**3GPP TSG-SA5 Meeting #155 *S5-243409***

**Jeju, Korea (Republic Of), 27th May 2024 - 31st May 2024**

|  |
| --- |
| *CR-Form-v12.3* |
| **CHANGE REQUEST** |
|  |
|  | **28.623** | **CR** | **0364** | **rev** | **1** | **Current version:** | **17.9.0** |  |
|  |
| *For* [***HELP***](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 |  | ME |  | Radio Access Network | **X** | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | Rel-17 CR 28.623 Fix trace attribute definition (stage 3, yang) |
|  |  |
| ***Source to WG:*** | Ericsson LM |
| ***Source to TSG:*** | S5 |
|  |  |
| ***Work item code:*** | TEI17 |  | ***Date:*** | 2024-05-17 |
|  |  |  |  |  |
| ***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 canbe 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-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19) Rel-20 (Release 20)* |
|  |  |
| ***Reason for change:*** | Support for reporting which measurements are supported exists in the NRM (supportedPerfMetrics, supportedTraceMetrics). PerfMetricJob uses such information to allow configuration of which measurements to collect. Similar functionality is however missing for TraceJob.  |
|  |  |
| ***Summary of change:*** | Add configuration of which trace mesages to collect to TraceJob  |
|  |  |
| ***Consequences if not approved:*** | There is no way to configure the trace messages despite them being reported in the supported trace messaage group. |
|  |  |
| ***Clauses affected:*** | Forge only |
|  |  |
|  | **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 28.622 CR 0360  |
|  |  |
| ***Other comments:*** | Forge MR link: <https://forge.3gpp.org/rep/sa5/MnS/-/merge_requests/1147> at commit 3737996cd90adbecccffcefac13e1a0a2295ccc5 |
|  |  |
| ***This CR's revision history:*** |  |

\*\*\* START OF CHANGE 1 \*\*\*

\*\*\* yang-models/\_3gpp-common-trace.yang \*\*\*

<CODE BEGINS>

module \_3gpp-common-trace {

 yang-version 1.1;

 namespace "urn:3gpp:sa5:\_3gpp-common-trace";

 prefix "trace3gpp";

 import \_3gpp-common-top { prefix top3gpp; }

 import \_3gpp-common-yang-types {prefix types3gpp; }

 import \_3gpp-common-yang-extensions { prefix yext3gpp; }

 import ietf-inet-types { prefix inet; }

 import \_3gpp-common-files { prefix files3gpp; }

 organization "3GPP SA5";

 contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

 description "Trace handling

 Copyright 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI,

 TTA, TTC). All rights reserved.";

 reference "3GPP TS 28.623

 Generic Network Resource Model (NRM)

 Integration Reference Point (IRP);

 Solution Set (SS) definitions

 3GPP TS 28.622

 Generic Network Resource Model (NRM)

 Integration Reference Point (IRP);

 Information Service (IS)";

 revision 2024-05-13 { reference "CR-0360" ; }

 revision 2024-04-05 { reference "CR-0341" ; }

 revision 2023-11-05 { reference "CR-0293 CR-0301" ; }

 revision 2023-09-17 { reference CR-0270 ; }

 revision 2023-07-25 { reference CR00252; }

 revision 2023-02-17 { reference "CR-0232"; }

 revision 2023-02-15 { reference "CR-0236"; }

 revision 2022-04-27 { reference "CR-0159"; }

 revision 2021-10-18 { reference "CR-0139"; }

 revision 2021-07-22 { reference "CR-0137"; }

 revision 2021-01-25 { reference "CR-0122"; }

 revision 2020-11-16 { reference "CR-0117"; }

 revision 2020-08-06 { reference "CR-0102"; }

 feature FilesUnderTraceJob {

 description "Files shall be contained under TraceJob";

 }

 grouping TraceReference {

 leaf mcc {

 mandatory true;

 type types3gpp:Mcc;

 }

 leaf mnc {

 mandatory true;

 type types3gpp:Mnc;

 }

 leaf traceId {

 mandatory true;

 type int64;

 }

 }

 grouping TraceJobGrp {

 leaf jobType {

 type enumeration {

 enum IMMEDIATE\_MDT\_ONLY;

 enum LOGGED\_MDT\_ONLY;

 enum TRACE\_ONLY;

 enum IMMEDIATE\_MDT\_AND\_TRACE;

 enum RLF\_REPORT\_ONLY;

 enum RCEF\_REPORT\_ONLY;

 enum LOGGED\_MBSFN\_MDT;

 }

 default TRACE\_ONLY;

 description "Specifies the MDT mode and it specifies also whether the

 TraceJob represents only MDT, Logged MBSFN MDT, Trace or a combined

 Trace and MDT job. The attribute is applicable for Trace, MDT, RCEF and

 RLF reporting.";

 reference "Clause 5.9a of 3GPP TS 32.422 for additional details on the

 allowed values.";

 }

 list listOfInterfaces {

 key idx;

 must 'count(MSCServerInterfaces)+count(MGWInterfaces)+count(RNCInterfaces)'

 +'+count(SGSNInterfaces)+count(GGSNInterfaces)+count(S-CSCFInterfaces)'

 +'+count(P-CSCFInterfaces)+count(I-CSCFInterfaces)+count(MRFCInterfaces)'

 +'+count(MGCFInterfaces)+count(IBCFInterfaces)+count(E-CSCFInterfaces)'

 +'+count(BGCFInterfaces)+count(ASInterfaces)+count(HSSInterfaces)'

 +'+count(EIRInterfaces)+count(BM-SCInterfaces)+count(MMEInterfaces)'

 +'+count(SGWInterfaces)+count(PDN\_GWInterfaces)+count(eNBInterfaces)'

 +'+count(en-gNBInterfaces)+count(AMFInterfaces)+count(AUSFInterfaces)'

 +'+count(NEFInterfaces)+count(NRFInterfaces)+count(NSSFInterfaces)'

 +'+count(PCFInterfaces)+count(SMFInterfaces)+count(SMSFInterfaces)'

 +'+count(UDMInterfaces)+count(UPFInterfaces)+count(ng-eNBInterfaces)'

 +'+count(gNB-CU-CPInterfaces)+count(gNB-CU-UPInterfaces)'

 +'+count(gNB-DUInterfaces)';

 description "Specifies the interfaces that need to be traced in the given

 ManagedEntityFunction.The attribute is applicable only for Trace. In

 case this attribute is not used, it carries a null semantic.";

 reference "Clause 5.5 of 3GPP TS 32.422 for additional details on the

 allowed values.";

 leaf idx { type uint32 ; }

 leaf-list MSCServerInterfaces {

 type enumeration {

 enum A ;

 enum Iu-CS ;

 enum Mc ;

 enum MAP-G ;

 enum MAP-B ;

 enum MAP-E ;

 enum MAP-F ;

 enum MAP-D ;

 enum MAP-C ;

 enum CAP ;

 }

 }

 leaf-list MGWInterfaces {

 type enumeration {

 enum Mc ;

 enum Nb-UP ;

 enum Iu-UP ;

 }

 }

 leaf-list RNCInterfaces {

 type enumeration {

 enum Iu-CS ;

 enum Iu-PS ;

 enum Iur ;

 enum Iub ;

 enum Uu ;

 }

 }

 leaf-list SGSNInterfaces {

 type enumeration {

 enum Gb ;

 enum Iu-PS ;

 enum Gn ;

 enum MAP-Gr ;

 enum MAP-Gd ;

 enum MAP-Gf ;

 enum Ge ;

 enum Gs ;

 enum S6d ;

 enum S4 ;

 enum S3 ;

 enum S13 ;

 }

 }

 leaf-list GGSNInterfaces {

 type enumeration {

 enum Gn ;

 enum Gi ;

 enum Gmb ;

 }

 }

 leaf-list S-CSCFInterfaces {

 type enumeration {

 enum Mw ;

 enum Mg ;

 enum Mr ;

 enum Mi ;

 }

 }

 leaf-list P-CSCFInterfaces {

 type enumeration {

 enum Gm ;

 enum Mw ;

 }

 }

 leaf-list I-CSCFInterfaces {

 type enumeration {

 enum Cx ;

 enum Dx ;

 enum Mg ;

 enum Mw ;

 }

 }

 leaf-list MRFCInterfaces {

 type enumeration {

 enum Mp ;

 enum Mr ;

 }

 }

 leaf-list MGCFInterfaces {

 type enumeration {

 enum Mg ;

 enum Mj ;

 enum Mn ;

 }

 }

 leaf-list IBCFInterfaces {

 type enumeration {

 enum Ix ;

 enum Mx ;

 }

 }

 leaf-list E-CSCFInterfaces {

 type enumeration {

 enum Mw ;

 enum Ml ;

 enum Mm ;

 enum Mi-Mg ;

 }

 }

 leaf-list BGCFInterfaces {

 type enumeration {

 enum Mi ;

 enum Mj ;

 enum Mk ;

 }

 }

 leaf-list ASInterfaces {

 type enumeration {

 enum Dh ;

 enum Sh ;

 enum ISC ;

 enum Ut ;

 }

 }

 leaf-list HSSInterfaces {

 type enumeration {

 enum MAP-C ;

 enum MAP-D ;

 enum Gc ;

 enum Gr ;

 enum Cx ;

 enum S6d ;

 enum S6a ;

 enum Sh ;

 }

 }

 leaf-list EIRInterfaces {

 type enumeration {

 enum MAP-F ;

 enum S13 ;

 enum MAP-Gf ;

 }

 }

 leaf-list BM-SCInterfaces {

 type enumeration {

 enum Gmb ;

 }

 }

 leaf-list MMEInterfaces {

 type enumeration {

 enum S1-MME ;

 enum S3 ;

 enum S6a ;

 enum S10 ;

 enum S11 ;

 enum S13 ;

 }

 }

 leaf-list SGWInterfaces {

 type enumeration {

 enum S4 ;

 enum S5 ;

 enum S8 ;

 enum S11 ;

 enum Gxc ;

 }

 }

 leaf-list PDN\_GWInterfaces {

 type enumeration {

 enum S2a ;

 enum S2b ;

 enum S2c ;

 enum S5 ;

 enum S6b ;

 enum Gx ;

 enum S8 ;

 enum SGi ;

 }

 }

 leaf-list eNBInterfaces {

 type enumeration {

 enum S1-MME ;

 enum X2 ;

 }

 }

 leaf-list en-gNBInterfaces {

 type enumeration {

 enum S1-MME ;

 enum X2 ;

 enum Uu ;

 enum F1-C ;

 enum E1 ;

 }

 }

 leaf-list AMFInterfaces {

 type enumeration {

 enum N1 ;

 enum N2 ;

 enum N8 ;

 enum N11 ;

 enum N12 ;

 enum N14 ;

 enum N15 ;

 enum N20 ;

 enum N22 ;

 enum N26 ;

 }

 }

 leaf-list AUSFInterfaces {

 type enumeration {

 enum N12 ;

 enum N13 ;

 }

 }

 leaf-list NEFInterfaces {

 type enumeration {

 enum N29 ;

 enum N30 ;

 enum N33 ;

 }

 }

 leaf-list NRFInterfaces {

 type enumeration {

 enum N27 ;

 }

 }

 leaf-list NSSFInterfaces {

 type enumeration {

 enum N22 ;

 enum N31 ;

 }

 }

 leaf-list PCFInterfaces {

 type enumeration {

 enum N5 ;

 enum N7 ;

 enum N15 ;

 }

 }

 leaf-list SMFInterfaces {

 type enumeration {

 enum N4 ;

 enum N7 ;

 enum N10 ;

 enum N11 ;

 enum S5-C ;

 enum N38 ;

 enum N16 ;

 enum N16a ;

 }

 }

 leaf-list SMSFInterfaces {

 type enumeration {

 enum N20 ;

 enum N21 ;

 }

 }

 leaf-list UDMInterfaces {

 type enumeration {

 enum N8 ;

 enum N10 ;

 enum N13 ;

 enum N21 ;

 }

 }

 leaf-list UPFInterfaces {

 type enumeration {

 enum N4 ;

 }

 }

 leaf-list ng-eNBInterfaces {

 type enumeration {

 enum NG-C ;

 enum Xn-C ;

 enum Uu ;

 }

 }

 leaf-list gNB-CU-CPInterfaces {

 type enumeration {

 enum NG-C ;

 enum Xn-C ;

 enum Uu ;

 enum F1-C ;

 enum E1 ;

 enum X2-C ;

 }

 }

 leaf-list gNB-CU-UPInterfaces {

 type enumeration {

 enum E1 ;

 }

 }

 leaf-list gNB-DUInterfaces {

 type enumeration {

 enum F1-C ;

 }

 }

 }

 leaf-list listOfNETypes {

 type enumeration {

 enum MSC\_SERVER;

 enum SGSN;

 enum MGW;

 enum GGSN;

 enum RNC;

 enum BM\_SC;

 enum MME;

 enum SGW;

 enum PGW;

 enum ENB;

 enum EN\_GNB;

 enum GNB\_CU\_CP;

 enum GNB\_CU\_UP;

 enum GNB\_DU;

 }

 description "Specifies in which type of ManagedFunction the trace should

 be activated. The attribute is applicable only for Trace with

 Signalling Based Trace activation. In case this attribute is not used,

 it carries a null semantic";

 reference "Clause 5.4 of 3GPP TS 32.422 for additional details on the

 allowed values";

 }

 list pLMNTarget {

 key "mcc mnc";

 description "Specifies which PLMN that the subscriber of the session to

 be recorded uses as selected PLMN. PLMN Target might differ from the

 PLMN specified in the Trace Reference";

 reference "Clause 5.9b of 3GPP TS 32.422";

 uses types3gpp:PLMNId;

 }

 leaf-list listOfTraceMetrics {

 when '../jobType = "TRACE\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type string;

 description "Specifies the metrics to be reported.";

 reference "Clause 10 of 3GPP TS 32.422";

 }

 leaf traceReportingConsumerUri {

 when '../traceReportingFormat = "STREAMING"';

 type inet:uri;

 description "URI of the Streaming Trace data reporting MnS consumer

 (a.k.a. streaming target).

 This attribute shall be present if file based trace data reporting is

 supported and traceReportingFormat set to 'file based' or when

 jobType is set to Logged MDT or Logged MBSFN MDT.";

 reference "Clause 5.9 of 3GPP TS 32.422";

 }

 leaf traceCollectionEntityIPAddress {

 when '../traceReportingFormat = "FILE\_BASED" or '

 +'../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

 type union {

 type inet:uri;

 type inet:ip-address;

 }

 mandatory true;

 description "Specifies the address of the Trace Collection Entity when

 the attribute traceReportingFormat is configured for the file-based

 reporting. The attribute is applicable for both Trace and MDT.";

 reference "Clause 5.9 of 3GPP TS 32.422";

 }

 leaf traceDepth {

 when '../jobType = "TRACE\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type enumeration {

 enum MINIMUM;

 enum MEDIUM;

 enum MAXIMUM;

 enum VENDORMINIMUM;

 enum VENDORMEDIUM;

 enum VENDORMAXIMUM;

 }

 default MAXIMUM;

 description "Specifies how detailed information should be recorded in the

 Network Element. The Trace Depth is a paremeter for Trace Session level,

 i.e., the Trace Depth is the same for all of the NEs to be traced in

 the same Trace Session.

 The attribute is applicable only for Trace, otherwise it carries a null

 semantic.";

 reference "Clause 5.3 of 3GPP TS 32.422";

 }

 list traceReference {

 uses TraceReference;

 key "mcc mnc traceId";

 max-elements 1;

 description "A globally unique identifier, which uniquely identifies the

 Trace Session that is created by the TraceJob.

 In case of shared network, it is the MCC and MNC of the Participating

 Operator that request the trace session that shall be provided.

 The attribute is applicable for both Trace and MDT.";

 }

 leaf jobId {

 type string;

 description "Identifier of a TraceJob";

 yext3gpp:inVariant;

 }

 leaf traceReportingFormat {

 type enumeration {

 enum FILE\_BASED;

 enum STREAMING;

 }

 default FILE\_BASED;

 description "Specifies the trace reporting format - streaming trace

 reporting or file-based trace reporting";

 reference "3GPP TS 32.422 clause 5.11";

 }

 list traceTarget {

 key "targetIdType targetIdValue";

 max-elements 1;

 leaf targetIdType {

 type enumeration {

 enum IMSI;

 enum IMEI;

 enum IMEISV;

 enum PUBLIC\_ID;

 enum UTRAN\_CELL;

 enum E\_UTRAN\_CELL;

 enum NG\_RAN\_CELL;

 enum ENB;

 enum RNC;

 enum GNB;

 enum SUPI;

 }

 }

 leaf targetIdValue {

 type string;

 }

 description "Specifies the target object of the Trace and MDT. The

 attribute is applicable for both Trace and MDT. This attribute

 includes the ID type of the target as an enumeration and the ID value.

 The traceTarget shall be public ID in case of a Management Based

 Activation is done to an ScscfFunction. The traceTarget shall be

 cell only in case of the UTRAN cell traffic trace function.

 The traceTarget shall be E-UtranCell only in case of E-UTRAN cell

 traffic trace function.The traceTarget shall be either IMSI or

 IMEI(SV) if the Trace Session is activated to any of the following

 ManagedEntity(ies):

 - HssFunction

 - MscServerFunction

 - SgsnFunction

 - GgsnFunction

 - BmscFunction

 - RncFunction

 - MmeFunction

 The traceTarget shall be IMSI if the Trace Session is activated to a

 ManagedEntity playing a role of ServinGWFunction.

 In case of signaling based Trace/MDT, the traceTarget attribute shall

 be able to carry (IMSI or IMEI(SV)or SUPI), the MDTAreaScope attribute

 shall be able to carry a list of (cell or E-UtranCell or NRCellDU or

 TA/LA/RA).

 In case of management based Immediate MDT, the traceTarget attribute

 shall be null value, the MDTAreaScope attribute shall carry a list of

 (Utrancell or E-UtranCell or NRCellDU).

 In case of management based Logged MDT, the traceTarget attribute

 shall carry an eBs or a RNC or gNBs. The Logged MDT should be initiated

 on the specified eNB or RNC or gNB in traceTarget. The MDTAreaScope

 attribute shall carry a list of (Utrancell or E-UtranCell or NRCellDU or

 TA/LA/RA).

 In case of RLF reporting, or RCEF reporting, the traceTarget

 attribute shall be null value, the MDTAreaScope attribute shall carry

 one or list of eNBs/gNBs";

 reference "3GPP TS 32.422";

 }

 leaf triggeringEvents {

 when '../jobType = "TRACE\_ONLY" or ' +

 '../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type string ;

 mandatory true;

 description "Specifies the triggering event parameter of the trace session.

 The attribute is applicable only for Trace. In case this attribute is

 not used, it carries a null semantic.";

 reference "Clause 5.1 of 3GPP TS 32.422";

 }

 leaf MDTAnonymizationOfData {

 when ../MDTAreaScope ;

 type enumeration {

 enum NO\_IDENTITY;

 enum TAC\_OF\_IMEI;

 }

 default NO\_IDENTITY;

 description "Specifies level of MDT anonymization.";

 reference "3GPP TS 32.422 clause 5.10.12.";

 }

 list MDTAreaConfigurationForNeighCell {

 when '../jobType = "LOGGED\_MDT\_ONLY"';

 key "idx";

 min-elements 1;

 leaf idx { type uint32 ; }

 description "It specifies the area for which UE is requested to perform

 measurement logging for neighbour cells which have list of frequencies.

 If it is not configured, the UE shall perform measurement logging for

 all the neighbour cells.

 Applicable only to NR Logged MDT.";

 reference "3GPP TS 32.422 clause 5.10.26.";

 leaf frequency {

 type string;

 }

 leaf cell {

 type string;

 }

 }

 leaf-list MDTAreaScope {

 type string;

 description "specifies MDT area scope when activates an MDT job.

 For RLF and RCEF reporting it specifies the eNB or list of eNBs where the

 RLF or RCEF reports should be collected.

 List of cells/TA/LA/RA for signaling based MDT or management based Logged

 MDT.

 List of cells for management based Immediate MDT.

 Cell, TA, LA, RA are mutually exclusive.

 One or list of eNBs for RLF and RCEFreporting";

 reference "Clause 5.10.2 of 3GPP TS 32.422";

 }

 leaf MDTCollectionPeriodRrmLte {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "250|500|1000|2000|3000|4000|6000|8000|12000|16000|20000|"

 +"24000|28000|32000|64000";

 }

 units milliseconds;

 description "Specifies the collection period for collecting RRM configured

 measurement samples for M2, M3 in LTE. The attribute is applicable only

 for Immediate MDT. In case this attribute is not used, it carries a

 null semantic.";

 reference "Clause 5.10.20 of 3GPP TS 32.422";

 }

 leaf MDTCollectionPeriodM6Lte {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "1024|2048|5120|10240";

 }

 units milliseconds;

 description "Specifies the collection period for the Packet Delay

 measurement (M6) for MDT taken by the eNB. The attribute is applicable

 only for Immediate MDT. In case this attribute is not used,

 it carries a null semantic.";

 reference "Clause 5.10.32 of TS 32.422 ";

 }

 leaf MDTCollectionPeriodM7Lte {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint16 {

 range 1..60 ;

 }

 description "It specifies the collection period for the Data Volume (M6)

 and Throughput measurements (M7) for UMTS MDT taken by RNC. The

 attribute is applicable only for Immediate MDT. In case this attribute

 is not used, it carries a null semantic.";

 reference "Clause 5.10.22 of TS 32.422 .";

 }

 leaf MDTCollectionPeriodRrmUmts {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "1024|1280|2048|2560|5120|"

 +"10240|60000";

 }

 units milliseconds;

 description "Specifies the collection period for collecting RRM configured

 measurement samples for M3, M4, M5 in UMTS. The attribute is applicable

 only for Immediate MDT. In case this attribute is not used, it carries

 a null semantic";

 reference "Clause 5.10.21 of 3GPP TS 32.422";

 }

 leaf MDTCollectionPeriodRrmNR {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "1024|2048|5120|10240|60000";

 }

 units milliseconds;

 description "Specifies the collection period for collecting RRM

 configured measurement samples for M4, M5 in NR. The attribute is

 applicable only for Immediate MDT. In case this attribute is not

 used, it carries a null semantic.";

 reference "Clause 5.10.30 of 3GPP TS 32.422";

 }

 leaf MDTCollectionPeriodM6NR {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type enumeration {

 enum 120ms;

 enum 240ms;

 enum 480ms;

 enum 640ms;

 enum 1024ms;

 enum 2048ms;

 enum 5120ms;

 enum 10240ms;

 enum 20480ms;

 enum 40960ms;

 enum 1min;

 enum 6min;

 enum 12min;

 enum 30min;

 }

 description "It specifies the collection period for the Packet Delay

 measurement (M6) for NR MDT taken by the gNB. The attribute is

 applicable only for Immediate MDT. In case this attribute is not used,

 it carries a null semantic.";

 reference "clause 5.10.34 of TS 32.422";

 }

 leaf MDTCollectionPeriodM7NR {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "1..60";

 }

 description "It specifies the collection period for the Packet Loss Rate

 measurement (M7) for NR MDT taken by the gNB. The attribute is

 applicable only for Immediate MDT. In case this attribute is not used,

 it carries a null semantic.";

 reference "clause 5.10.35 of TS 32.422";

 }

 leaf MDTEventListForTriggeredMeasurement {

 when '../jobType = "LOGGED\_MDT\_ONLY"';

 type enumeration {

 enum OUT\_OF\_COVERAGE ;

 enum A2\_EVENT ;

 }

 mandatory true;

 description "Specifies event types for event triggered measurement in the

 case of logged NR MDT. Each trace session may configure at most one

 event. The UE shall perform logging of measurements only upon certain

 condition being fulfilled:

 - Out of coverage.

 - A2 event.";

 reference "Clause 5.10.28 of 3GPP TS 32.422";

 }

 leaf MDTEventThreshold {

 type int64;

 description "Specifies the threshold which should trigger the reporting

 in case A2 event reporting in LTE or 1F/1l event in UMTS. The attribute

 is applicable only for Immediate MDT and when reportingTrigger is

 configured for A2 event in LTE or 1F event or 1l event in UMTS. In

 case this attribute is not used, it carries a null semantic.";

 reference "Clauses 5.10.7 and 5.10.7a of 3GPP TS 32.422";

 }

 leaf MDTListOfMeasurements {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"';

 type int64;

 mandatory true;

 description "It specifies the UE measurements that shall be collected in

 an Immediate MDT job. The attribute is applicable only for Immediate MDT.

 In case this attribute is not used, it carries a null semantic.";

 reference "3GPP TS 32.422 clause 5.10.3";

 }

 leaf MDTLoggingDuration {

 when '../jobType = "LOGGED\_MDT\_ONLY" or'

 + ' ../jobType = "LOGGED\_MBSFN\_MDT"';

 type uint32 {

 range "600|1200|2400|3600|5400|7200";

 }

 units seconds;

 mandatory true;

 description "Specifies how long the MDT configuration is valid at the

 UE in case of Logged MDT. The attribute is applicable only for

 Logged MDT and Logged MBSFN MDT. In case this attribute is not used, it

 carries a null semantic.";

 reference "5.10.9 of 3GPP TS 32.422";

 }

 leaf MDTLoggingInterval {

 when '../jobType = "LOGGED\_MDT\_ONLY" or'

 + ' ../jobType = "LOGGED\_MBSFN\_MDT"';

 type uint32 {

 range "1280|2560|5120|10240|20480|"

 +"30720|40960|61440";

 }

 units milliseconds;

 mandatory true;

 description "Specifies the periodicty for Logged MDT. The attribute is

 applicable only for Logged MDT and Logged MBSFN MDT. In case this

 attribute is not used, it carries a null semantic";

 reference "5.10.8 of 3GPP TS 32.422";

 }

 leaf MDTLoggingEventThreshold {

 when '../jobType = "LOGGED\_MDT\_ONLY" or'

 + ' ../jobType = "LOGGED\_MBSFN\_MDT"';

 type uint32 {

 range "0..127";

 }

 description "It specifies the threshold which should trigger

 the reporting in case of event based reporting of logged NR MDT.

 The attribute is applicable only for Logged MDT and when MDTReportType

 is configured for event triggered reporting and when

 MDTEventListForTriggeredMeasurement is configured for L1 event.

 In case this attribute is not used, it carries a null semantic.";

 reference "clause 5.10.36 of TS 32.422";

 }

 leaf MDTLoggedHysteresis {

 when '../jobType = "LOGGED\_MDT\_ONLY" or '

 + '../jobType = "LOGGED\_MBSFN\_MDT"';

 type uint32 {

 range "0..30";

 }

 description "It specifies the hysteresis used within the entry and leave

 condition of the L1 event based reporting of logged NR MDT.

 The attribute is applicable only for Logged MDT, when MDTReportType

 is configured for event triggered reporting and when

 MDTEventListForTriggeredMeasurement is configured for L1 event.

 In case this attribute is not used, it carries a null semantic.";

 reference "clause 5.10.37 of TS 32.422";

 }

 leaf MDTLoggedTimeToTrigger {

 when '../jobType = "LOGGED\_MDT\_ONLY" or '

 + '../jobType = "LOGGED\_MBSFN\_MDT"';

 type int32 ;

 description "It specifies the threshold which should trigger

 the reporting in case of event based reporting of logged NR MDT.

 The attribute is applicable only for Logged MDT, when MDTReportType

 is configured for event triggered reporting and when

 MDTEventListForTriggeredMeasurement is configured for L1 event.

 In case this attribute is not used, it carries a null semantic.";

 reference "clauses 5.10.38 of TS 32.422";

 }

 leaf-list MDTMBSFNAreaList {

 when '../jobType = "LOGGED\_MBSFN\_MDT"';

 type string;

 min-elements 1;

 max-elements 8;

 description "The MBSFN Area consists of a MBSFN Area ID and Carrier

 Frequency (EARFCN). The target MBSFN area List can have up to 8 entries.

 This parameter is applicable only if the job type is Logged MBSFN MDT.";

 reference "5.10.25 of 3GPP TS 32.422";

 }

 leaf MDTMeasurementPeriodLTE {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "1024|1280|2048|2560|5120|"

 +"10240|60000";

 }

 units milliseconds;

 mandatory true;

 description "It specifies the measurement period for the Data Volume and

 Scheduled IP throughput measurements for MDT taken by the eNB.

 The attribute is applicable only for Immediate MDT. In case this

 attribute is not used, it carries a null semantic.";

 reference "Clause 5.10.23 of 3GPP TS 32.422";

 }

 leaf MDTMeasurementPeriodUMTS {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + 'or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint32 {

 range "250|500|1000|2000|3000|4000|6000|8000|12000|16000|20000|"

 +"24000|28000|32000|64000";

 }

 units milliseconds;

 mandatory true;

 description "It specifies the measurement period for the Data Volume and

 Throughput measurements for MDT taken by RNC.

 The attribute is applicable only for Immediate MDT. In case this

 attribute is not used, it carries a null semantic.";

 reference "Clause 5.10.22 of 3GPP TS 32.422";

 }

 leaf MDTMeasurementQuantity {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint64 ;

 mandatory true;

 description "It specifies the measurements that are collected in an MDT

 job for a UMTS MDT configured for event triggered reporting.";

 reference "Clause 5.10.15 of 3GPP TS 32.422";

 }

 leaf MDTM4ThresholdUmts {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type uint16 {

 range 0..31 ;

 }

 description "It specifies the threshold which should trigger

 the reporting in case of event-triggered periodic reporting for M4

 (UE power headroom measurement) in UMTS. In case this attribute is

 not used, it carries a null semantic.";

 reference "3GPP TS 32.422 clause 5.10.A";

 }

 list MDTPLMList {

 when '../jobType = "LOGGED\_MDT\_ONLY"';

 key "mcc mnc";

 uses types3gpp:PLMNId;

 min-elements 1;

 max-elements 16;

 description "It indicates the PLMNs where measurement collection, status

 indication and log reporting is allowed.";

 reference "Clause 5.10.24 of 3GPP TS 32.422";

 }

 leaf MDTPositioningMethod {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

 type enumeration {

 enum GNSS;

 enum E\_CELL\_ID;

 }

 mandatory true;

 description "It specifies what positioning method should be used in the

 MDT job.";

 reference "Clause 5.10.19 of 3GPP TS 32.422";

 }

 leaf MDTReportAmount {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' and ../MDTReportingTrigger = "PERIODICAL"';

 type union {

 type uint32 {

 range "1|4|8|16|32|64" ;

 }

 type enumeration {

 enum INFINITY;

 }

 }

 mandatory true;

 description "It specifies the number of measurement reports that shall be

 taken for periodic reporting while the UE is in connected.

 The attribute is applicable only for Immediate MDT and when

 MDTReportingTrigger is configured for periodical measurements. In

 case this attribute is not used, it carries a null semantic.";

 reference "Clause 5.10.6 of 3GPP TS 32.422";

 }

 leaf MDTReportingTrigger {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"';

 type enumeration {

 enum PERIODICAL;

 enum A2\_FOR\_LTE;

 enum 1F\_FOR\_UMTS;

 enum 1I\_FOR\_UMTS\_MCPS\_TDD;

 enum A2\_TRIGGERED\_PERIODIC\_FOR\_LTE;

 enum ALL\_CONFIGURED\_RRM\_FOR\_LTE;

 enum ALL\_CONFIGURED\_RRM\_FOR\_UMTS;

 }

 description "It specifies whether periodic or event based measurements

 should be collected.

 The attribute is applicable only for Immediate MDT and when the

 MDTListOfMeasurements is configured for M1 (for both UMTS and LTE)

 or M2 (only for UMTS). In case this attribute is not used, it carries

 a null semantic.";

 reference "Clause 5.10.4 of 3GPP TS 32.422";

 }

 leaf MDTReportInterval {

 when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

 + ' and ../MDTReportingTrigger = "PERIODICAL"';

 type uint32 {

 range "120|240|250|480|500|640|1000|1024|2000|2048|3000|4000|"

 +"5120|6000|8000|10240|12000|16000|20000|"

 +"24000|28000|32000|60000|64000|"

 +"360000|720000|1800000|3600000";

 }

 units milliseconds;

 mandatory true;

 description "It specifies the interval between the periodical measurements

 that shall be taken when the UE is in connected mode.

 The attribute is applicable only for Immediate MDT and when

 MDTReportingTrigger is configured for periodical measurements. In case

 this attribute is not used, it carries a null semantic.";

 reference "5.10.5 of 3GPP TS 32.422";

 }

 leaf MDTReportType {

 when '../jobType = "LOGGED\_MDT\_ONLY"';

 type enumeration {

 enum PERIODICAL;

 enum EVENT\_TRIGGERED;

 }

 mandatory true;

 description "It specifies report type for logged NR MDT";

 reference "Clause 5.10.27 of 3GPP TS 32.422";

 }

 leaf MDTSensorInformation {

 type bits {

 bit BAROMETRIC\_PRESSURE;

 bit UE\_SPEED;

 bit UE\_ORIENTATION;

 }

 default "";

 description "It specifies which sensor information shall be included in

 logged NR MDT and immediate NR MDT measurement if they are available.

 The following sensor measurement can be included or excluded for

 the UE.";

 reference "Clause 5.10.29 of 3GPP TS 32.422";

 }

 leaf MDTTraceCollectionEntityID {

 when '../jobType = "LOGGED\_MDT\_ONLY" or '

 + '../jobType = "LOGGED\_MBSFN\_MDT"';

 type uint8;

 mandatory true;

 description "It specifies the TCE Id which is sent to the UE in

 Logged MDT.";

 reference "Clause 5.10.11 of 3GPP TS 32.422";

 }

 }

 grouping TraceSubtree {

 description "Contains classes that manage Tracing.

 Should be used in all classes (or classes inheriting from)

 - SubNnetwork

 - ManagedElement

 - ManagedFunction

 If a YANG module wants to augment these classes/list/groupings they must

 augment all user classes!";

 list TraceJob {

 description "Represents the Trace Control and Configuration parameters of a

 particular Trace Job (see TS 32.421 and TS 32.422 for details).

 It can be name-contained by SubNetwork, ManagedElement, ManagedFunction

 or NetworkSliceSubnet.

 To activate Trace Jobs, a MnS consumer has to create TraceJob object

 instances on the MnS producer. A MnS consumer can activate a Trace Job

 for another MnS consumer since it is not required the value of

 traceCollectionEntityAddress or streamingTraceConsumerUri to be his

 own.

 When a MnS consumer wishes to deactivate a Trace Job, the MnS consumer

 shall delete the corresponding TraceJob instance.

 For details of management Trace Job activation/deactivation see clause

 4.1.1.1.2 of TS 32.422.

 The attribute traceReference specifies a globally unique ID and

 identifies a Trace session. One Trace Session may be activated to

 multiple Network Elements. The traceReference is populated by the

 consumer that makes the request for a Trace Session.

 The attribute tjJobType specifies the kind of data to collect.

 Dependent on the selected type various parameters shall be available.

 The attributes tjJobType, tjTraceReference,

 tjTraceCollectionEntityAddress and tjTraceReportingFormat are

 mandatory for all job types. If streaming reporting is selected

 for tjTraceReportingFormat, tjStreamingTraceConsumerURI shall be

 present additionally. The attribute tjPLMNTarget shall be present

 if trace activation method is management based.

 For the different job types the attributes are differentiated as follows:

 - In case of TRACE\_ONLY additionally the following attributes shall be

 available: listOfNeTypes, traceDepth, traceTarget and

 triggeringEvent.

 For this case the optional attribute listOfInterfaces allows to

 specify the interfaces to be recorded.

 - In case of IMMEDIATE\_MDT\_ONLY additionally the following attributes

 shall be available:

 - traceTarget

 - MDTAnonymizationOfData,

 - MDTListOfMeasurements,

 - MDTCollectionPeriodRrmUmts (conditional for M3, M4 and M5 in UMTS),

 - MDTMeasurementPeriodUMTS (conditional for M6 and M7 in UMTS),

 - MDTCollectionPeriodRrmLte (conditional for M2 and M3 in LTE),

 - MDTMeasurementPeriodLTE (conditional for M4 and M5 in LTE),

 - MDTCollectionPeriodM6Lte (conditional for M6 in LTE),

 - MDTCollectionPeriodM7Lte (conditional for M7 in LTE),

 - MDTCollectionPeriodRrmNR (conditional for M4 and M5 in NR),

 - MDTCollectionPeriodM6NR (conditional for M6 in NR),

 - MDTCollectionPeriodM7NR (conditional for M7 in NR),

 - MDTReportInterval (conditional for M1 in LTE or NR and M1/M2 in

 UMTS),

 - MDTReportAmount (conditional for M1 in LTE or NR and M1/M2 in

 UMTS),

 - MDTReportingTrigger (conditional for M1 in LTE or NR and M1/M2 in

 UMTS),

 - MDTEventThreshold (conditional for A2 event reporting or A2 event

 triggered periodic reporting),

 - MDTMeasurementQuantity (conditional for 1F event reporting).

 For this case the optional attribute MDTAreaScope allows to specify

 the area in terms of cells or Tracking Area/Routing Area/Location area

 where the MDT data collection shall take place and the optional

 attributes MDTPositioningMethod, MDTSensorInformation allow to

 specify the positioning methods to use or the sensor information to

 include.

 - In case of IMMEDIATE\_MDT\_AND\_TRACE both additional attributes of

 TRACE\_ONLY and IMMEDIATE\_MDT\_ONLY shall apply.

 - In case of LOGGED\_MDT\_ONLY additionally the following attributes

 shall be available: traceTarget, MDTAnonymizationOfData,

 MDTTraceCollectionEntityID, MDTLoggingInterval,

 MDTLoggingDuration, MDTReportType,

 MDTEventListForTriggeredMeasurements.

 For this case the optional attribute MDTAreaScope allows to specify

 the area in terms of cells or Tracking Area/Routing Area/Location area

 where the MDT data collection shall take place, the optional attribute

 MDTPLMNList allows to specify the PLMNs where measurement collection,

 status indication and log reporting is allowed, the optional attribute

 MDTAreaConfigurationForNeighCell allows to specify the area for

 which UE is requested to perform measurements logging for neighbour

 cells which have list of frequencies and the optional attribute

 MDTSensorInformation allows to specify the sensor information to

 include.

 - In case of RLF\_REPORT\_ONLY and RCEF\_REPORT\_ONLY additionally the

 attribute traceTarget shall be available, the optional attribute

 MDTAreaScope allows to specify the eNB or list of eNBs or gNB or

 list of gNBs where the reports should be collected.

 - In case of LOGGED\_MBSFN\_MDT additionally the following attributes

 shall be available: MDTAnonymizationOfData, MDTLoggingInterval,

 MDTLoggingDuration, MDTMBSFNAreaList.

 Creation and deletion of TraceJob instances by MnS consumers is optional;

 when not supported, the TraceJob instances may be created and deleted by

 the system or be pre-installed.";

 key id;

 uses top3gpp:Top\_Grp ;

 container attributes {

 uses TraceJobGrp ;

 }

 uses files3gpp:FilesSubtree {

 if-feature FilesUnderTraceJob;

 }

 }

 }

}

<CODE ENDS>

\*\*\* END OF CHANGE 1 \*\*\*