**3GPP TSG-WG SA4 Meeting #117E e-meeting  *S4-220122***

**Elbonia, February 14 – 23, 2022 (revision of S4-220xxx)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **26.114** | **CR** | **0524** | **rev** | **-** | **Current version:** | **17.3.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps | **X** | ME | **X** | Radio Access Network |  | Core Network |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Add support of per-slice QoE measurement | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Huawei, HiSilicon | | | | | | | | | |
| ***Source to TSG:*** | SA4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | NR\_QoE-Core | | | | |  | ***Date:*** | | | 2022-01-28 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***Release:*** | | | Rel-17 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | In the LSes from RAN3 (R3-214477, R3-216225), the slice ID is agreed to be added into the QoE reports for per-slice QoE reporting and evalution.  In current RAN3 agreements, the slice scope shall be outside of the transparent QoE configuraiton container and RAN can based on the slice scope to send the QoE configuration to the UEs. During the PDU Session Establishment procedure, the slice Id related to the established PDU Session will be sent to RAN by SMF. So RAN can decide to send QoE configuration to the UEs where there are PDU Sessions using the target network slices. However, RAN cannot figure out whether the media streaming service is using the target network slice or not. Therefore, the slice scope/fiter is also needed in the QoE report.  Besides, as indicated in LS R2-2202018 from RAN2, the maximum size of one QoE config/report container has changed, so alignments are needed from the SA4 perspective. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Add support of per-slice QoE measurements and calrify the maximum size of QoE containers. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Unalignments between SA4 and RAN3. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 16.4, 16.5.1 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

\* \* \* \* First change \* \* \* \*

## 16.4 Metrics Reporting

When a session is started, the MTSI client must determine whether QoE reporting is required for the session. If the parameter "Enabled" is set to false, no QoE reporting shall be done. If the "Enabled" parameter is set to true the optional "Rules" parameters are checked (sub-clause 16.3.3) to define if QoE reporting shall be done.

Once the need for QoE reporting has been established, the client shall continuously compute all specified metrics for each measurement interval period, according to the "Measure-Resolution" parameter (sub-clause 16.3.2). In order to bound the resources used by metrics reporting, the minimum values for the Measure-Resolution and Sending-Rate are specified to be 5 seconds and 30 seconds respectively. The computed metrics are represented in a vector format, adding an additional metric value to each metric vector after each new measurement interval period.

Note that the calculated metrics shall only cover one measurement interval. For instance, if the corruption duration extends longer than to the end of the current measurement interval, only the portion which fits into the current measurement interval shall be reported. The remaining portion of the corruption duration shall be reported as belonging to the next measurement interval.

The end of the session will normally not correspond to the end of a measurement interval period, so the metrics for the last measurement interval period will typically be calculated over a time shorter than the configured measurement interval. Note, however, that these last metrics shall still be added to the metrics vectors and reported to the server.

It is possible for the server to use the start and stop timestamps, together with the knowledge of the configured measurement interval, to derive the actual length of the last measurement interval period, but any specific action or interpretation of these last shorter measurements is out of scope of this specification.

The MTSI client shall send QoE report messages to the server in accordance with the specified reporting interval "Sending-Rate" (sub-clause 16.3.2). All stored metrics data shall then be sent to the server, and then deleted from the metrics storage.

Note that if the reporting interval is not an integer multiple of the measurement interval, only the measurement interval periods which have been fully passed shall be included in the report. The ongoing not-passed measurement interval period shall be included in the next report. The only exception is at the end of the session, where also the last ongoing measurement interval period shall be directly calculated and included in the report.

If QoE configuration has been done via the OMA MO, the client shall send QoE reports using the HTTP (RFC 2616 [73]) POST request carrying XML formatted metadata. If the optional "APN" parameter is defined in the OMA managed object, that APN shall be used for establishing the PDP context or EPS bearer on which the QoE metric reports will be transmitted. The MTSI client randomly selects one of the URIs from the MO "Server" parameter, with uniform distribution.

If QoE configuration has been done via the QMC functionality (see clause 16.5), the client shall also send the QoE reports as described in clause 16.5. Note that for QMC scheme, the SliceId shall be included in the QoE report for per-slice QoE reporting and evaluation.

Each QoE report is formatted in XML according the following XML schema (sub-clause 16.4.1). An informative example of a single reception report XML object is also given (sub-clause 16.4.2). The reports should be compressed using GZIP only if the MO parameter "Format" specifies this.

Each QoE Metrics element has a set of attributes and any number of media level QoE Metrics elements. All attributes are defined in sub-clause 16.4.1 and correspond to the QoE metrics listed in sub-clause 16.2. Individual metrics can be selected as described in sub-clause 16.3.2.

Except for the media level QoE metrics, the following parameters shall be reported for each report:

- The *callId* attribute identifies the call identity of the SIP session.

- The *clientId* attribute is unique identifier for the receiver, e.g. an MSISDN of the UE as defined in [80].

- The *startTime* and *stopTime* attributes identifies the client NTP time when the measurements included in the report were started and stopped. The time is based on the local real-time clock in the client, and might not be consistent with the true NTP time. However, assuming that the reporting is done without any extra delay the server can use the *stopTime* attribute to correct the timestamps if necessary.

- The *mediaId* attribute shall be reported for each media level QoE report, and identifies the port number for the media.

If the attribute *qoeReferenceId* was defined in the QMC configuration (see clause 16.5.2), the value shall be copied into each QoE report, to facilitate network-side correlation (see [178]). If this attribute was defined, the attribute *recordingSessionId* shall also be returned for each QoE report. The *recordingSessionId* is a two-byte octet defined by the client. It shall remain the same for all QoE reports belonging to the same session, and it should be different for QoE reports belonging to different sessions.

### 16.4.1 XML schema for QoE report message

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

targetNamespace="urn:3gpp:metadata:2008:MTSI:qoereport"

xmlns="urn:3gpp:metadata:2008:MTSI:qoereport"

elementFormDefault="qualified">

<xs:element name="QoeReport" type="QoeReportType"/>

<xs:complexType name="QoeReportType">

<xs:sequence>

<xs:element name="statisticalReport" type="starType" minOccurs="0"

maxOccurs="unbounded"/>

<xs:any namespace="##other" processContents="skip" minOccurs="0"

maxOccurs="unbounded"/>

</xs:sequence>

<xs:anyAttribute processContents="skip"/>

</xs:complexType>

<xs:complexType name="starType">

<xs:sequence>

<xs:element name="mediaLevelQoeMetrics" type="mediaLevelQoeMetricsType" minOccurs="1"

maxOccurs="unbounded"/>

</xs:sequence>

<xs:attribute name="startTime" type="xs:unsignedLong" use="required"/>

<xs:attribute name="stopTime" type="xs:unsignedLong" use="required"/>

<xs:attribute name="callId" type="xs:string" use="required"/>

<xs:attribute name="clientId" type="xs:string" use="required"/>

<xs:attribute name="sliceId" type="xs:UnsignedLong" use="optional"/>

<xs:attribute name="qoeReferenceId" type="xs:hexBinary" use="optional"/>

<xs:attribute name="recordingSessionId" type="xs:hexBinary" use="optional"/>

<xs:anyAttribute processContents="skip"/>

</xs:complexType>

<xs:complexType name="mediaLevelQoeMetricsType">

<xs:sequence>

<xs:any namespace="##other" processContents="skip" minOccurs="0"

maxOccurs="unbounded"/>

</xs:sequence>

<xs:attribute name="mediaId" type="xs:integer" use="required"/>

<xs:attribute name="totalCorruptionDuration" type="unsignedLongVectorType"  
 use="optional"/>

<xs:attribute name="numberOfCorruptionEvents" type="unsignedLongVectorType"  
 use="optional"/>

<xs:attribute name="corruptionAlternative" type="xs:string" use="optional"/>

<xs:attribute name="totalNumberofSuccessivePacketLoss" type="unsignedLongVectorType"

use="optional"/>

<xs:attribute name="numberOfSuccessiveLossEvents" type="unsignedLongVectorType"   
 use="optional"/>

<xs:attribute name="numberOfReceivedPackets" type="unsignedLongVectorType"   
 use="optional"/>

<xs:attribute name="framerate" type="doubleVectorType" use="optional"/>

<xs:attribute name="totalJitterDuration" type="doubleVectorType" use="optional"/>

<xs:attribute name="numberOfJitterEvents" type="unsignedLongVectorType"

use="optional"/>

<xs:attribute name="totalSyncLossDuration" type="doubleVectorType" use="optional"/>

<xs:attribute name="numberOfSyncLossEvents" type="unsignedLongVectorType"

use="optional"/>

<xs:attribute name="networkRTT" type="unsignedLongVectorType" use="optional"/>

<xs:attribute name="internalRTT" type="unsignedLongVectorType" use="optional"/>

<xs:attribute name="codecInfo" type="stringVectorType" use="optional"/>

<xs:attribute name="codecProfileLevel" type="stringVectorType" use="optional"/>

<xs:attribute name="codecImageSize" type="stringVectorType" use="optional"/>

<xs:attribute name="averageCodecBitrate" type="doubleVectorType" use="optional"/>

<xs:attribute name="callSetupTime" type="xs:unsignedLong" use="optional"/>

<xs:anyAttribute processContents="skip"/>

</xs:complexType>

<xs:simpleType name="doubleVectorType">

<xs:list itemType="xs:double"/>

</xs:simpleType>

<xs:simpleType name="stringVectorType">

<xs:list itemType="xs:string"/>

</xs:simpleType>

<xs:simpleType name="unsignedLongVectorType">

<xs:list itemType="xs:unsignedLong"/>

</xs:simpleType>

</xs:schema>

### 16.4.2 Example XML for QoE report message

Below is one example of QoE report message, in this example the measurement interval is 20 seconds, the reporting interval is 5 minutes, but the call ends after 55 seconds.

<?xml version="1.0" encoding="UTF-8"?>

<QoeReport xmlns="urn:3gpp:metadata:2008:MTSI:qoereport"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="urn:3gpp:metadata:2008:MTSI:qoereport qoereport.xsd">

<statisticalReport

startTime="1219322514"

stopTime="1219322569"

clientId="clientID"

callId="callID">

qoeReferenceId="240F512A"

sliceId="01000FFF"

recordingSessionId="0001"

<mediaLevelQoeMetrics

mediaId="1234"

totalCorruptionDuration="480 0 120"

numberOfCorruptionEvents="5 0 2"

corruptionAlternative="a"

totalNumberofSuccessivePacketLoss="24 0 6"

numberOfSuccessiveLossEvents="5 0 2"

numberOfReceivedPackets="535 645 300"

framerate="50.0 49.2 50.0"

numberOfJitterEvents="0 1 0"

totalJitterDuration="0 0.346 0"

networkRTT="120 132 125"

internalRTT="20 24 20"

codecInfo="AMR-WB/16000/1 = ="

averageCodecBitRate="12.4 12.65 12.7"/>

callSetupTime="345"

<mediaLevelQoeMetrics

mediaId="1236"

totalCorruptionDuration="83 0 0"

numberOfCorruptionEvents="1 0 0"

corruptionAlternative="b"

totalNumberofSuccessivePacketLoss="3 0 0"

numberOfSuccessiveLossEvents="2 0 0"

numberOfReceivedPackets="297 300 225"

framerate="14.7 15.0 14.9"

numberOfJitterEvents="0 0 0"

totalJitterDuration="0 0 0"

numberOfSyncLossEvents="0 1 0"

totalSyncLossDuration="0 0.789 0"

networkRTT="220 232 215"

internalRTT="27 20 25"

codecInfo="H263-2000/90000 = ="

codecProfileLevel="profile=0;level=45 = ="

codecImageSize="176x144 = ="

averageCodecBitRate="124.5 128.0 115.1"/>

callSetupTime="345"/>

</statisticalReport>

</QoeReport>

\* \* \* \* Fourth change \* \* \* \*

### 16.5.1 Configuration and reporting

As an alternative to configuration via OMA-DM, the QoE configuration can optionally be specified by the QoE Measurement Collection (QMC) functionality. In this case the QoE configuration is received via specific RRC [158] messages over the control plane, and the QoE reporting is also sent back via RRC messages over the control plane.

If QMC is supported, the UE shall support the following QMC functionalities:

- QoE Configuration: The QoE configuration will be delivered via RRC to the UE as a container according to "Application Layer Measurement Configuration" (see [158]) for UMTS, and "measConfigApplicationLayer" (see [160]) for LTE. The container is an octet string with a maximum length of 8000 bytes, with gzip-encoded data (see [71]) stored in network byte order. When the container is uncompressed it is expected to conform to XML-formatted QoE configuration data according to clause 16.5.2 in the current specification. This uncompressed QoE Configuration shall be delivered to the MTSI client. The interface towards the RRC signalling is handled by the AT command +CAPPLEVMC [161].

- QoE Metrics: QoE Metrics from the MTSI client shall be XML-formatted according to clause 16.4 in the current specification. The XML data shall be compressed with gzip (see [71]) and stored in network byte order into an octet string container with a maximum length of 8000 bytes (without RRC segmentation) or 144 000 bytes (with RRC segmentation). The container shall be delivered via RRC to the RNC according to "Application Layer Measurement Reporting" (see [158]) for UMTS, and to the eNB according to "measReportApplicationLayer" (see [160]) for LTE. The behaviour if the compressed data is larger than the maximum number of allwoed bytes is unspecified in this version of the specification. The interface towards the RRC signalling is handled by the AT command +CAPPLEVMR [161].

- The UE shall also set the QMC capability "QoE Measurement Collection for MTSI services" (see [158]) to TRUE for UMTS, and include the QMC capability "qoe-mtsi-MeasReport" (see [160]) for LTE.

The QoE configuration AT command +CAPPLEVMC [161] may also indicate with an Within-area Indication if the UE is inside or outside a wanted geographic area. Such an indication may arrive with or without any QoE configuration container attached. If the MTSI client is informed that it is not inside the area, it shall not start any new QoE measurements even if it has received a valid QoE configuration container, but shall continue measuring for already started sessions.

When a new session is started, the QoE reporting AT command +CAPPLEVMR [161] shall be used to send a Recording Session Indication. Such an indication does not contain any QoE report, but indicates that QoE recording has started for a session.

When the QoE configuration is to be released, an unsolicited result code associated with the AT command +CAPPLEVMC [161] and containing the parameter <start-stop\_reporting> set to "1", shall be sent to the MTSI client as notification of a discard request. Then the MTSI client shall stop collecting quality metrics and discard any already collected information [178].

The exact implementation is not specified here, but an example signalling diagram below shows the QMC functionality with a hypothetical "QMC Handler" entity.

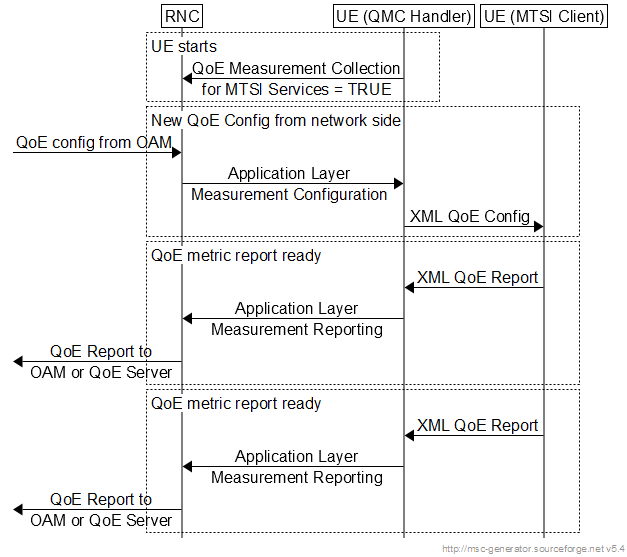


Figure 16.5.1-1: Example signalling diagram for UMTS

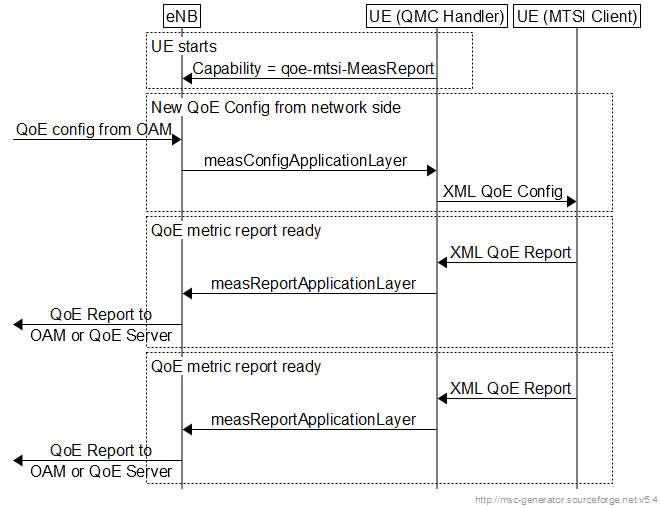


Figure 16.5.1-2: Example signalling diagram for LTE

Note that the QMC Handler is only shown here as one possible implementation, and it need not be implemented as such. The corresponding QMC functionality could be built into the MTSI client or into other UE entities. In this version of the specification the detailed implementation of the above functionalities is left to the UE vendor.

\* \* \* \* Fifth change \* \* \* \*

\* \* \* \* End of changes \* \* \* \*