**3GPP TSG-RAN WG2 Meeting #118 electronic *R2-2206506***

**Online, May 9 – 20th, 2022**

**Agenda Item: 6.10.1.1**

**Source: Thales**

**Title: Summary of [AT118-e][119][NTN] Coarse UE location info (Thales)**

**Document for: Discussion and Decision**

# Introduction

This document aims to summarize the following discussion.

**[AT118-e][119][NTN] Coarse UE location info (Thales)**

Scope: Discuss the coarse UE location information format (e.g. whether to use the most significant bits of GNSS longitude/latitude or also consider the proposals in [R2-2205574](file:///C:\Data\3GPP\Extracts\R2-2205574%20Coarse%20location%20format.docx)) and the reporting mechanism (e.g. reuse the mechanism for reporting commonLocationInfo)

Intended outcome: Summary of the offline discussion with list of proposals

Deadline (for companies' feedback):  Thursday 2022-05-19 08:00 UTC

Deadline (for rapporteur's summary in R2-2206506):  Thursday 2022-05-19 10:00 UTC

Proposals marked "for agreement" in R2-2206506 not challenged until Thursday 2022-05-19 20:00 UTC will be declared as agreed via email by the session chair.

This offline discussion aims at defining the coarse location format as outcome of RAN2#118-e.

# Discussion – 1st round

The following proposals on the coarse location format in [1] are submitted to the discussion.

## 2.1 Which definition to start from ?

Proposal 1 When defining a coarse UE location representation format, use the definition of EllipsoidPoinWithAltitude in TS 37.355 and round the coordinates to fewer bits to achieve a suitable accuracy.

**Question 4.5.1: Do companies agree with the proposal above ?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree/Disagree** | **Comments/Suggestions** |
| Xiaomi | Disagree | 37.355 includes the following formats, and all of them(except high accuracy ellipsoid point) include degreeLatitude/latitudeSign and degreeLongitude, which are the items useful for NTN coarse location report:  LocationCoordinates ::= CHOICE {  ellipsoidPoint Ellipsoid-Point,  ellipsoidPointWithUncertaintyCircle Ellipsoid-PointWithUncertaintyCircle,  ellipsoidPointWithUncertaintyEllipse EllipsoidPointWithUncertaintyEllipse,  polygon Polygon,  ellipsoidPointWithAltitude EllipsoidPointWithAltitude,  ellipsoidPointWithAltitudeAndUncertaintyEllipsoid  EllipsoidPointWithAltitudeAndUncertaintyEllipsoid,  ellipsoidArc EllipsoidArc,  ...,  highAccuracyEllipsoidPointWithUncertaintyEllipse-v1510  HighAccuracyEllipsoidPointWithUncertaintyEllipse-r15,  highAccuracyEllipsoidPointWithAltitudeAndUncertaintyEllipsoid-v1510  HighAccuracyEllipsoidPointWithAltitudeAndUncertaintyEllipsoid-r15  }  We prefer that UE can choose any format except high accuracy ellipsoid point above it supports for coarse location report. |
| Ericsson | Agree | Using the *EllipsoidPointWithAltitude* IE as the baseline is a good choice. Then the coordinates should be rounded to fewer bits, where the number of bits depends on the latitude. And the altitude should be optional, and its presence configurable depending on the intended use of the reported coarse UE location. A possible alternative could be to use both the *EllipsoidPoint* IE (when the altitude is not reported) and the *EllipsoidPointWithAltitude* IE (when the altitude is reported) as the baseline. |
| Thales | Agree | We are fine with the proposal. Not very convinced about the added value of adding the altitude. |
| OPPO | Agree | Fine with start point. |
| Lenovo | Agree with comments | We wonder why altitude is necessary. |
| Nokia | Resolved online | But we think altitude may not be needed, so EllipsoidPoint IE is enough |

## 2.4 Altitude ?

Proposal 2 Make it configurable whether an indication of the altitude should be included in a report of a coarse UE location.

**Question 4.5.2: Do companies agree with the proposal above ?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree/Disagree** | **Comments/Suggestions** |
| Xiaomi | Disagree | For location report for SA2/RAN3 purposes, altitude is not needed. For RAN2 HO, altitude is not needed; For RAN2 SMTC adjustment, we agree to use propogation delay difference; For TA report, we have TA report MAC CE. Thus, we don’t see the need to report altitude. |
| Ericsson | Agree | Whether the altitude is needed depends on the intended use of the reported coarse UE location. If the coarse UE location is reported instead of the UE’s TA for the purpose of determining a suitable UE specific Koffset, the altitude measure is important. On the other hand, if the purpose of reporting the coarse UE location is to determine which country the UE is located in, the altitude indication is not needed. |
| Thales | Disagree | Same view as Xiaomi since the main intend is to determine which country the UE is located in |
| OPPO | Disagree | Agree with Xiaomi. |
| Lenovo | Disagree | Agree with Xiaomi that altitude is not needed. |
| Nokia | Disagree | We should simplify the whole thing, so for this purpose we suggest not to have too many choices. |

## 2.3 Compensation scheme for different latitudes

Proposal 3 Compensate for the different longitude density at different latitudes, so that a reported coarse UE location cannot be too accurate.

**Question 4.5.3: Do companies agree with the proposal above ?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree/Disagree** | **Comments/Suggestions** |
| Xiaomi | Agree | We agree that the closer to the pole, one degree of longitude would corresponds to smaller distance. To achieve same location accuracy, the required most significant bits need to be reduced. |
| Ericsson | Agree | The intention of the choice reporting a coarse UE location in the first place is that it is a compromise between accuracy and privacy, where the target accuracy should be roughly comparable to a terrestrial cell with 2 km radius. If such a compensation is for different longitude densities is not used, the accuracy of the reported location will differ greatly between reported locations close to the equator and reported locations close to the poles, and hence the coarse UE location reporting would not live up to the intended compromise. Therefore, compensation for different longitude densities is crucial to make the coarse UE location reporting fulfil its requirements.  Remember that we are not attempting to save bits here, but to make sure the reporting is sufficiently private. |
| Thales | Agree |  |
| OPPO | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree | Agree with Ericsson. |

## 2.4 Representation format of coarse UE location information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Latitude range in degrees (°)** | **Latitude range represented with 11 bits** | **Number of bits representing the longitude** | **Longitude range** | **Number of steps a “latitude circle” is divided into** |
| 0° - 59.978022° | 0 - 1364 | 13 | ±4095 | 8191 |
| 59.978022° - 75.538462° | 1365 - 1718 | 12 | ±2047 | 4095 |
| 75.538462° - 82.835165° | 1719 - 1884 | 11 | ±1023 | 2047 |
| 82.835165° - 86.439560° | 1885 - 1966 | 10 | ±511 | 1023 |
| 86.439560° - 88.197802° | 1967 - 2006 | 9 | ±255 | 511 |
| 88.197802° - 89.120879° | 2007 - 2027 | 8 | ±127 | 255 |
| 89.120879° - 89.560440° | 2028 - 2037 | 7 | ±63 | 127 |
| 89.560440° - 89.824176° | 2038 - 2043 | 6 | ±31 | 63 |
| 89.824176° - 89.912088° | 2044 - 2045 | 5 | ±15 | 31 |
| 89.912088° - 89.956044° | 2046 | 4 | ±7 | 15 |
| 89.956044° - 90° | 2047 | 3 | ±3 | 7 |

Proposal 4 Adopt the above table for the definition of the representation format of a coarse UE location.

**Question 4.5.4: Do companies agree with the proposal above ?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree/Disagree** | **Comments/Suggestions** |
| Xiaomi | Agree with comment | Agree with the intention, but we think the needed bits for each row should be N-1. I.e. for first row, 13-1 = 12; for second row, 12-1 = 11bits.  Because, as we calculated for longitude at equator, ±2km range corresponds to 12 bits of degreeLongitude (1 bit corresponds to 1.19 meter). The total bits of degreeLongitude is 24 bits (including sign). thus, 12 MSB of degreeLongitude(24-12) is needed for ±2km accuracy. |
| Ericsson | Agree | Due to the quantization step it is impossible to achieve a perfect representation, but this table comes reasonably close. |
|  |  |  |
| Nokia | Agree |  |

## 2.5 fields of coarse UE location information

Proposal 5 For representing a coarse UE location, use 12 bits for the latitude (where one bit indicates “north”/”south”), 3-13 bits for the longitude (where the number of bits depends on the latitude), and 4 bits for the optional altitude (where one bit is used to indicate “height”/”depth”).

**Question 4.5.5: Do companies agree with the proposal above ?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree/Disagree** | **Comments/Suggestions** |
| Xiaomi | Disagree | For latitude, we agree that 11 most significant bits of degreeLatitude is required (±2km range corresponds to 12 bits of degreeLatitude, 23-12 = 11 MSB), additionally we need latitudeSign IE to indicate the north/south of latitude.  For longitude at equator, same as latitude, ±2km range corresponds to 12 bits of degreeLongitude, thus, 12 MSB of degreeLongitude(24-12) is needed for ±2km accuracy.  For Altitude, it is not needed. |
| Ericsson | Agree |  |
| Thales | Agree |  |
| OPPO |  | Altitude is not needed. |
| Lenovo |  | Altitude is not needed. |
| Nokia | OK | Apart from the altitude. |

## 2.6 ASN.1 code

Two options are proposed for ASN.1 code of the coarse UE location information

Option 1: “*CoarseEllipsoidPointWithOptionalAltitude*”

CoarseEllipsoidPointWithOptionalAltitude ::= SEQUENCE {

latitudeSign ENUMERATED {north, south},

degreesLatitude INTEGER (0..2047), -- 11 bit field

degreesLongitude CHOICE {

longForLatRange1 INTEGER (-4095..4095), -- For latitudes 0-1364. 13 bit field.

LongForLatRange2 INTEGER (-2047..2047), -- For latitudes 1365-1718. 12 bit field.

LongForLatRange3 INTEGER (-1023..1023), -- For latitudes 1719-1884. 11 bit field.

LongForLatRange4 INTEGER (-511..511), -- For latitudes 1885-1966. 10 bit field.

LongForLatRange5 INTEGER (-255..255), -- For latitudes 1967-2006. 9 bit field.

LongForLatRange6 INTEGER (-127..127), -- For latitudes 2007-2027. 8 bit field.

LongForLatRange7 INTEGER (-63..63), -- For latitudes 2028-2037. 7 bit field.

LongForLatRange8 INTEGER (-31..31), -- For latitudes 2038-2043. 6 bit field.

LongForLatRange9 INTEGER (-15..15), -- For latitudes 2044-2045. 5 bit field.

LongForLatRange10 INTEGER (-7..7), -- For latitude 2046. 4 bit field.

LongForLatRange11 INTEGER (-3..3) -- For latitude 2047. 3 bit field.

},

altitudeDirection ENUMERATED {height, depth} OPTIONAL,

altitude INTEGER (0..7) OPTIONAL -- 3 bit field. Step size: 4.096 km.

}

Option 2: ASN.1 code is divided into two Ies:

* one with altitude included (tentatively denoted as “*CoarseEllipsoidPointWithAltitude*”)
* and one without the altitude included (tentatively denoted as “*CoarseEllipsoidPoint*”).

CoarseEllipsoidPointWithAltitude ::= SEQUENCE {

latitudeSign ENUMERATED {north, south},

degreesLatitude INTEGER (0..2047), -- 11 bit field

degreesLongitude CHOICE {

longForLatRange1 INTEGER (-4095..4095), -- For latitudes 0-1364. 13 bit field.

LongForLatRange2 INTEGER (-2047..2047), -- For latitudes 1365-1718. 12 bit field.

LongForLatRange3 INTEGER (-1023..1023), -- For latitudes 1719-1884. 11 bit field.

LongForLatRange4 INTEGER (-511..511), -- For latitudes 1885-1966. 10 bit field.

LongForLatRange5 INTEGER (-255..255), -- For latitudes 1967-2006. 9 bit field.

LongForLatRange6 INTEGER (-127..127), -- For latitudes 2007-2027. 8 bit field.

LongForLatRange7 INTEGER (-63..63), -- For latitudes 2028-2037. 7 bit field.

LongForLatRange8 INTEGER (-31..31), -- For latitudes 2038-2043. 6 bit field.

LongForLatRange9 INTEGER (-15..15), -- For latitudes 2044-2045. 5 bit field.

LongForLatRange10 INTEGER (-7..7), -- For latitude 2046. 4 bit field.

LongForLatRange11 INTEGER (-3..3) -- For latitude 2047. 3 bit field.

},

altitudeDirection ENUMERATED {height, depth},

altitude INTEGER (0..7) -- 3 bit field. Step size: 4.096 km.

}

CoarseEllipsoidPoint ::= SEQUENCE {

latitudeSign ENUMERATED {north, south},

degreesLatitude INTEGER (0..2047), -- 11 bit field

degreesLongitude CHOICE {

longForLatRange1 INTEGER (-4095..4095), -- For latitudes 0-1364. 13 bit field.

LongForLatRange2 INTEGER (-2047..2047), -- For latitudes 1365-1718. 12 bit field.

LongForLatRange3 INTEGER (-1023..1023), -- For latitudes 1719-1884. 11 bit field.

LongForLatRange4 INTEGER (-511..511), -- For latitudes 1885-1966. 10 bit field.

LongForLatRange5 INTEGER (-255..255), -- For latitudes 1967-2006. 9 bit field.

LongForLatRange6 INTEGER (-127..127), -- For latitudes 2007-2027. 8 bit field.

LongForLatRange7 INTEGER (-63..63), -- For latitudes 2028-2037. 7 bit field.

LongForLatRange8 INTEGER (-31..31), -- For latitudes 2038-2043. 6 bit field.

LongForLatRange9 INTEGER (-15..15), -- For latitudes 2044-2045. 5 bit field.

LongForLatRange10 INTEGER (-7..7), -- For latitude 2046. 4 bit field.

LongForLatRange11 INTEGER (-3..3) -- For latitude 2047. 3 bit field.

}

}

**Question 4.5.6: Which option (see above) do companies prefer for ASN.1 code of coarse UE location information ?**

**• Option 1: CoarseEllipsoidPointWithOptionalAltitude IE**

**• Option 2: CoarseEllipsoidPointWithAltitude IE + CoarseEllipsoidPoint IE**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option 1 / Option 2** | **Comments/Suggestions** |
| Ericsson | Option 1 (slight preference) | Both options works and they are more or less equivalent. Maybe a slight preference for option 1, because of a slightly smaller specification effort. |
| Thales | Neutral |  |
| OPPO |  | Altitude may not be needed. |
| Lenovo |  | Altitude is not needed. |
| Nokia |  | Agree, altitude may not be needed, so none of these. IE without Altitude should be used. |

## 2.7 Reporting mechanism

**Question 4.5.7: Which reporting mechanisms do companies prefer**

* **Option 1: After AS security is established, gNB can obtain a GNSS-based location information from the UE using existing signalling method, i.e., by configuring includeCommonLocationInfo in the corresponding reportConfig**
* **Option 2: After AS security is established, gNB can obtain a GNSS-based coarse location information from the UE by indicates a new IE ‘includeCoarseCommonLocationInfo’ in ReportConfigNR**
* **Option 3: other**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option 1 / Option 2** | **Comments/Suggestions** |
| Xiaomi | Option 2 | The issue with option 1 is that UE doesn’t know whether to report full GNSS location or coarse GNSS location. Option 2 can solve the issue. |
| Ericsson | Option 1/2 and Option 3 | Both options are relevant. It may be good that the reporting of the UE’s coarse location is not tied to reporting of RRM measurements, since the use of the RRM measurement results and the use of the UE’s coarse location information may be quite different, used for different purposes and in different contexts. In one use case, the UE reports its coarse location on request from the network. For this, we would suggest *UEInformationRequest*/*UEInformationResponse* messages (i.e. option 2). But we also need event-triggered reporting of the UE’s coarse location, which option 1 could provide.  UEInformationRequest/Response would be quite straightforward:  UEInformationRequest-v1700-IEs ::= SEQUENCE {  successHO-ReportReq-r17 ENUMERATED {true} OPTIONAL, -- Need N  coarseLocationRequest-r17 ENUMERATED {true} OPTIONAL, -- Need N  nonCriticalExtension SEQUENCE {} OPTIONAL  }  UEInformationResponse-v1700-IEs ::= SEQUENCE {  successHO-Report-r17 SuccessHO-Report-r17 OPTIONAL,  connEstFailReportList-r17 ConnEstFailReportList-r17 OPTIONAL,  coarseLocationInfo CoarseEllipsoidPointWithOptionalAltitude-r17 OPTIONAL,  nonCriticalExtension SEQUENCE {} OPTIONAL  } |
| Thales | Neutral |  |
| OPPO | Option 3 | We also think using *UEInformationRequest*/*UEInformationResponse* messages is simple and straightforward. |
| Lenovo | Neutral |  |
| Nokia | Option 1 | We think a legacy mechanism should be used. It is true, this is not related to RRM measurements, but the UE can report empty/no results for RRM measurement (using the same reporting framework). |

# Way forward

During the on line session, the following was agreed:

Agreements:

1. When defining a coarse UE location representation format, RAN2 agrees in principle to use the definition of EllipsoidPointWithAltitude in TS 37.355 and round the coordinates to fewer bits to achieve a suitable accuracy. No altitude is reported. The actual name could be the existing EllipsoidPoint (no altitude included) or coarseEllipsoidPoint.

On the basis of this agreement and considering responses collected in 1st round, a new set of questions are proposed.

## 3.1 Which definition to start from ?

TS 37.355 defines several IEs to describe a geographical shape among which two have been identified as of particular relevance for the discussion

Option 1: IE EllipsoidPoint

Option 2: IE EllipsoidPointWithAltitude

Moderator suggests adopting option 1 since it is the simplest and does not require any CR to TS 37.355. Moreover, it is fits the purpose since we don’t need the altitude. Therefore Moderator suggests the following:

**Proposal 3.1: The coarse UE location representation format is based on the existing IE EllipsoidPoint**

## 3.2 Generating the Coarse UE location information

For generating the coarse UE location, the UE is expected to round the coordinates to fewer bits to achieve a suitable accuracy.

There are three ways the generation of this coarse UE location can be achieved:

* Option 1: UE by implementation sets to zero different length of LSBs of longitude and of latitude based on the accuracy requirement (to compensate for the different longitude density).
* Option 2: Define a table to set different length of LSBs of longitude to zero based on the value of latitude that the UE will use in its rounding.
* Option 3: Always set a fixed number of LSB of longitude to 0; This number can be [12]

Note that in the 1st round discussion, there seems to be a majority view to compensate for the different longitude density at different latitudes when doing the “rounding”.

Moderator suggests adopting option 1 since it has the least spec impact and highest flexibility. Opt3 doesn’t support the compensation for the different longitude density at different latitudes. Therefore Moderator suggests the following:

**Proposal 3.2.1: By implementation, the UE generates the coarse UE location by setting to zero different length of LSBs of the longitude and of the latitude based on the accuracy requirement (to compensate for the different longitude density)**

**Proposal 3.2.2: In Rel-17, the accuracy requirement is approximately 2 km.**

## 3.3 Signalling the Coarse UE location information

From the previous round, it seems there are three options

* **Option 1: After AS security is established, gNB can obtain a GNSS-based location information from the UE using existing signalling method, i.e., by configuring includeCommonLocationInfo in the corresponding reportConfig**
* **Option 2: After AS security is established, gNB can obtain a GNSS-based coarse location information from the UE by indicating a new IE ‘includeCoarseCommonLocationInfo’ in ReportConfigNR**
* **Option 3: After AS security is established, gNB can obtain a GNSS-based coarse location information from the UE by adding coarseLocationRequest-r17 and coarseLocationInfo in respectively the UEInformationRequest and the UEInformationResponse.**

Moderator suggests adopting option 3 since it is a clean solution. Therefore Moderator suggests the following:

**Proposal 3.3: After AS security is established, gNB can obtain a GNSS-based coarse location information from the UE by adding coarseLocationRequest-r17 and coarseLocationInfo in respectively the UEInformationRequest and the UEInformationResponse**

## 3.4 Capturing the requirement for coarse location report

There are different options to capture the requirement for coarse location report in TS 38.331:

* Option 1: Directly describe the requirement where the LSBs are set;
* Option 2: Have a dedicated chapter to define the requirement of coarse location;
* Option 3: Describe the requirement in the field description of the coarse UE location info request

Moderator suggests adopting option 3 since it is the solution having less spec impact. Therefore Moderator suggests the following:

**Proposal 3.4: The requirement for coarse location report are captured in TS 38.331 directly in the field description of the IE coarseLocationRequest-r17**

# 4. Summary and Proposals

From the above, moderator suggests as a way forward the following proposals:

**Proposal 3.1: The coarse UE location representation format is based on the existing IE EllipsoidPoint**

**Proposal 3.2.1: By implementation, the UE generates the coarse UE location by setting to zero different length of LSBs of the longitude and of the latitude based on the accuracy requirement (to compensate for the different longitude density)**

**Proposal 3.2.2: In Rel-17, the accuracy requirement is approximately 2 km.**

**Proposal 3.3: After AS security is established, gNB can obtain a GNSS-based coarse location information from the UE by adding coarseLocationRequest-r17 and coarseLocationInfo in respectively the UEInformationRequest and the UEInformationResponse**

**Proposal 3.4: The requirement for coarse location report are captured in TS 38.331 directly in the field description of the IE coarseLocationRequest-r17**

Based on this (and possible feedbacks), the moderator will prepare a TP to TS 38.331.

# 4. References

1. R2-2205574 Coarse location format Ericsson
2. R2-2205574 Coarse location format Ericsson

# Contact information

|  |  |
| --- | --- |
| Company | Delegate contact |
| COMPANY\_NAME | NAME ([email@address.com](mailto:email@address.com)) |
| Thales | nicolas.chuberre@thalesaleniaspace.com |
| Lenovo | Min Xu (xumin13@lenovo.com) |
|  |  |
|  |  |
|  |  |