001 m/s; range 0.001-0.255 m/s. |

==========================TEXT PROPOSAL END=================================

###### Question2-4: Do companies agree with the above text proposal for the bounding parameters for orbit clock error?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the Ies according to the previous discussion results at first, so we can figure out if these Ies are required. Too early to define the stage3 Ies. |
| Swift Navigation | Yes, with comments | Further information is provided in our response to Question 2-2. Generally speaking, the *Integrity-OrbitClockErrorBounds* are used to statistically bound the residual Orbit and Clock errors after the positioning corrections (e.g. RTK, SSR) have been applied. We can add these descriptions to the Stage 2 specifications (TS 38.305).  We also think a separate scale factor is needed for the mean and covariance:  Integrity-OrbitClockErrorBoundsElement-r17 ::= SEQUENCE {  svID-r17 SV-ID,  orbitClockErrorCovarianceScaleFactor-r17 INTEGER (1..255), orbitClockErrorMeanScaleFactor-r17 INTEGER (1..255),  orbitClockRateErrorCovarianceScaleFactor-r17 INTEGER (1..250),  orbitClockRateErrorMeanScaleFactor-r17 INTEGER (1..250)…  }  -- ASN1STOP |
| ***orbitClockErrorCovarianceScaleFactor***  This field specifies the Satellite Orbit and Clock Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockErrorCovarianceShapeMatrix* to restore the full values of the covariance in the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClock* = *orbitClockErrorMeanShapeVector*[3] \* *orbitClockErrorMeanScaleFactor* * *stdDevClock* = *sqrt*(*orbitClockErrorCovarianceShapeMatrix* [9] \* *orbitClockErrorCovarianceScaleFactor)*   The bound is *meanClock* + *K* \* *stdDevClock* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  The scale factor is calculated using:  Range is 0.025-55 m. |
| ***orbitClockErrorMeanScaleFactor***  This field specifies the Satellite Orbit and Clock Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockErrorMeanShapeVector* to restore the full values of the mean in the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClock* = *orbitClockErrorMeanShapeVector*[3] \* *orbitClockErrorMeanScaleFactor* * *stdDevClock* = *sqrt*(*orbitClockErrorCovarianceShapeMatrix* [9] \* *orbitClockErrorCovarianceScaleFactor)*   The bound is *meanClock* + *K* \* *stdDevClock* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  The scale factor is calculated using:  Range is 0.025-55 m. |
| ***orbitClockRateErrorCovScaleFactor***  This field specifies the Satellite Orbit and Clock Rate Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockRateErrorCovarianceShapeMatrix* to restore the full values of the covariance in the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClockRate* = *orbitClockRateErrorMeanShapeVector*[3] \* *orbitClockRateErrorMeanScaleFactor* * *stdDevClockRate* = *sqrt*(*orbitClockRateErrorCovarianceShapeMatrix* [9] \* *orbitClockRateErrorCovarianceScaleFactor)*   The bound is *meanClockRate* + *K* \* *stdDevClockRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.001 m/s; range 0.001-0.255 m/s. |
| ***orbitClockRateErrorMeanScaleFactor***  This field specifies the Satellite Orbit and Clock Rate Residual Error Bounds Scale Factor which is the scale factor to apply to *orbitClockRateErrorCovarianceShapeMatrix* and *orbitClockRateErrorBiasVector* to restore the full values of the paired overbounding model parameters.  For example, to calculate the clock error bound:   * *meanClockRate* = *orbitClockRateErrorMeanShapeVector*[3] \* *orbitClockRateErrorMeanScaleFactor* * *stdDevClockRate* = *sqrt*(*orbitClockRateErrorCovarianceShapeMatrix* [9] \* *orbitClockRateErrorCovarianceScaleFactor)*   The bound is *meanClockRate* + *K* \* *stdDevClockRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.001 m/s; range 0.001-0.255 m/s. |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Agree with Qualcomm and ESA |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think the proposals are going in the right direction, but would like more time to study the details. Our answer is, otherwise, similar to that for questions 2-2 and 2-3. |
| MELCO | Partially Yes | orbitClockErrorMeanShapeVector, orbitClockErrorCovarianceShapeMatrix, orbitClockErrorScaleFactor are natural extension of parameters used in an well-described algorithm as ARAIM. There should be no problem to standardize them once how to map to those of well-described algorithm is explained.  epochTime, iod-ssr, validityPeriodSeconds, validityPeriodDays, svID are supplementary parameters to let user to use above parameters consistently. There should be no problem to standardize them together with the above parameters.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-4 Summary

TBD

### Ionosphere and its error bounding parameters

In [5], the parameters for Ionosphere are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-**IonosphereParameters*

The IE *GNSS-Integrity-IonosphereParameters* is used by the location server to provide low update rate integrity parameters related to ionosphere. Bounding parameters are not included in this message but in the IE *GNSS-Integrity-IonosphereErrorBounds.*

-- ASN1START

GNSS-Integrity-IonosphereParameters-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

validityPeriod-r17 CHOICE {

validityPeriodSeconds-r17 INTEGER (1..86400),

validityPeriodDays-r17 INTEGER (1..365)

},

pIonosphereFault-r17 INTEGER (0..255),

tIonosphereFault-r17 INTEGER (1..256),

tCorrelationIonosphere-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

tCorrelationIonosphereRate-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

…

}

-- ASN1STOP

|  |  |
| --- | --- |
| **Conditional presence** | **Explanation** |
| *seq* | The field may be present if the integrity service supports time-based estimation techniques such as Kalman filtering as the user must be provided with information about the time correlation of errors. |

|  |
| --- |
| ***GNSS-Integrity-IonosphereParameters* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***Iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***validityPeriodSeconds***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***validityPeriodDays***  This field specifies the Validity Duration in days. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*. A day is defined to be 86,400 seconds.  Scale factor 1 day; range 1-365 days. |
| ***pIonosphereFault***  This field specifies the Probability of Onset of Ionosphere Fault per Time Unit which is the probability of occurrence of ionosphere error to exceed the residual error bound for more than the Time to Alert (TTA).  This field specifies the onset probability that the residual range or range rate error exceeds a bound created using the minimum allowed inflation factor *Kmin*, and bounding parameters as *mean* + *Kmin* \* *stdDev* where *Kmin* = *normInv*(*irMaximum* / 2).  The probability is calculated by where *n* is the value of *pIonosphereFault* and the range is 10-10.2 to 1 per hour. |
| ***tIonosphereFault***  This field specifies the Mean Ionospheric Fault Duration which is the mean duration between when an ionospheric integrity violation occurs, and the user is alerted by the service through the DNU flags (or the integrity violation is over).  Scale factor 1 s; range 1-256 s. |
| ***tCorrelationIonosphere***  This field specifies the Ionosphere Range Error Correlation Time which is the upper bound of the correlation time of the ionosphere residual range error.  The time is calculated using:  Range is 1-28,200 s. |
| ***tCorreleationIonosphereRate***  This field specifies the Ionosphere Range Rate Error Correlation Time which is the upper bound of the correlation time of the ionosphere residual range rate error.  The time is calculated using:  Range is 1-28,200 s. |

==========================TEXT PROPOSAL END=================================

###### Question2-5: Do companies agree with the above text proposal for Ionosphere parameters?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the Ies according to the previous discussion results at first, so we can figure out if these Ies are required. Too early to define the stage3 Ies. |
| Swift Navigation | Yes | Further information is provided in our response to Question 2-2. Generally speaking, the *Integrity-IonosphereParameters* are necessary for the UE to determine if its internal assumptions on the integrity risks associated with the Ionospheric bounds are compatible with the assumptions that have been made when deriving these bounds at the Network (as indicated by the Residual Risk parameter). |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Agree with Qualcomm and CATT |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think the proposals are going in the right direction, but would like more time to study the details. Our answer is, otherwise, similar to that for questions 2-2 and 2-3. |
| MELCO | Partially Yes | These are parameters which can be used in an extensional algorithm from a well-described algorithm as ARAIM, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or disclosed. |

###### Question2-5 Summary

TBD

Then, the bounding parameters for Ionosphere error are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-IonosphereErrorBounds*

The IE *GNSS-Integrity-IonosphereErrorBounds* is used by the location server to provide integrity bounding parameters relating to the ionosphere and ionosphere rate residual errors after application of the SSR corrections.

-- ASN1START

GNSS-Integrity-IonosphereErrorBounds-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

correctionPointSetID-r17 INTEGER (0..16383),

validityPeriod-r17 INTEGER (1..86400),

gridList-r17 Integrity-SSR-IonosphereGridList-r17,

...

}

Integrity-IonosphereGridList-r17 ::= SEQUENCE (SIZE(1..64)) OF

Integrity-IonosphereGridElement-r17

Integrity-IonosphereGridElement-r17 ::= SEQUENCE {

satList-r17 Integrity-SSR-IonosphereSatList-r17,

...

}

Integrity-IonosphereSatList-r17 ::= SEQUENCE (SIZE(1..64)) OF

Integrity-IonosphereSatElement-r17

Integrity-IonosphereSatElement-r17 ::= SEQUENCE {

svID-r17 SV-ID,

meanIonosphere-r17 INTEGER (0..255),

stdDevIonosphere-r17 INTEGER (0..255),

meanIonosphereRate-r17 INTEGER (0..255),

stdDevIonosphereRate-r17 INTEGER (0..255),

...

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-IonosphereErrorBounds* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***correctionPointSetID***  This field provides the ID of the *GNSS-SSR-CorrectionPoints* set. The *GNSS-Integrity-IonosphereErrorBounds* are valid for the correction points provided in IE *GNSS-SSR-CorrectionPoints* with the same *correctionPointSetID.* |
| ***validityPeriod***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***gridList***  This field provides the ionosphere error bounds for up to 64 correction points defined in IE *GNSS-SSR-CorrectionPoints*.  If the IE *GNSS-SSR-CorrectionPoints,* which belongs to the *correctionPointSetID*, includes the *listOfCorrectionPoints*, the *gridList* includes the same number of entries, and listed in the same order, as in the *listOfCorrectionPoints.*  If the IE *GNSS-SSR-CorrectionPoints,* which belongs to this *correctionPointSetID*, includes the *arrayOfCorrectionPoints* the *gridList* includes the same number of entries, and listed in the same order, as defined by the enabled bits in the *bitmaskOfGrids*. |
| ***svID***  This field specifies the satellite for which ionosphere error bounds are provided. |
| ***meanIonosphere***  This field specifies the Ionosphere Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual ionosphere error.  The bound is *meanIonosphere* + *K* \* *stdDevIonosphere* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  The mean is calculated using:  Range is 0-17.5 m. |
| ***stdDevIonosphere***  This field specifies the Ionosphere Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual ionosphere error.  The standard deviation is calculated using:  Range is 0-17.5 m. |
| ***meanIonosphereRate***  This field specifies the Ionosphere Rate Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual ionosphere rate error.  The bound is *meanIonosphereRate* + *K* \* *stdDevIonosphereRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***stdDevIonosphereRate***  This field specifies the Ionosphere Rate Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual ionosphere rate error.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |

==========================TEXT PROPOSAL END=================================

###### Question2-6: Do companies agree with the above text proposal for the bounding parameters for Ionosphere error?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the IEs according to the previous discussion results at first, so we can figure out if these IEs are required. Too early to define the stage3 IEs. |
| Swift Navigation | Yes | Further information is provided in our response to Question 2-2. Generally speaking, the *Integrity-IonosphereErrorBounds* are used to statistically bound the residual Ionospheric errors after the positioning corrections (e.g. RTK, SSR) have been applied. We can add these descriptions to the Stage 2 specifications (TS 38.305). |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Agree with Qualcomm and CATT |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think the proposals are going in the right direction, but would like more time to study the details. Our answer is, otherwise, similar to that for questions 2-2 and 2-3. |
| MELCO | Partially Yes | meanIonosphere and stdDevIonosphere are natural extension of parameters used in an well-described algorithm as ARAIM. There should be no problem to standardize them once how to map to those of well-described algorithm is explained.  epochTime, iod-ssr, correctionPointSetID, validityPeriod, gridList, svID are supplementary parameters to let user to use above parameters consistently. There should be no problem to standardize them together with the above parameters.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-6 Summary

TBD

### Troposphere and its error bounding parameters

In [5], the parameters for Troposphere are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-**TroposphereParameters*

The IE *GNSS-Integrity-TroposphereParameters* is used by the location server to provide low update rate integrity parameters related to troposphere. Bounding parameters are not included in this message but in the IE *GNSS-Integrity-TroposphereErrorBounds.*

-- ASN1START

GNSS-Integrity-TroposphereParameters-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

validityPeriod-r17 CHOICE {

validityPeriodSeconds-r17 INTEGER (1..86400),

validityPeriodDays-r17 INTEGER (1..365)

},

pTroposphereFault-r17 INTEGER (0..255),

tTroposphereFault-r17 INTEGER (1..256),

tCorrelationTroposphere-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

tCorrelationTroposphereRate-r17 INTEGER (1..255) OPTIONAL, -- Cond seq

...

}

-- ASN1STOP

|  |  |
| --- | --- |
| **Conditional presence** | **Explanation** |
| *seq* | The field may be present if the integrity service supports time-based estimation techniques such as Kalman filtering as the user must be provided with information about the time correlation of errors. |

|  |
| --- |
| ***GNSS-Integrity-TroposphereParameters* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***validityPeriodSeconds***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*.  Scale factor 1 s; range 1-86,400 s. |
| ***validityPeriodDays***  This field specifies the Validity Duration in days. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*. A day is defined to be 86,400 seconds.  Scale factor 1 day; range 1-365 days. |
| ***pTroposphereFault***  This field specifies the Probability of Onset of Troposphere Fault per Time Unit which is the probability of occurrence of troposphere error to exceed the residual error bound for more than the Time to Alert (TTA).This field specifies the onset probability that the residual range or range rate error exceeds a bound created using the minimum allowed inflation factor *Kmin*, and bounding parameters as *mean* + *Kmin* \**stdDev* where *Kmin* = *normInv*(*irMaximum* / 2).  The probability is calculated by where *n* is the value of *pTroposphereFault* and the range is 10-10.2 to 1 per hour. |
| ***tTroposphereFault***  This field specifies the Mean Troposphere Fault Duration which is the mean duration between when a troposphere integrity violation occurs, and the user is alerted by the service through the DNU flags (or the integrity violation is over).  Scale factor 1 s; range 1-256 s. |
| ***tCorrelationTroposphere***  This field specifies the Troposphere Range Error Correlation Time which is the upper bound of the correlation time of the troposphere residual range error.  The time is calculated using:  Range is 1-28,200 s. |
| ***tCorreleationTroposphereRate***  This field specifies the Troposphere Range Rate Error Correlation Time which is the upper bound of the correlation time of the troposphere residual range rate error.  The time is calculated using:  Range is 1-28,200 s. |

==========================TEXT PROPOSAL END=================================

Companies are invited to review the above the text proposal for troposphere parameters and answer the following question:

###### Question2-7: Do companies agree with the above text proposal for the Troposphere parameters?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the IEs according to the previous discussion results at first, so we can figure out if these IEs are required. Too early to define the stage3 IEs. |
| Swift Navigation | Yes | Further information is provided in our response to Question 2-2. Generally speaking, the *Integrity-TroposphereParameters* are necessary for the UE to determine if its internal assumptions on the integrity risks associated with the Tropospheric bounds are compatible with the assumptions that have been made when deriving these bounds at the Network (as indicated by the Residual Risk parameter). |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Agree with Qualcomm and CATT |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think the proposals are going in the right direction, but would like more time to study the details. Our answer is, otherwise, similar to that for questions 2-2 and 2-3. |
| MELCO | Partially Yes | These are parameters which can be used in an extensional algorithm from a well-described algorithm as ARAIM, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or disclosed. |

###### Question2-7 Summary

TBD

Then, the bounding parameters for bias are provided under the following text proposal

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-TroposphereErrorBounds*

The IE *GNSS-Integrity-TroposphereErrorBounds* is used by the location server to provide integrity bounding parameters relating to the troposphere and troposphere rate residual errors after application of the SSR corrections.

-- ASN1START

GNSS-Integrity-TroposphereErrorBounds-r17 ::= SEQUENCE {

epochTime-r17 GNSS-SystemTime,

iod-ssr-r17 INTEGER (0..15),

correctionPointSetID-r17 INTEGER (0..16383),

validityPeriod-r17 INTEGER (1..86400),

gridList-r17 Integrity-SSR-TroposphereGridList-r17,

...

}

Integrity-TroposphereGridList-r17 ::= SEQUENCE (SIZE(1..64)) OF

Integrity-TroposphereGridElement-r17

Integrity-TroposphereGridElement-r17 ::= SEQUENCE {

meanTroposphereVerticalHydroStaticDelay-r17 INTEGER (0..255),

stdDevTroposphereVerticalHydroStaticDelay-r17 INTEGER (0..255),

meanTroposphereVerticalWetDelay-r17 INTEGER (0..255),

stdDevTroposphereVerticalWetDelay-r17 INTEGER (0..255),

meanTroposphereVerticalHydroStaticDelayRate-r17 INTEGER (0..255),

stdDevTroposphereVerticalHydroStaticDelayRate-r17 INTEGER (0..255),

meanTroposphereVerticalWetDelayRate-r17 INTEGER (0..255),

stdDevTroposphereVerticalWetDelayRate-r17 INTEGER (0..255),

...

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-TroposphereErrorBounds* field descriptions** |
| ***epochTime***  This field specifies the epoch time of the constellation integrity values. The *gnss-TimeID* in *GNSS-SystemTime* shall be the same as the *GNSS-ID* in IE *GNSS-GenericAssistDataElement*. |
| ***iod-ssr***  This field specifies the Issue of Data number for the SSR data the integrity values are applicable to. |
| ***correctionPointSetID***  This field provides the ID of the *GNSS-SSR-CorrectionPoints* set. The *GNSS-Integrity-TroposphereErrorBounds* are valid for the correction points provided in IE *GNSS-SSR-CorrectionPoints* with the same *correctionPointSetID.* |
| ***validityPeriod***  This field specifies the Validity Duration in seconds. The integrity values are only valid for the time interval from *epochTime* to *epochTime* + *validityPeriod*. Scale factor 1 s; range 1-86,400 s. |
| ***gridList***  This field provides the troposphere error bounds to the hydro static and wet vertical components prior to the application of the mapping function, for up to 64 correction points defined in IE *GNSS-SSR-CorrectionPoints*.  If the IE *GNSS-SSR-CorrectionPoints,* which belongs to the *correctionPointSetID*, includes the *listOfCorrectionPoints*, the *gridList* includes the same number of entries, and listed in the same order, as in the *listOfCorrectionPoints.*  If the IE *GNSS-SSR-CorrectionPoints,* which belongs to this *correctionPointSetID*, includes the *arrayOfCorrectionPoints* the *gridList* includes the same number of entries, and listed in the same order, as defined by the enabled bits in the *bitmaskOfGrids*. |
| ***meanTroposphereVerticalHydroStaticDelay***  This field specifies the Vertical Hydro Static Troposphere Delay Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual troposphere error in the vertical hydro static delay component.  The bound is *meanTroposphereVerticalHydroStaticDelay* + *K* \* *stdDevTroposphereVerticalHydroStaticDelay* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.005 m; range 0-1.275 m. |
| ***stdDevTroposphereVerticalHydroStaticDelay***  This field specifies the Vertical Hydro Static Troposphere Delay Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual troposphere error in the vertical hydro static delay component.  Scale factor 0.005 m; range 0-1.275 m. |
| ***meanTroposphereVerticalWetDelay***  This field specifies the Vertical Wet Troposphere Delay Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual troposphere error in the vertical wet delay component.  The bound is *meanTroposphereVerticalWetDelay* + *K* \* *stdDevTroposphereVerticalWetDelay* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.005 m; range 0-1.275 m. |
| ***stdDevTroposphereVerticalWetDelay***  This field specifies the Vertical Wet Troposphere Delay Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual troposphere error in the vertical wet delay component.  Scale factor 0.005 m; range 0-1.275 m. |
| ***meanTroposphereVerticalHydroStaticDelayRate***  This field specifies the Vertical Hydro Static Troposphere Delay Rate Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual troposphere rate error in the vertical hydro static delay component.  The bound is *meanTroposphereVerticalHydroStaticDelayRate* + *K* \* *stdDevTroposphereVerticalHydroStaticDelayRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***stdDevTroposphereVerticalHydroStaticDelayRate***  This field specifies the Vertical Hydro Static Troposphere Delay Rate Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual troposphere rate error in the vertical hydro static delay component.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***meanTroposphereVerticalWetDelayRate***  This field specifies the Vertical Wet Troposphere Delay Rate Error Bound Mean which is the mean value for a paired overbounding model that bounds the residual troposphere rate error in the vertical wet delay component.  The bound is *meanTroposphereVerticalWetDelayRate* + *K* \* *stdDevTroposphereVerticalWetDelayRate* and shall be so that the probability of it to be exceeded shall be lower than *IRallocation* for *irMinimum* < *IRallocation* < *irMaximum*., where *K* = *normInv*(*IRallocation* / 2).  This *IRallocation* is a fraction of the Target Integrity Risk that represents the integrity risk budget available.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |
| ***stdDevTroposphereVerticalWetDelayRate***  This field specifies the Vertical Wet Troposphere Delay Rate Error Bound Standard Deviation which is the standard deviation for a paired overbounding model that bounds the residual troposphere rate error in the vertical wet delay component.  Scale factor 0.00005 m/s; range 0-0.01275 m/s. |

==========================TEXT PROPOSAL END=================================

Companies are invited to review the above the text proposal for troposphere parameters and answer the following question:

###### Question2-8: Do companies agree with the above text proposal for the bounding parameters for Troposphere error?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Not sure | Not sure how the value range is defined; |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? How have the value ranges etc. be determined? What should a UE generally do with all this data? We should agree first on what assistance data are strictly needed (and why) to determine integrity of GNSS before jumping into encoding details. |
| CATT | Not sure | Please classify and specify the IEs according to the previous discussion results at first, so we can figure out if these IEs are required. Too early to define the stage3 IEs. |
| Swift Navigation | Yes | Further information is provided in our response to Question 2-2. Generally speaking, the *Integrity-TroposphereErrorBounds* are used to statistically bound the residual Tropospheric errors after the positioning corrections (e.g. RTK, SSR) have been applied. We can add these descriptions to the Stage 2 specifications (TS 38.305). |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Agree with Qualcomm and CATT |
| ZTE |  | Agree with QC and CATT |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters and values. Too early to jump to stage-3 coding. |
| u-blox | Not yet | We think the proposals are going in the right direction, but would like more time to study the details. Our answer is, otherwise, similar to that for questions 2-2 and 2-3. |
| MELCO | Partially Yes | meanTroposphereVerticalWetDelay and stdDevTroposphereVerticalWetDelay are natural extension of parameters used in an well-described algorithm as ARAIM. There should be no problem to standardize them once how to map to those of well-described algorithm is explained.  epochTime, iod-ssr, correctionPointSetID, validityPeriod, gridList, svID are supplementary parameters to let user to use above parameters consistently. There should be no problem to standardize them together with the above parameters.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-8 Summary

TBD

### Periodic assistance data

In [5], the following text proposal has been proposed for periodic assistance data

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-PeriodicAssistData*

The IE *GNSS-PeriodicAssistData* is used by the location server to provide control parameters for a periodic assistance data delivery session (e.g., interval and duration) to the target device.

NOTE: Omission of a particular assistance data type field in IE *GNSS-PeriodicAssistData* means that the location server does not provide this assistance data type in a data transaction of a periodic assistance data delivery session, as described in clauses 5.2.1a and 5.2.2a. Inclusion of no assistance data type fields in IE *GNSS-PeriodicAssistData* means that a periodic assistance data delivery session is terminated.

-- ASN1START

GNSS-PeriodicAssistData-r15 ::= SEQUENCE {

gnss-RTK-PeriodicObservations-r15 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

glo-RTK-PeriodicBiasInformation-r15 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-RTK-MAC-PeriodicCorrectionDifferences-r15

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-RTK-PeriodicResiduals-r15 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-RTK-FKP-PeriodicGradients-r15 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicOrbitCorrections-r15

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicClockCorrections-r15

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicCodeBias-r15 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

...,

[[

gnss-SSR-PeriodicURA-r16 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicPhaseBias-r16 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicSTEC-Correction-r16 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-SSR-PeriodicGriddedCorrection-r16 GNSS-PeriodicControlParam-r15 OPTIONAL -- Need ON

]] ,

[[

gnss-Integrity-PeriodicServiceAlert-r17 GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicTroposphereErrorBounds-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicConstellationAlert-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicConstellationParameters-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicBiasErrorBounds-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicOrbitClockErrorBounds-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicIonosphereParameters-r17

GNSS-PeriodicControlParam-r15 OPTIONAL, -- Need ON

gnss-Integrity-PeriodicIonosphereErrorBounds-r17

GNSS-PeriodicControlParam-r15 OPTIONAL -- Need ON

]]

}

-- ASN1STOP

==========================TEXT PROPOSAL END=================================

We would like the companies to check on the above text proposal and verify if it is correct.

###### Question2-9: Do companies agree with the above text proposal for the periodic assistance data?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Yes |  |
| Qualcomm |  | See our response to Question 2-1. This follows automatically once the assistance data have been agreed. |
| CATT |  | Why is gnss-Integrity-PeriodicServiceAlert sent to UE periodically? Will the service alert be changed periodically? |
| Swift Navigation | Yes | Subject to whether we consolidate the proposed integrity messages with some of the existing SSR messages. Service Alert will be sent periodically because its specific purpose is to alert the user in the case of a service issue that impacts integrity. We anticipate this needs to be sent at a reasonably high rate given it directly impacts the TTA. |
| Nokia |  | We don’t have any agreement about this yet, so we should not agree any stage-3 details |
| ZTE |  | We should firstly discuss whether all these IEs can be periodic, then decide ASN.1 |
| vivo |  | There are no explicit agreements for the stage-3 parameters. Too early to jump to stage-3 coding. |
| u-blox |  | See our answer to question 2-1. There would seem to be scope for consolidation and/or cobining the integrity with the corrections in some cases. |
| MELCO | Yes | We agree that assistance data are provided periodically once they have been agreed. |

###### Question2-9 Summary

TBD

## GNSS integrity service parameter assistance data

In [5], the following text proposals has been proposed for assistance data for service. But also, in this IE *GNSS-Integrity-ServiceAlert*, the Ionosphere DNU and Troposphere DNU are also included.

==========================TEXT PROPOSAL BEGIN=================================

#### *– GNSS-Integrity-ServiceParameters*

The IE *GNSS-Integrity-ServiceParameters* is used by the location server to provide the range of Integrity Risk (IR) that can be met when using integrity data provided by the service in the other *GNSS-Integrity* IEs. The range shall not change during a session.

-- ASN1START

GNSS-Integrity-ServiceParameters-r17 ::= SEQUENCE {

irMinimum-r17 INTEGER (0..255),

irMaximum-r17 INTEGER (0..255),

...

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-ServiceParameters* field descriptions** |
| ***irMinimum***  This field specifies the Minimum Integrity Risk (IR) which is the minimum IR for which the error bounds provided in the other *GNSS-Integrity* IEs is valid.  The IR is calculated by where *n* is the value of *irMinimum* and the range is 10-10.2 to 1. |
| ***irMaximum***  This field specifies the Maximum Integrity Risk (IR) which is the maximum IR for which the error bounds provided in the other *GNSS-Integrity* IEs is valid.  The IR is calculated by where *n* is the value of *irMaximum* and the range is 10-10.2 to 1. |

#### *– GNSS-Integrity-ServiceAlert*

The IE *GNSS-Integrity-ServiceAlert* is used by the location server to indicate whether the service can be used for integrity related applications.

-- ASN1START

GNSS-Integrity-ServiceAlert-r17 ::= SEQUENCE {

serviceDoNotUse-r17 BOOLEAN,

ionosphereDoNotUse-r17 BOOLEAN,

troposphereDoNotUse-r17 BOOLEAN,

...

}

-- ASN1STOP

|  |
| --- |
| ***GNSS-Integrity-ServiceAlert* field descriptions** |
| ***serviceDoNotUse***  This field specifies the Service DNU Flag which indicates whether the service can be used for integrity related applications (FALSE) or not (TRUE). Shall be set to Do Not Use (DNU) during a service testing phase or in case the service is unsafe. |
| ***ionosphereDoNotUse***  This field specifies the Ionosphere DNU Flag which indicates whether the ionospheric products can be used for integrity related applications (FALSE) or not (TRUE). |
| ***troposphereDoNotUse***  This field specifies the Troposphere DNU Flag which indicates whether the tropospheric products can be used for integrity related applications (FALSE) or not (TRUE). |

==========================TEXT PROPOSAL END================================

In the above text proposal, assistance data has been given for TIR and alert of DNU. We would like the companies to check on the above text proposal and verify if it is correct.

###### Question2-10: Do companies agree with the above text proposal for the assistance data of GNSS integrity service?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Yes |  |
| Qualcomm | Not yet. | There is no background, justification, etc. provided. Why is all this needed? |
| CATT | No | The service for UE-based integrity calculation should not be sent to UE via assistance data.  These IEs should be optional. |
| Swift Navigation | Yes | The Alerts should be sent in the assistance data because they are dynamic; these are the actual alerts provided to the user that flag the possibility of a feared event. |
| ESA | Not before confirmation of alignment with RTCM | Same answer as for 2-3 |
| Nokia |  | Further discussion is needed, we cannot agree stage-3 details for now |
| ZTE |  | Agree to coordinate with RTCM first before determining stage-3 details |
| vivo | Not sure | There are no explicit agreements for the stage-3 parameters. Too early to jump to stage-3 coding. |
| u-blox | Not sure | It is not clear why these are needed. Surely if the integrity service delivers integrity parameters they are trustworthy? |
| MELCO | Partially Yes | serviceDoNotUse is a parameter well-known in GNSS community and used in an well-described algorithm as ARAIM. There should be no problem to standardize them.  ionosphereDoNotUse and troposphereDoNotUse are natural extension of parameters used in an well-described algorithm as ARAIM. There should be no problem to standardize them once how to map to those of well-described algorithm is explained.  The others are parameters which can be used in an extensional algorithm, and people might not be very familiar with them. It would help people to proceed to standardize them if such algorithm is available in a reference or is disclosed. |

###### Question2-10 Summary:

TBD

In the phase I of the discussion, Franuhofer also mentioned that svDoNotUseFlag should also be added.

badSignalID GNSS-SignalIDs OPTIONAL, -- Need OP

However, for the current spec, this has already been included under the *GNSS-RealTimeIntegrity* IE.

– *GNSS-RealTimeIntegrity*

The IE *GNSS-RealTimeIntegrity* is used by the location server to provide parameters that describe the real-time status of the GNSS constellations. *GNSS-RealTimeIntegrity* data communicates the health of the GNSS signals to the mobile in real‑time.

The location server shall always transmit the *GNSS-RealTimeIntegrity* with the current list of unhealthy signals (i.e., not only for signals/SVs currently visible at the reference location), for any GNSS positioning attempt and whenever GNSS assistance data are sent. If the number of bad signals is zero, then the *GNSS-RealTimeIntegrity* IE shall be omitted.

-- ASN1START

GNSS-RealTimeIntegrity ::= SEQUENCE {

gnss-BadSignalList GNSS-BadSignalList,

…

}

GNSS-BadSignalList ::= SEQUENCE (SIZE(1..64)) OF BadSignalElement

BadSignalElement ::= SEQUENCE {

badSVID SV-ID,

badSignalID GNSS-SignalIDs OPTIONAL, -- Need OP

…

}

-- ASN1STOP

| ***GNSS-RealTimeIntegrity* field descriptions** |
| --- |
| ***gnss-BadSignalList***  This field specifies a list of satellites with bad signal or signals. |
| ***badSVID***  This field specifies the GNSS *SV‑ID* of the satellite with bad signal or signals. |
| ***badSignalID***  This field identifies the bad signal or signals of a satellite. This is represented by a bit string in *GNSS-SignalIDs*, with a one‑value at a bit position means the particular GNSS signal type of the SV is unhealthy; a zero‑value means healthy. Absence of this field means that all signals on the specific SV are bad. |

###### Question2-11: Do companies agree with the above assistance data can be reused for GNSS integrity in R17?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
| Intel | Yes |  |
| Qualcomm | Yes |  |
| CATT | Yes |  |
| Swift Navigation | Yes, with comments | It should be possible that this IE can be reused but please note that the intention of the TP in [5] was to use the proposed *GNSS-Integrity-ConstellationAlert-r17*, which looks to have been missed in the text descriptions presented in Phase 2? The existing *GNSS-RealTimeIntegrity* IE does not provide the ability to flag DNU at the constellation level, but this is probably acceptable. If the existing IE is reused we may wish to update the descriptions to align with the “Do Not Use” terminology. |
| ESA | Yes | It provides basic integrity support – flag unhealthy satellites and signals |
| Nokia | Yes |  |
| ZTE | Yes |  |
| vivo | Yes |  |
| u-blox |  | It is not clear to us what a “bad signal” is. A proper understanding of this is needed before it is meaningful in an integrity context. |
| MELCO | Yes |  |

###### Question2-11 Summary:

TBD

## Relation with RTCM

In RAN2#114, a LS to RTCM has been sent [8]. The LS seeks coordinated work between 3GPP and RTCM for the work on GNSS integrity. However, up until now, there is no reply LS received.

In [6], the relationship between the on-going discussion on GNSS integrity in 3GPP and that in RTCM is discussed. [6] mentioned that the proposed TP includes some parameters that are currently not included in the RTCM standard, but for which [6] thinks that are useful. [6] also thinks a useful next step is that RAN2 can propose to RTCM that they consider adopting the baseline SSR correction messages already supported in LPP, and also proposed to wait for the reply LS from RTCM, given that we have sent the LS in RAN2#113

[4] also mentioned about how to handle the parameters by RTCM. They argue that if new parameters are introduced by RTCM, they should be added to 3GPP spec in the future releases.

In the first phase of the email discussion, companies also mention that we should wait for the decisions from RTCM before we jump to conclusions in 3GPP.

Companies are welcomed to provide inputs to the following open questions:

###### Question2-12: Regarding the discussion on GNSS integrity in RTCM

* + ***What is the status/progress/timeline of the discussion for GNSS integrity in RTCM?***
  + ***How to coordinate the discussion on assistance data for GNSS integrity in LPP with the RTCM?***

|  |  |
| --- | --- |
| **Company** | **Comments** |
| Intel | Based on their LS. Seems they can only finish their work in the middle of next year which seems late to us. |
| Qualcomm | Draft RTCM specs may be available much earlier than a final/published spec, since RTCM normally publishes a spec only after some successful IOT (which takes the time). Any draft spec (if provided by RTCM) could be used for general alignment. Details may change (e.g., in the encoding, value ranges, etc.) but this can be corrected via normal CR process later. However, the important issue at this stage is that the general directions are aligned. As commented above, all the proposed details in this email discussion came “out of the blue” without any supporting background information.  These details should have been the main scope of the study!  If RTCM does not provide a draft spec to 3GPP, we should go for the minimum in this Release (as e.g., in [3]) and add more “advanced concepts” in Rel-18 based on published RTCM spec. |
| CATT | Prefer to wait for the draft spec from RTCM and try to align with RTCM as much as possible in Rel-17. |
| Swift Navigation | **Status:** RTCM SC-134 is targeting a first release in mid-2022 but only for OSR (see [R2-2106105](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_114-e/Docs/R2-2106105.zip) and the LS). They are also analysing SSR Integrity for future releases but this is contingent on a full SSR positioning standard (PPP-RTK) being adopted in the parallel RTCM SC-104 committee, which has not been decided or defined (and will not be within the R17 timeline). If RTCM does decide to adopt SSR, we strongly encourage that the chosen format be consistent with LPP SSR to ensure there is common alignment between the two protocols, and therefore a common baseline for the Integrity extensions. Indeed, we agree with QC that it is important that the general directions are aligned, even if the details are not yet confirmed in RTCM, given these details can be updated later in LPP via normal CR processes. We are confident the RTCM and 3GPP integrity directions are already well aligned, as shown in [R2-2106105](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_114-e/Docs/R2-2106105.zip), and are happy to check with RTCM on this point.  **Coordination:** We encourage open sharing and alignment between 3GPP and RTCM. For expediency, the package of text proposals being discussed in RAN2 could be checked with RTCM for comment. We should also suggest that RTCM adopt the LPP SSR positioning messages to establish a common baseline for SSR and Integrity.  In response to QC, the Study provided a comprehensive introduction to the concepts of GNSS integrity and the associated error sources that must be bound in order to mitigate the probability of Integrity Events. But it was explicitly noted and agreed in Table 9.4.1.1 that:  NOTE: The positioning integrity assistance information Ies are FFS as part of the WI. |
| ESA | RTCM sent to us a reply LS on the 3rd of September addressing a number of points (<https://www.3gpp.org/Liaisons/Incoming_LSs/R2-meeting.htm>). Furthermore, RTCM suggested that more feedback will be sent to RAN2 after the RTCM meeting that took place at the end of September. This additional feedback has not been received yet. The LS can be found at:  In preparation for next meeting and at the guidance of the NR Pos chairman, ESA is preparing a discussion paper on this LS.  In short, from RTCM document we realised:   * RTCM SC134 work is focused on both SSR and OSR * A first standard for integrity is expected to be published in Q2 2022.   Furthermore, at the last International Committee on GNSS (September 2021), RTCM SC134 gave a presentation on its work on integrity. RAN2 delegates could look at this presentation as complementary to the LS sent to us: we can find also a table of content of the draft RTCM specs for integrity and some high-level details on the messages under consideration: <https://www.unoosa.org/documents/pdf/icg/2021/ICG15/29.pdf>  We agree with Swift that in absence of any other information until this point, we should attach the current draft TP to a new LS to RTCM and ask for comments. Lastly, if no feedback is received by end of the Work Item, we support QC proposal to defer the details on integrity to Release 18 and make use of RTCM specs (which are expected to be published in Q2 2022) – maybe could be done via TEI once RTCM specs are available without the need for a new WI?! |
| Nokia | Basically we share the view as Qualcomm, CATT and ESA. If no draft spec. is received from RTCM during Rel-17 time frame, we don’t think this is a serious problem to keep Rel-17 at a basic level. At least 3GPP has already adopted some fundamental mechanisms such as KPI transferring and integrity result reporting in LPP, we can already claim Rel-17 support GNSS positioning integrity to a certain extent. We can always enhance it for more advanced use cases in later releases. |
| ZTE | Agree with ESA that we send LS with TPs first, if there is no feedback, details can be postponed to the later releases. |
| vivo | Agree with Nokia. |
| u-blox | We would prefer the work not to be held up waiting for RTCM, although we are in favoor of the best possible alignment with them and would support sharing our work with RTCM to help expedite the standardization process on both sides. In this respect we agree with Swift’s proposal.  However if agreement is not possible at this stage we would be able to support Nokia’s and Qualcomm’s proposals for a basic integrity profile provided that there is an ongoing effort to standardize the advanced integrity implementation in Rel-18 with full RTCM alignment. |
| MELCO | It is not always necessary to wait the publication from RTCM. Depending on parameters we can proceed to standardize part of them as they are well-known in GNSS community and used in a well-described algorithm as ARAIM.  For further alignment between 3GPP and RTCM, consistency between SSR and OSR is important. For the full consistency, “Local environment feared events” needs to be discussed in the near future as RTCM would consider those event in OSR. |

###### Question2-12 Summary

TBD

Conclusions of Phase II

TBD

# References

1. TR 38.857, Study on NR Positioning Enhancements (Release 17), V17.0.0.
2. R2-2109029, Summary on agenda item 8.11.5 on GNSS positioning integrity, Qualcomm.
3. R2-2108340, "Bounding GNSS errors for positioning integrity", ESA, Nokia, Nokia Shanghai Bell.
4. R2-2108385, "Considerations on GNSS positioning integrity support", Qualcomm Incorporated.
5. R2-2108475, "Text Proposal on GNSS Integrity Assistance Data", Swift Navigation, Ericsson, Mitsubishi Electric Corporation.
6. R2-2108474, "Discussion on GNSS Integrity Assistance Data", Swift Navigation, Ericsson, Mitsubishi Electric Corporation.
7. R2-2108396, " GNSS positioning integrity ", Ericsson
8. R2-2106596 LS to RTCM on GNSS integrity assistance data ESA LS out To:RTCM SC134 Cc: RTCM, RTCM SC104