ETSI TS 132 446 V10.7.0 (2012-09)

Technical Specification

Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Trace Management Integration Reference Point (IRP); Solution Set (SS) definitions (3GPP TS 32.446 version 10.7.0 Release 10)

Reference

RTS/TSGS-0532446va70

Keywords

GSM,LTE,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part of a TS-family covering the 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; as identified below:

32.441 "Trace Management Integration Reference Point (IRP): Requirements".

32.442 "Trace Management Integration Reference Point (IRP): Information Service (IS)".

32.446 "Trace Management Integration Reference Point (IRP): Solution Set (SS) definitions".

The present document is part of a TS-family which describes the information service necessary for the Telecommunication Management (TM) of 3G systems. The TM principles and TM architecture are specified in 3GPP TS 32.101 [2] and 3GPP TS 32.102 [3].

Trace provides very detailed information on call level for a specific subscriber or MS. This data is an additional information source to Performance Measurements and allows deeper investigations in problems solving or in case of optimization.

1 Scope

The present document specifies the Solution Set definitions for the IRP whose semantics are specified in Trace Management IRP: Information Service (3GPP TS 32.442 [5]).

This Solution Set specification is related to 3GPP TS 32.442 V10. 6.X.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 32.101: "Telecommunication management; Principles and high level requirements".

[3] 3GPP TS 32.102: "Telecommunication management; Architecture".

[4] 3GPP TS 32.150: "Telecommunication management; Integration Reference Point (IRP) Concept and definitions".

[5] 3GPP TS 32.442: "Telecommunication management; Trace Management Integration Reference Point (IRP): Information Service (IS)".

[6] 3GPP TS 32.441: "Telecommunication management; Trace Management Integration Reference Point (IRP): Requirements".

[7] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP): Requirements".

[8] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management; Information Service (IS)"

[9] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".

[10] 3GPP TS 32.306: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Solution Set definitions"

[11] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management: Information Service (IS)".

[12] OMG TC Document telecom/98-11-01: "OMG Notification Service". http://www.omg.org/technology/documents/

[13] 3GPP TS 32.342: "Telecommunication management; File Transfer (FT) Integration Reference Point (IRP): Information Service (IS)".

[14] W3C REC-xml-20001006: "Extensible Markup Language (XML) 1.0 (Second Edition)".

[15] W3C REC-xmlschema-0-20010502: "XML Schema Part 0: Primer".

[16] W3C REC-xmlschema-1-20010502: "XML Schema Part 1: Structures".

[17] W3C REC-xmlschema-2-20010502: "XML Schema Part 2: Datatypes".

[18] W3C REC-xml-names-19990114: "Namespaces in XML".

[19] 3GPP TS 32.421: "Telecommunication management; Subscriber and Equipment Trace; Concept and requirements".

[20] 3GPP TS 32.422: "Telecommunication management; Subscriber and equipment trace; Trace control and configuration management'

[21] W3C SOAP 1.1 specification (http://www.w3.org/TR/2000/NOTE-SOAP-20000508/)

[22] W3C XPath 1.0 specification (http://www.w3.org/TR/1999/REC-xpath-19991116)

[23] W3C WSDL 1.1 specification (http://www.w3.org/TR/2001/NOTE-wsdl-20010315)

[24] W3C SOAP 1.2 specification (http://www.w3.org/TR/soap12-part1/)

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1], 3GPP TS 32.101 [2], 3GPP TS 32.102 [3], 3GPP TS 32.150 [4] apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

IRP document version number string (or "IRPVersion"): See 3GPP TS 32.311 [7].

IRP: See 3GPP TS 32.101 [2].

IRPAgent: See 3GPP TS 32.102 [3].

IRPManager: See 3GPP TS 32.102 [3].

XML file: file containing an XML document

XML document: composed of the succession of an optional XML declaration followed by a root XML element, see [14].

XML declaration: it specifies the version of XML being used, see [14].

XML element: has a type, is identified by a name, may have a set of XML attribute specifications and is either composed of the succession of an XML start-tag followed by the XML content of the XML element followed by an XML end-tag, or composed simply of an XML empty-element tag; each XML element may contain other XML elements, see [14].

empty XML element: having an empty XML content; an empty XML element still possibly has a set of XML attribute specifications; an empty XML element is either composed of the succession of an XML start-tag directly followed by an XML end-tag, or composed simply of an XML empty-element tag, see [14].

XML content (of an XML element): empty if the XML element is simply composed of an XML empty-element tag; otherwise the part, possibly empty, of the XML element between its XML start-tag and its XML end-tag, see [14].

XML start-tag: the beginning of a non-empty XML element is marked by an XML start-tag containing the name and the set of XML attribute specifications of the XML element, see [14].

XML end-tag: the end of a non-empty XML element is marked by an XML end-tag containing the name of the XML element, see [14].

XML empty-element tag: composed simply of an empty-element tag containing the name and the set of XML attribute specifications of the XML element, see [14].

XML attribute specification: has a name and a value, see [14].

DTD: defines structure and content constraints to be respected by an XML document to be valid with regard to this DTD, see [14].

XML schema: more powerful than a DTD, an XML schema defines structure and content constraints to be respected by an XML document to conform with this XML schema; through the use of XML namespaces several XML schemas can be used together by a single XML document; an XML schema is itself also an XML document that shall conform with the XML schema for XML schemas, see [15], [16] and [17].

XML namespace: enables qualifying element and attribute names used in XML documents by associating them with namespaces identified by different XML schemas, see [15], [16] and [17].

XML complex type: defined in an XML schema; cannot be directly used in an XML document; can be the concrete type or the derivation base type for an XML element type or for another XML complex type; ultimately defines constraints for an XML element on its XML attribute specifications and/or its XML content, see [15], [16] and [17].

XML element type: declared by an XML schema; can be directly used in an XML document; as the concrete type of an XML element, directly or indirectly defines constraints on its XML attribute specifications and/or its XML content; can also be the concrete type or the derivation base type for another XML element type, see [15], [16] and [17].

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1], in 3GPP TS 32.101 [2], 3GPP TS 32.102 [3], 3GPP TS 32.150 [4], and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

CM	Configuration Management
CORBA	Common Object Request Broker Architecture
DN	Distinguished Name
EM	Element Manager
IDL	Interface Definition Language
IRP	Integration Reference Point
Itf-N	Interface N
IS	Information Service
MOC	Managed Object Class
NE	Network Element
OMG	Object Management Group
SS	Solution Set
TS	Technical Specification

4 Solution Set definitions

This specification defines the following 3GPP Trace Management IRP Solution Set definitions:

Annex A provides the CORBA Solution Set.
Annex B provides the XML definitions.
Annex C provides the SOAP Solution Set.

Annex A (normative): CORBA Solution Set

This annex contains the CORBA Solution Set for the IRP whose semantics is specified in Trace Management IRP: Information Service (3GPP TS 32.442 [5]).

A.1 Architectural features

The overall architectural feature of Trace Management IRP is specified in 3GPP TS 32.442 [5].

A.1.1 Syntax for Distinguished Names

The format of a Distinguished Name is defined in 3GPP TS 32.300 [9].

The version of this IRP is represented as a string (see also clause 3 for versions).

A.1.2 Notification Services

Notifications are sent according to the Notification IRP: CORBA SS (see 3GPP TS 32.306 [10]).

The contents of the TraceIRP notifications are defined in the present document.

A.1.3 Push and Pull Style

OMG Notification Service defines two styles of interaction. One is called push style. In this style, IRPAgent pushes notifications to IRPManager as soon as they are available. The other is called pull style. In this style, IRPAgent keeps the notifications till IRPManager requests for them.

This CORBA SS specifies that support of Push style is Mandatory (M) and that support of Pull style is Optional (O).

A.1.4 Support multiple notifications in one push operation

For efficiency reasons, IRPAgent may send multiple notifications using one single push operation. To pack multiple notifications into one push operation, IRPAgent may wait and not invoke the push operation as soon as notifications are available. To avoid IRPAgent to wait for an extended period of time that is objectionable to IRPManager, IRPAgent shall implement an IRPAgent wide timer configurable by administrator. On expiration of this timer, IRPAgent shall invoke push if there is at least one notification to be conveyed to IRPManager. This timer is re-started after each push invocation.

A.1.5 Trace Management Notification Interface

OMG CORBA Notification push operation is used to realise the notification of `TraceIRP Notifications`. All the notifications in this interface are implemented using this `push_structured_event` method.

A.1.5.1 Method `push` (M)

```
module CosNotifyComm {
…
Interface SequencePushConsumer : NotifyPublish {
void push_structured_events(
in CosNotification::EventBatch notifications)
            raises( CosEventComm::Disconnected);
…
}; // SequencePushConsumer
…
}; // CosNotifyComm
```

NOTE 1: The push_structured_events method takes an input parameter of type EventBatch as defined in the OMG CosNotification module (OMG Notification Service [12]). This data type is the same as a sequence of Structured Events. Upon invocation, this parameter will contain a sequence of Structured Events being delivered to IRPManager by IRPAgent to which it is connected.

NOTE 2: The maximum number of events that will be transmitted within a single invocation of this operation is controlled by IRPAgent wide configuration parameter.

NOTE 3: The amount of time the supplier (IRPAgent) of a sequence of Structured Events will accumulate individual events into the sequence before invoking this operation is controlled by IRPAgent wide configuration parameter as well.

NOTE 4: IRPAgent may push EventBatch with only one Structured Event..

A.2 Mapping

A.2.1 Operation and Notification mapping

TraceIRP: IS 3GPP TS 32.442 [5] defines semantics of operation and notification visible across the TraceIRP. Table A.2.1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table A.2.1: Mapping from IS Operations and Notification to SS equivalents

IS Operations/ notification 3GPP TS 32.442 [5]	SS Method	Qualifier
activateTraceJob	activate_trace_job	M
deactivateTraceJob	deactivate_trace_job	M
listTraceJob	list_trace_job	M
listActivatedTraceJob	list_activated_trace_job	M
notifyTraceRecordingSessionFailure	push_structured_events(See subclause A.1.5.1)	O
notifyTraceSessionLocalActivation	push_structured_events (See subclause A.1.5.1)	O

A.2.2 Operation parameter mapping

The TraceIRP: IS 3GPP TS 32.442 [5] defines semantics of parameters carried in operations across the TraceIRP. The following tables indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table A.2.2.1: Mapping from IS `activateTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iocInstance	KernelCmConstDefs::DN moInstance	M
listOfInterfaces	TraceIRPConstDefs::ListOfInterfaces list_of_interfaces	O
listOfNeTypes	TraceIRPConstDefs::ListOfNeTypes list_of_ne_types	CM
traceDepth	TraceIRPConstDefs::TraceDepth trace_depth	M
traceReference	TraceIRPConstDefs::TraceReference trace_reference	M
traceTarget	TraceIRPConstDefs::TraceTarget trace_target	M
triggeringEvent	TraceIRPConstDefs:: TriggeringEvent triggering_event	CO
traceCollectionEntityAddress	TraceIRPConstDefs::TraceCollectionEntityAddress trace_collection_entity_address	CM
jobType	TraceIRPConstDefs:: JobType job_type	M
mdtAreaScope	TraceIRPConstDefs::DNSet	CM
listOfMeasurements	TraceIRPConstDefs:: ListOf Measurements list_of_measurements	CM
reportingTrigger	TraceIRPConstDefs::ReportingTrigger reporting_trigger	CM
reportInterval	TraceIRPConstDefs::ReportInterval report_interval	CM
reportAmount	TraceIRPConstDefs::ReportAmount report_amount	CM
eventThreshold	TraceIRPConstDefs::EventThreshold event_threshold	CM
loggingInterval	TraceIRPConstDefs::LoggingInterval logging_interval	CM
loggingDuration	TraceIRPConstDefs::LoggingDuration logging_duration	CM
anonymizationOfMDTData	TraceIRPConstDefs: AnonymizationOfMDTData anonymization_Of_MDT_Data	CM
measurementQuantity	TraceIRPConstDefs::MeasurementQuantity measurement_quantity	CM
unsupportedList	TraceIRPConstDefs:: UnsupportedList unsupportedList	M
status	Return value of type TraceIRPConstDefs::Result Exception: ActivateTraceJob, InvalidTraceDepth, InvalidTraceTarget, NotUniqueTraceReference ManagedGenericIRPSystem::InvalidParameter, ManagedGenericIRPSystem::ParameterNotSupported	M

Table A.2.2.2: Mapping from IS `deactivateTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReference	TraceIRPConstDefs:: TraceReference trace_reference	M
traceTarget	TraceIRPConstDefs::TraceTarget trace_target	M
status	Return value of type TraceIRPConstDefs::Result Exception: DeactivateTraceJob, NotUniqueTraceReference	M
traceRecordingSessionReference	TraceIRPConstDefs::TraceRecordingSessionReference trace_recording_session_reference	CM

Table A.2.2.3: Mapping from IS `listTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReference	TraceIRPConstDefs::TraceReference trace_reference	M
status	Return value of type TraceIRPConstDefs::Result Exception: ListTraceJob, NotUniqueTraceReference	M
iocInstance	KernelCmConstDefs::DN moInstance	M
listOfInterfaces	TraceIRPConstDefs::ListOfInterfaces list_of_interfaces	O
traceDepth	TraceIRPConstDefs::TraceDepth trace_depth	M
traceRecordingSessionReference	TraceIRPConstDefs::TraceRecordingSessionReference trace_recording_session_reference	CM
traceTarget	TraceIRPConstDefs::TraceTarget trace_target	M
jobType	TraceIRPConstDefs::JobType job_type	M
mdtAreaScope	TraceIRPConstDefs::DNSet	CM
listOfMeasurements	TraceIRPConstDefs:: ListOf Measurements list_of_measurements	CM
reportingTrigger	TraceIRPConstDefs::ReportingTrigger reporting_trigger	CM
reportInterval	TraceIRPConstDefs::ReportInterval report_interval	CM
reportAmount	TraceIRPConstDefs::ReportAmount report_amount	CM
eventThreshold	TraceIRPConstDefs::EventThreshold event_threshold	CM
loggingInterval	TraceIRPConstDefs::LoggingInterval logging_interval	CM
loggingDuration	TraceIRPConstDefs::LoggingDuration logging_duration	CM
triggeringEvent	TraceIRPConstDefs::TriggeringEvent triggering_event	O
traceCollectionEntityAddress	TraceIRPConstDefs::TraceCollectionEntityAddress trace_collection_entity_address	CM
anonymizationOfMDTData	TraceIRPConstDefs: AnonymizationOfMDTData anonymization_Of_MDT_Data	CM
measurementQuantity	TraceIRPConstDefs::MeasurementQuantity measurement_quantity	CM

Table A.2.2.4: Mapping from IS `listActivatedTraceJobs` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReferenceList	TraceMIRPConstDefs::TraceReferenceList trace_reference_list	M
status	Return value of type TraceIRPConstDefs::Result	M

A.2.3 Notification parameter mapping

The TraceIRP: IS 3GPP TS 32.442 [5] defines semantics of parameters carried in notifications. The following table indicates the mapping of these parameters to their OMG CORBA Structured Event (defined in OMG Notification Service [12]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [12], is:

```
Header
      Fixed Header
           domain_name
           type_name
           event_name
      Variable Header
Body
      filterable_body_fields
      remaining_body
```

The following tables list all OMG Structured Event attributes in the second column. The first column identifies the TraceIRP: IS 3GPP TS 32.442 [5] defined notification parameters.

Table A.2.3.1: Mapping for `notifyTraceRecordingSessionFailure`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS attribute.	domain_name	M	It carries the IRP document version number string. See subclause 3.1. It indicates the syntax and semantics of the Structured Event as defined by the present document.
notificationType	type_name	M	This is constant string "notifyTraceRecordingSessionFailure".
There is no corresponding IS attribute.	event_name	M	It carries no information.
There is no corresponding IS attribute.	Variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. Name of this NV pair is the MANAGED_OBJECT_INSTANCE of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
notificationId	One NV pair of remaining_body	M	Name of NV pair is the NOTIFICATION_ID of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the EVENT_TIME of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the SYSTEM_DN of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
traceRecordingSessionReference	One NV pair of remaining_body	O	Name of NV pair is the TRACE_RECORDING_SESSION_REFERENCE of TraceIRPNotifications::notifyTraceRecordingSessionFailure. Value of NV pair is TraceRecordingSessionReference of module TraceIRPConstDefs.
traceReference	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_REFERENCE of TraceIRPNotifications::notifyTraceRecordingSessionFailure. Value of NV pair is TraceReference of module TraceIRPConstDefs.
reason	One NV pair of remaining_body	O	Name of NV pair is the REASON of traceIRPNotifications:: notifyTraceRecordingSessionFailure. Value of NV pair is a string.

Table A.2.3.2: Mapping for `notifyTraceSessionLocalActivation`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS attribute.	domain_name	M	It carries the IRP document version number string. See subclause 3.1. It indicates the syntax and semantics of the Structured Event as defined by the present document.
notificationType	type_name	M	This is constant string "notifyThresholdMonitorObjectCreation".
There is no corresponding IS attribute.	event_name	M	It carries no information.
There is no corresponding IS attribute.	Variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. Name of this NV pair is the MANAGED_OBJECT_INSTANCE of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
notificationId	One NV pair of remaining body	M	Name of NV pair is the NOTIFICATION_ID of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the EVENT_TIME of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the SYSTEM_DN of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
traceReference	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_REFERENCE of module TraceIRPNotifications::notifyTraceSessionLocalActivation. Value of NV pair is TraceReference of module TraceIRPConstDefs.
traceTarget	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_TARGET of module TraceIRPNotifications::notifyTraceSessionLocalActivation. Value of NV pair is TraceTarget of module TraceIRPConstDefs.
iOCInstance	One NV pair of filterable_body_fields	M	Name of NV pair is the IOC_INSTANCE of module TraceIRPNotifications::notifyTraceSessionLocalActivation. Value of NV pair is MOClassName of module TraceIRPConstDefs.

Table A.2.3.3: Mapping for `notifyTraceSessionIdentities`

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS attribute.	domain_name	M	It carries the IRP document version number string. See subclause 3.1. It indicates the syntax and semantics of the Structured Event as defined by the present document.
notificationType	type_name	M	This is constant string "notifyTraceSessionIdentities".
There is no corresponding IS attribute.	event_name	M	It carries no information.
There is no corresponding IS attribute.	Variable Header		
objectClass, objectInstance	One NV pair of filterable_body_fields	M	NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. Name of this NV pair is the MANAGED_OBJECT_INSTANCE of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
notificationId	One NV pair of remaining body	M	Name of NV pair is the NOTIFICATION_ID of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
eventTime	One NV pair of filterable_body_fields	M	Name of NV pair is the EVENT_TIME of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
systemDN	One NV pair of filterable_body_fields	M	Name of NV pair is the SYSTEM_DN of interface AttributeNameValue of module NotificationIRPConstDefs. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.306 [10]).
traceReference	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_REFERENCE of module TraceIRPNotifications::notifyTraceSessionLocalActivation . Value of NV pair is TraceReference of module TraceIRPConstDefs.
traceRecordingSessionReference	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_RECORDING_SESSION_REFERENCE of TraceIRPNotifications::notifyTraceSessionIdentities Value of NV pair is TraceReference of module TraceIRPConstDefs.
traceTarget	One NV pair of filterable_body_fields	M	Name of NV pair is the TRACE_TARGET of module TraceIRPNotifications::notifyTraceSessionIdentities. Value of NV pair is TraceTarget of module TraceIRPConstDefs.
iOCInstance	One NV pair of filterable_body_fields	M	Name of NV pair is the IOC_INSTANCE of module TraceIRPNotifications::notifyTraceSessionIdentities. Value of NV pair is MOClassName of module TraceIRPConstDefs.

A.3 Solution Set definitions

A.3.1 IDL definition structure

Clause A.3.2 defines the constants and types used by the Trace Management IRP.

Clause A.3.3 defines the operations which are performed by the Trace Management IRP agent.

Clause A.3.4 defines the notifications which are emitted by the Trace Management IRP agent.

A.3.2 IDL specification (file name "TraceIRPConstDefs.idl")

```
//File: TraceIRPConstDefs.idl
#ifndef _Trace_IRP_CONST_DEFS_IDL_
#define _Trace_IRP_CONST_DEFS_IDL_
#include <KernelCmConstDefs.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* ## Module: TraceIRPConstDefs
This module contains commonly used definitions for Trace IRP
===============================================================
*/
module TraceIRPConstDefs
{

enum Result Enum {OK, FAILURE, PARTIAL_SUCCESS};



typedef struct TraceReference
    {       short mcc;
        short mnc;
        unsigned long traceId;
    };
typedef sequence<TraceReference> TraceReferenceList;


typedef unsigned long TraceRecordingSessionReference;

typedef string TraceCollectionEntityAddress;




/* the values of the InterfaceBitmap is coming from the ListOfInterfaces trace parameter definition
in 3GPP TS 32.422. The InterfaceBitmap shall carry the decimal value that is calculated from the
bitmap, defined in TS 32.422.*/

typedef struct Interfaces
    {   NeType NetworkElement;
        Integer InterfaceBitmap;
    };
typedef sequence <Interfaces> ListofInterfaces;


/*
ListOfInterfacesOptional is a type carrying a conditional parameter.
The boolean shall be TRUE, if the operation request uses this parameter. In this case the value is
present. Otherwise the value is absent.
*/
union ListOfInterfacesOptional switch (boolean)
   {
   case TRUE: ListOfInterfaces value;
   };



enum NeType {MSC_SERVER,MGW,RNC,SGSN,GGSN,BM_SC,eNB,MME,SGW,PGW };

typedef sequence<NeType> ListOfNeTypes;


enum TraceDepth {MINIMUM, MEDIUM, MAXIMUM, VENDORMINIMUM, VENDORMEDIUM, VENDORMAXIMUM};

enum TraceTarget {IMSI, IMEI, IMEISV, PUBLIC_ID, UTRAN_CELL, E-UTRAN_CELL, eNB, RNC};

enum JobType {IMMEDIATE_MDT_ONLY, LOGGED_MDT_ONLY, TRACE_ONLY, IMMEDIATE_MDT_TRACE };
typedef sequence <KernelCmConstDefs::DN> DNSet;

enum ReportAmount {1,2,4,8,16,32,64,INFINITY};
enum ReportInterval
{250ms,500ms,1000ms,2000ms,3000ms,4000ms,6000ms,8000ms,12000ms,16000ms,20000ms,24000ms,28000ms,32000
ms,64000ms,120ms,240ms,480ms,640ms,1024ms,2048ms,5120ms,10240ms,60000ms,360000ms,720000ms,1800000ms,
3600000ms};
```

```
enum LoggingInterval {1.28s,2.56s,5.12s,10.24s,20.48s,30.72s,40.96s,61.44s};
enum LoggingDuration {600s,1200s,2400s,3600s,5400s,7200s};
enum AnonymizationOfMDTData {NO_Identity, TAC_of_IMEI};

/* the values of the EventBitmap is coming from the TriggeringEvent trace parameter definition in
3GPP TS 32.422. The EventBitmap shall carry the decimal value that is calculated from the
triggereing event bitmap as defined in TS 32.422.*/

typedef struct Events
    {   NeType NetworkElement;
        Integer EventBitmap;
    };
typedef sequence <Interfaces> TriggeringEvent;

/*
MeasurementsBitMap is used for MDT measurements item selection. Define it as a type of Integer means
the selection is mapped into each bit of total 16 bit, as defined in 32.422 Claus 5.
*/

/* the values of the MeasurementsBitMap is coming from the TriggeringEvent trace parameter
definition in 3GPP TS 32.422. The MeasurementsBitMap shall carry the decimal value that is
calculated from the MDT Measurements bitmap as defined in TS 32.422.*/

enum MobileTechType {UMTS, LTE};
typedef struct Measurements
    {   MobileTechType MobilityTech;
        Integer MeasurementsBitMap;
    };
typedef sequence <Measurements> ListOfMeasurements;

/* Event threshold parameter carries the threshold values used to event triggered MDT reporting. In
LTE case either the EventThreshold RSRP or EventThresholdRSRQ is selected, in UMTS either
EventThreshold1F or EventThreshold1l is selected. */


union EventThreshold switch (long)
    {
    case 0:
        integer EventThresholdRSRP;
    case 1:
        integer EventThresholdRSRQ;
    case 2:
        integer EventThreshold1F;
    case 3:
        integer EventThreshold1l;
    }


/* the values of the MeasurementQuantity is coming from the MeasurementQuantity MDT parameter
definition in 3GPP TS 32.422. The MeasurementsQuanityt shall carry the decimal value that is
calculated from the MDT Measurement Quantity bitmap as defined in TS 32.422.*/

Integer MeasurementQuantity;

/*
ReportingTriggerBitMap is used for MDT measurements reporting. Define it as a type of Integer means
the reporting mechanism is mapped into each bit of total 8 bit, as defined in 32.422 Claus 5.
*/

/* the values of the ReportingTriggerBitMap is coming from the TriggeringEvent trace parameter
definition in 3GPP TS 32.422. The ReportingTriggerBitMap shall carry the decimal value that is
calculated from the MDT Measurements bitmap as defined in TS 32.422.*/

typedef struct ReportingTrigger
    {   MobileTechType MobilityTech;
        Integer ReportingTriggerBitMap;
    };



/*
TriggeringEventConditional is a type carrying a conditional parameter.
The boolean shall be TRUE, if the operation the condition is fulfilled and the request uses this
parameter. In this case the value is present. Otherwise the value is absent.
*/
union TriggeringEventConditional switch (boolean)
   {
   case TRUE: TriggeringEvent value;
   };



enum UnsupportedItem {MANAGED_ENTITY, TRACE_DEPTH, LIST_OF_INTERFACES, TRACE_TARGET, MDT_AREA_SCOPE,
LIST_OF_MEASUREMENTS, REPORTING_TRIGGER, REPORT_INTERVAL, REPORT_AMOUNT, EVENT_THRESHOLD,
LOGGING_INTERVAL, LOGGING_DURATION, ANONYMIZATION_OF_MDT_DATA, REASON};

typedef sequence<UnsupportedItem> UnsupportedList;




/**
* This block identifies attributes which are included as part of the
* notifications defined within TraceIRP. These attribute values should not
* clash with those defined for the attributes of notification
* header (see IDL of Notification IRP).
*/

interface AttributeNameValue
{
   const string TRACE_RECORDING_SESSION_REFERENCE = "TRACE_RECORDING_SESSION_REFERENCE";
   const string TRACE_REFERENCE = "TRACE_REFERENCE";
   const string TRACE_TARGET = "TRACE_TARGET";
   const string MO_INSTANCE = "MO_INSTANCE";
   const string REASON = "REASON";
};

};
```

A.3.3 IDL specification (file name 'TraceIRPSystem.idl')

```
//File: TraceIRPSystem.idl
#ifndef _TRACE_IRP_SYSTEM_IDL_
#define _TRACE_IRP_SYSTEM_IDL_

#include <KernelCmConstDefs.idl>
#include <GenericIRPManagementConstDefs.idl>
#include <GenericIRPManagementSystem.idl>#include <TraceIRPConstDefs.idl>

//This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* Module: TraceIRPSystem
This module contains the specification of all operations of Trace IRP Agent.
=============================================================================
*/
module TraceIRP
{
   exception ActivateTraceJob { string reason; };
   exception NotUniqueTraceReference { string reason; };
   exception DeactivateTraceJob { string reason; };
   exception ListTraceJob { string reason; };
   exception ListActivatedTraceJob { string reason; };

  interface TraceIRP
  {

    /**
    * Request to activate a TraceJob through Itf-N.
    **/

    TraceIRPConstDefs::ResultEnum activateTraceJob (
      in KernelCmConstDefs::DN                             moInstance,
      in TraceIRPConstDefs::ListOfInterfacesOptional       listOfInterfaces,
      in TraceIRPConstDefs::ListOfNeTypes                  listOfNeTypes,
      in TraceIRPConstDefs::TraceDepth                     traceDepth,
      in TraceIRPConstDefs::TraceReference                 traceReference,
      in TraceIRPConstDefs::TraceTarget                    traceTarget,
      in TraceIRPConstDefs::TriggeringEventConditional     triggeringEvent,
      in TraceIRPConstDefs::TraceCollectionEntityAddress   traceCollectionEntityAddress,
      in TraceIRPConstDefs::JobType                        jobType,
      in TraceIRPConstDefs::DNSet                          mdtAreaScope,
      in TraceIRPConstDefs::ListOfMeasurements             listOfMeasurements,
      in TraceIRPConstDefs::ReportingTrigger               reportingTrigger,
      in TraceIRPConstDefs::ReportInterval                 reportInterval,
      in TraceIRPConstDefs::ReportAmount                   reportAmount,
      in TraceIRPConstDefs::EventThreshold                 eventThreshold,
      in TraceIRPConstDefs::LoggingInterval                loggingInterval,
      in TraceIRPConstDefs::LoggingDuration                loggingDuration,
      in TraceIRPConstDefs::AnonymizationOfMDTData         anonymizationOfMDTData,
     in TraceIRPConstDefs::MeasurementQuantity          measurementQuantity,
      out TraceIRPConstDefs:: UnsupportedList              unsupportedList
    )
  raises (ActivateTraceJob,
          GenericIRPManagementSystem::InvalidParameter,
          GenericIRPManagementSystem::ValueNotSupported,
          GenericIRPManagementSystem::OperationNotSupported,
          NotUniqueTraceReference);

    /**
    * Request to deactivate a TraceJob through Itf-N.
    **/

    TraceIRPConstDefs::ResultEnum deactivateTraceJob (
      in TraceIRPConstDefs::TraceReference                  traceReference,
      in TraceIRPConstDefs::TraceTarget                     traceTarget,
      out TraceIRPConstDefs::TraceRecordingSessionReference traceRecordingSessionReference)
    raises (DeactivateTraceJob,
            NotUniqueTraceReference,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported);

    /**
    * Request to list the parameters of a specific TraceJob through Itf-N.
    **/

    TraceIRPConstDefs::ResultEnum listTraceJob (
      in TraceIRPConstDefs::TraceReference                           traceReference,
      out KernelCmConstDefs::DN                                      moInstance,
      out TraceIRPConstDefs::ListOfInterfaces                        listOfInterfaces,
      out TraceIRPConstDefs::TraceDepth                              traceDepth,
      out TraceIRPConstDefs::TraceRecordingSessionReference          traceRecordingSessionReference,
      out TraceIRPConstDefs::TraceTarget                             traceTarget,
      out TraceIRPConstDefs::TriggeringEvent                         triggeringEvent,
      out TraceIRPConstDefs::TraceCollectionEntityAddress traceCollectionEntityAddress,
      out TraceIRPConstDefs::JobType                                  jobType,
      out TraceIRPConstDefs::DNSet                                   mdtAreaScope,
      out TraceIRPConstDefs::ListOfMeasurements                       listOfMeasurements,
      out TraceIRPConstDefs::ReportingTrigger                         reportingTrigger,
      out TraceIRPConstDefs::ReportInterval                           reportInterval,
      out TraceIRPConstDefs::ReportAmount                             reportAmount,
      out TraceIRPConstDefs::Integer                                  eventThreshold,
      out TraceIRPConstDefs::LoggingInterval                          loggingInterval,
      out TraceIRPConstDefs::LoggingDuration                          loggingDuration,
      out TraceIRPConstDefs::AnonymizationOfMDTData                  anonymizationOfMDTData,
      out TraceIRPConstDefs::MeasurementQuantity                       measurementQuantity)

    raises (ListTraceJob,
            NotUniqueTraceReference,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported);

    /**
    * Request to list the activated TraceJobs through Itf-N.
    **/

    TraceIRPConstDefs::ResultEnum listActivatedTraceJob (
      out TraceIRPConstDefs::TraceReferenceList            traceReferenceList)
    raises (ListActivatedTraceJob,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported);
  };

};
#endif // _TRACE_IRP_SYSTEM_IDL_
```

A.3.4 IDL specification (file name 'TraceIRPNotifications.idl')

```
//File: TraceIRPNotifications.idl
#ifndef _TRACE_IRP_NOTIFICATIONS_IDL_
#define _TRACE_IRP_NOTIFICATIONS_IDL_

#include <TraceIRPConstDefs.idl>
#include <NotificationIRPNotifications.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* Module: TraceIRPNotifications
This module contains the specification of all notifications of Trace IRP Agent.
=============================================================================
*/
module TraceIRPNotifications
{

  /**
  * Constant definitions for the notifyTraceRecordingSessionFailure notification
  **/

  interface NotifyTraceRecordingSessionFailure: NotificationIRPNotifications::Notify
  {
    const string EVENT_TYPE = "notifyTraceRecordingSessionFailure";

    /**
    * This constant defines the name of the TraceRecordingSessionReference property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceRecordingSessionReference.
    **/

    const string TRACE_RECORDING_SESSION_REFERENCE =
TraceIRPConstDefs::AttributeNameValue::TRACE_RECORDING_SESSION_REFERENCE;

    /**
    * This constant defines the name of the TraceReference property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceReference.
    **/

    const string TRACE_REFERENCE = TraceIRPConstDefs::AttributeNameValue::TRACE_REFERENCE;


    /**
    * This constant defines the name of the reason property.
    * The data type for the value of this property is string.
    */

    const string REASON = TraceIRPConstDefs::AttributeNameValue::REASON;
   };

  /**
  * Constant definitions for the notifyTraceSessionLocalActivation notification
  **/

  interface NotifyTraceSessionLocalActivation: NotificationIRPNotifications::Notify
  {
    const string EVENT_TYPE = "notifyTraceSessionLocalActivation";

    /**
    * This constant defines the name of the TraceReference property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceReference.
    **/

    const string TRACE_REFERENCE = TraceIRPConstDefs::AttributeNameValue::TRACE_REFERENCE;

    /**
    * This constant defines the name of the TraceTarget property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceTarget.
    **/

    const string TRACE_TARGET = TraceIRPConstDefs::AttributeNameValue::TRACE_TARGET;
    /**
    * This constant defines the name of the Managed Entity Object Instance property.
    * The data type for the value of this property is string.
    */

    const string MO_INSTANCE = TraceIRPConstDefs::AttributeNameValue::MO_INSTANCE;
   };


  /**
  * Constant definitions for the notifyTraceSessionIdentities notification
  **/

  interface NotifyTraceSessionIdentities: NotificationIRPNotifications::Notify
  {
    const string EVENT_TYPE = "notifyTraceSessionIdentities";

    /**
    * This constant defines the name of the TraceReference property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceReference.
    **/

    const string TRACE_REFERENCE = TraceIRPConstDefs::AttributeNameValue::TRACE_REFERENCE;

    /**
    * This constant defines the name of the TraceRecordingSessionReference property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceRecordingSessionReference.
    **/

    const string TRACE_RECORDING_SESSION_REFERENCE =
        TraceIRPConstDefs::AttributeNameValue::TRACE_RECORDING_SESSION_REFERENCE;


    /**
    * This constant defines the name of the TraceTarget property.
    * The data type for the value of this property is
    * TraceIRPConstDefs::TraceTarget.
    **/

    const string TRACE_TARGET = TraceIRPConstDefs::AttributeNameValue::TRACE_TARGET;


    /**
    * This constant defines the name of the Managed Entity Object Instance property.
    * The data type for the value of this property is string.
    */

    const string MO_INSTANCE = TraceIRPConstDefs::AttributeNameValue::MO_INSTANCE;
   };

};

#endif // _TRACE_IRP_NOTIFICATIONS_IDL_
```

Annex B (normative): XML definitions

This annex contains the XML definitions for the Trace Management IRP for the IRP whose semantics is specified in Trace Management IRP: Information Service (3GPP TS 32.442 [5]).

This XML definitions specification defines the XML syntax of the Trace Management IRP XML Data File.

B.1 Architectural Features

The overall architectural feature of Trace Management IRP is specified in 3G TS 32.442 [5]. This clause specifies features that are specific to the XML definitions.

B.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [9].

B.1.2 Notification Services

This annex defines the XML syntax of Trace Management IRP notifications that is to be used for the Trace Management IRP SOAP Solution Set and in conjunction with Notification Log IRP XML definitions for Notification Log IRP XML Data File and the NL IRP XML Notification Format.

B.1.3 IOC definitions

This annex defines the XML syntax for the IOC definitions of the Trace Management IRP IS [5], which are used by the XML definitions for the Trace Management IRP notifications and the Trace Management IRP IS operations.

B.2 Mapping

Not present in the current version of this specification.

B.3 Solution Set definitions

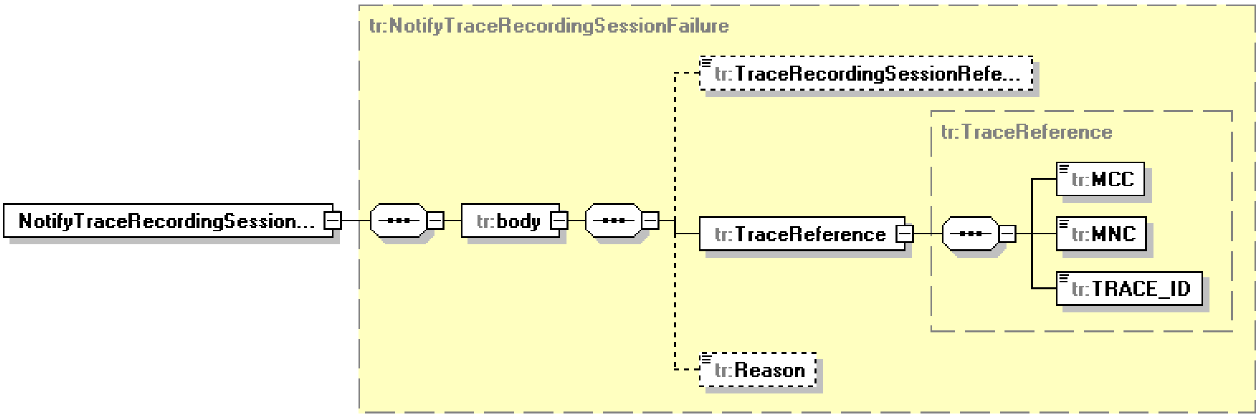
B.3.1 XML definition structure

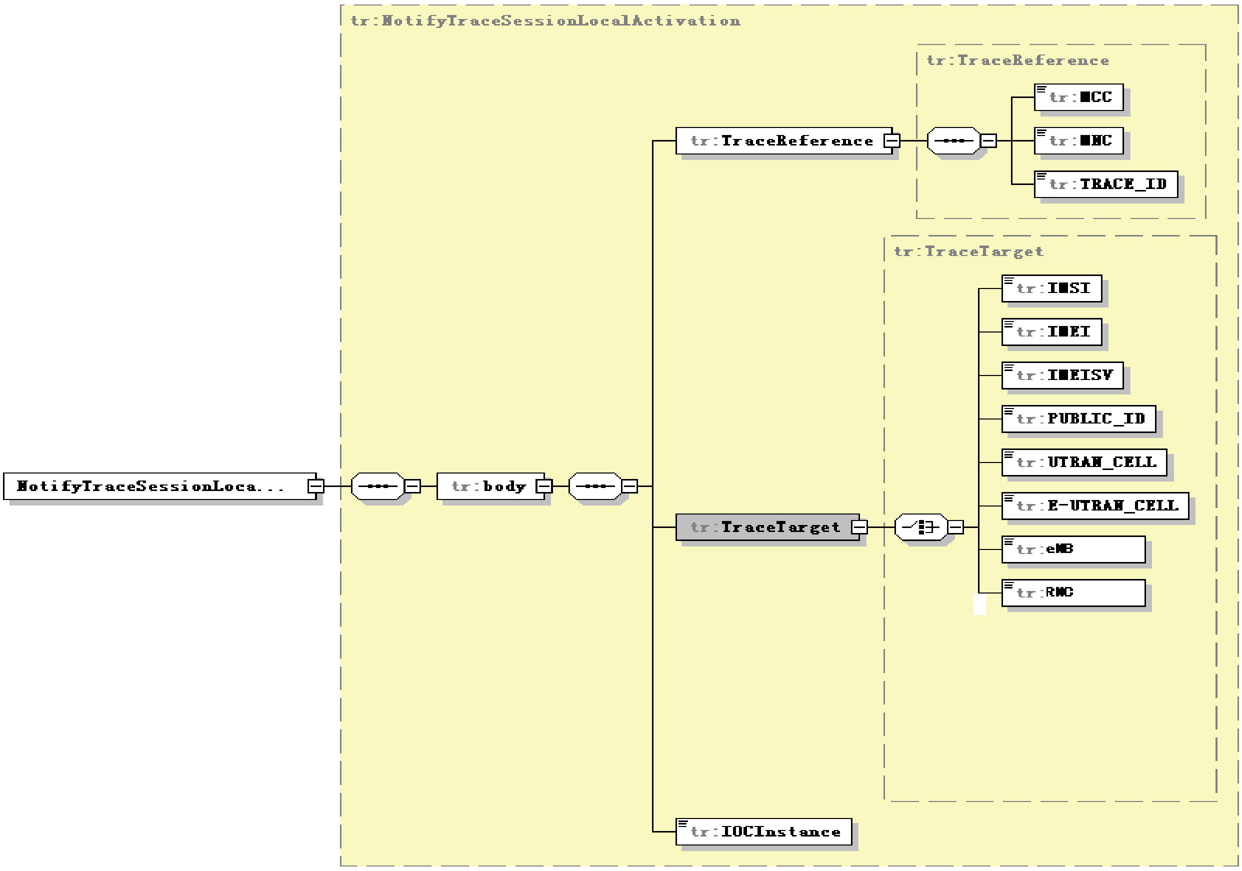
Clause B.3.2 provides a graphical representation of the XML elements.

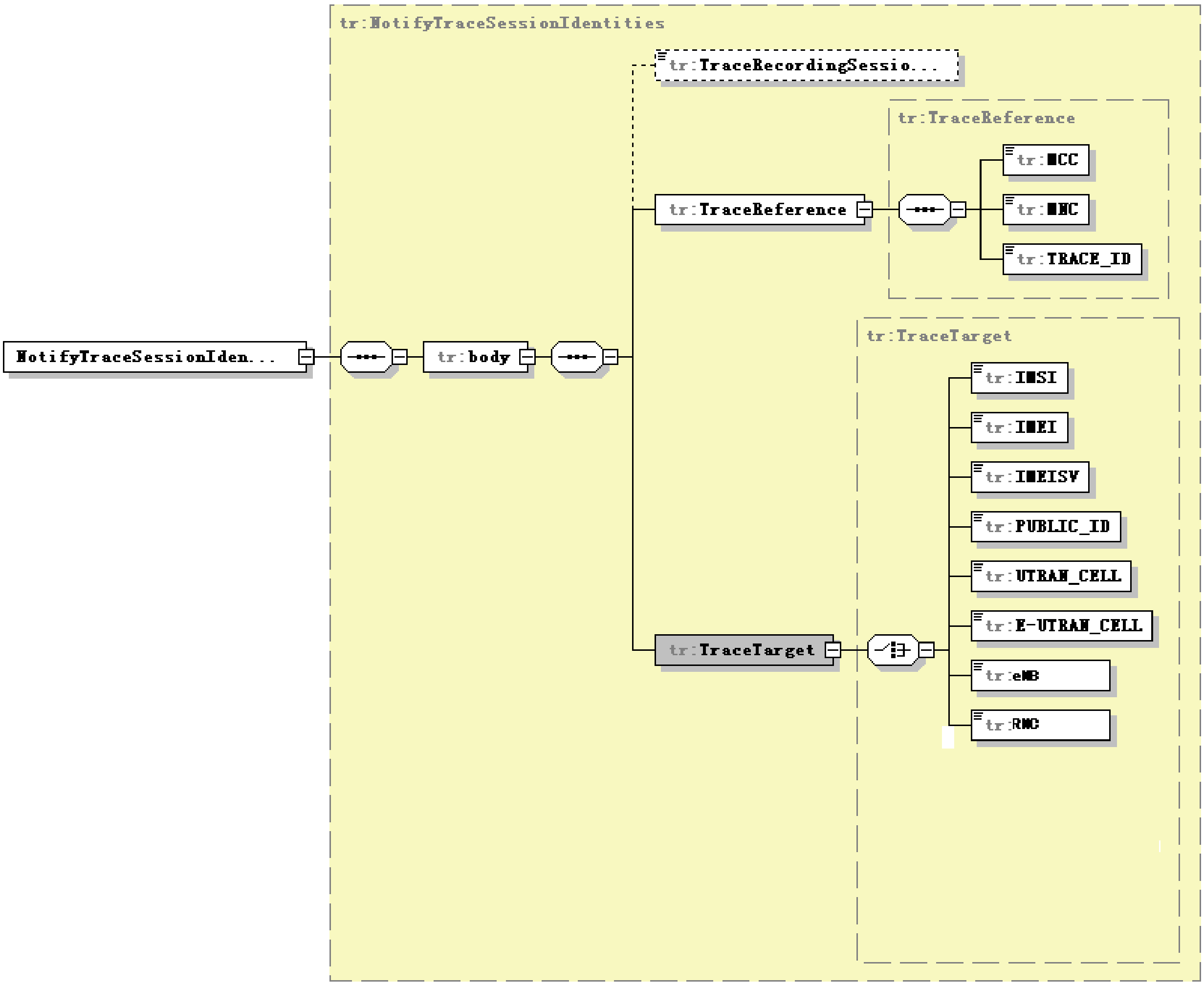
Clause B.3.3 provides XML definitions of Trace Management IRP notifications as defined in [5]. These definitions are to be used for the Trace Management IRP SOAP Solution Set. For Trace IRP XML File Name Conventions the generic file name definitions as specified by the FT IRP apply (see [13]).

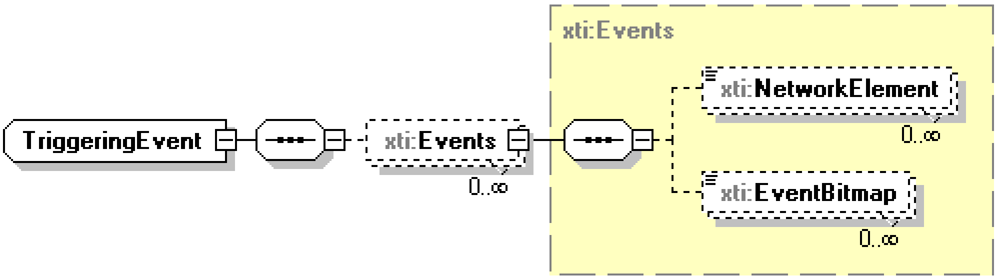
Clause B.3.4 provides XML definitions of Trace Management IOC as defined in [5].

B.3.2 Graphical Representation


tr:NotifyTraceRecordingSessionFailure
NotifyTraceRecordingSession...
tr:body
tr:TraceRecordingSessionRefe...
tr:TraceReference
tr:TraceReference
tr:MCC
tr:MNC
tr:TRACE_ID
tr:Reason




tr:NotifyTraceSessionIdentities
NotifyTraceSessionIden...
tr:body
tr:TraceRecordingSessio...
tr:TraceReference
tr:TraceReference
tr:MCC
tr:MNC
tr:TRACE_ID
tr:TraceTarget
tr:TraceTarget
tr:IMSI
tr:IMEI
tr:IMEISV
tr:PUBLIC_ID
tr:UTRAN_CELL
tr:E-UTRAN_CELL
tr:eNB
tr:RNC


xti:Events
TriggeringEvent
xti:Events
0..∞
xti:NetworkElement
0..∞
xti:EventBitmap
0..∞

B.3.3 XML Schema 'tMIRPNotif.xsd'

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  3GPP TS 32.446 Trace Management IRP Notification
  Trace IRP specific data file XML schema
  tMIRPNotif.xsd
-->
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:tr="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPNotif"
xmlns:xe="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification"
targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPNotif"
elementFormDefault="qualified">
    <import namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification"/>
    <!-- XML types specific for trace IRP notifications -->
    <complexType name="TraceReference">
        <sequence>
            <element name="MCC" type="short"/>
            <element name="MNC" type="short"/>
            <element name="TRACE_ID" type="integer"/>
        </sequence>
    </complexType>
    <complexType name="NotifyTraceRecordingSessionFailure">
        <complexContent>
            <extension base="xe:Notification">
                <sequence>
                    <element name="body">
                        <complexType>
                            <sequence>
                                <element name="TraceRecordingSessionReference" type="integer"
minOccurs="0"/>
                                <element name="TraceReference" type="tr:TraceReference"/>
                                <element name="Reason" type="string" minOccurs="0"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NotifyTraceRecordingSessionFailure"
type="tr:NotifyTraceRecordingSessionFailure"/>
    <complexType name="TraceTarget">
        <choice>
            <element name="IMSI" type="string"/>
            <element name="IMEI" type="string"/>
            <element name="IMEISV" type="string"/>
            <element name="PUBLIC_ID" type="string"/>
            <element name="UTRAN_CELL" type="string"/>
            <element name="E-UTRAN_CELL" type="string"/>
            <element name="eNB" type="string"/>
            <element name="RNC" type="string"/>
        </choice>
    </complexType>
    <complexType name="NotifyTraceSessionLocalActivation">
        <complexContent>
            <extension base="xe:Notification">
                <sequence>
                    <element name="body">
                        <complexType>
                            <sequence>
                                <element name="TraceReference" type="tr:TraceReference"/>
                                <element name="TraceTarget" type="tr:TraceTarget"/>
                                <element name="IOCInstance" type="string"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NotifyTraceSessionLocalActivation" type="tr:NotifyTraceSessionLocalActivation"/>
    <complexType name="NotifyTraceSessionIdentities">
        <complexContent>
            <extension base="xe:Notification">
                <sequence>
```

```
                    <element name="body">
                        <complexType>
                            <sequence>
                                <element name="TraceRecordingSessionReference" type="integer"
minOccurs="0"/>
                                <element name="TraceReference" type="tr:TraceReference"/>
                                <element name="TraceTarget" type="tr:TraceTarget"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NotifyTraceSessionIdentities" type="tr:NotifyTraceSessionIdentities"/>
</schema>
```

B.3.4 XML Schema 'tMIRPIOCs.xsd'

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
    3GPP TS 32.446 Trace Management IRP IOC XML Schema
    tMIRPIOCs.xsd
-->
<schema xmlns:xti="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPIOCs"
xmlns:xe="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification"
xmlns:xn=http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm
xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPIOCs" <import
namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm"/>
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <complexType name="ListOfInterfaces">
        <sequence>
            <element name="Interface" type="integer" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <simpleType name="NeType">
        <restriction base="string">
            <enumeration value="MSC_SERVER"/>
            <enumeration value="MGW"/>
            <enumeration value="RNC"/>
            <enumeration value="SGSN"/>
            <enumeration value="GGSN"/>
            <enumeration value="BM_SC"/>
            <enumeration value="eNB"/>
            <enumeration value="MME"/>
            <enumeration value="SGW"/>
            <enumeration value="PGW"/>
        </restriction>
    </simpleType>
    <simpleType name="MobilityTechType">
        <restriction base="string">
            <enumeration value="UMTS"/>
            <enumeration value="LTE"/>
        </restriction>
    </simpleType>
    <complexType name="ListOfNeTypes">
        <sequence>
            <element name="NE" type="xti:NeType" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <complexType name="DNSet">
        <sequence>
           <element name="DN" type="xn:DN" minOccurs="0" maxOccurs="unbounded"/>
         </sequence>
    </complexType>
    <simpleType name="TraceDepth">
        <restriction base="string">
            <enumeration value="MINIMUM"/>
            <enumeration value="MEDIUM"/>
            <enumeration value="MAXIMUM"/>
            <enumeration value="VENDORMINIMUM"/>
            <enumeration value="VENDORMEDIUM"/>
            <enumeration value="VENDORMAXIMUM"/>
        </restriction>
    </simpleType>
        <restriction base="string">
            <enumeration value="IMSI"/>
            <enumeration value="IMEI"/>
            <enumeration value="IMEISV"/>
            <enumeration value="PUBLIC_ID"/>
            <enumeration value="UTRAN_CELL_ID"/>
            <enumeration value="EUTRAN_CELL_ID"/>
            <enumeration value="eNB_ID"/>
            <enumeration value="RNC_ID"/>
        </restriction>
    </simpleType>
    <!-- the values of the EventBitmap is coming from the TriggeringEvent trace parameter definition
in 3GPP TS 32.422 -->
    <complexType name="Events">
        <sequence>
            <element name="NetworkElement" type="xti:NeType" minOccurs="0" maxOccurs="unbounded"/>
            <element name="EventBitmap" type="integer" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <complexType name=" TriggeringEvent">
        <sequence>
            <element name="Events" type="xti:Events" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <simpleType name="UnsupportedItem">
        <restriction base="string">
            <enumeration value="MANAGED_ENTITY"/>
            <enumeration value="TRACE_DEPTH"/>
            <enumeration value="LIST_OF_INTERFACES"/>
            <enumeration value="TRACE_TARGET"/>
            <enumeration value="LIST_OF_MEASUREMENTS"/>
            <enumeration value="MDT_AREA_SCOPE"/>
            <enumeration value="REPORTING_TRIGGER"/>
            <enumeration value="REPORT_INTERVAL"/>
            <enumeration value="REPORT_AMOUNT"/>
            <enumeration value="EVENT_THRESHOLD"/>
            <enumeration value="LOGGING_INTERVAL"/>
            <enumeration value="LOGGING_DURATION"/>
            <enumeration value="ANONYMIZATION_OF_MDT_DATA"/>
            <enumeration value="MEASUREMENT_QUANTITY"/>

            <enumeration value="REASON"/>
        </restriction>
    </simpleType>
    <simpleType name="JobType">
        <restriction base="string">
            <enumeration value="IMMEDIATE_MDT_ONLY"/>
            <enumeration value="LOGGED_MDT_ONLY"/>
            <enumeration value="TRACE_ONLY"/>
            <enumeration value="IMMEDIATE_MDT AND TRACE"/>
        </restriction>
    </simpleType>
    <complexType name="Measurements">
        <sequence>
            <element name="MobilityTech" type="xti:MobilityTechType" minOccurs="0"
maxOccurs="unbounded"/>
            <element name="MeasurementsBitmap" type="integer" minOccurs="0" maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <complexType name="ListOfMeasurements">
        <sequence>
            <element name="Measurements" type="xti:Measurements" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <complexType name="ReportingTrigger">
        <sequence>
            <element name="MobilityTech" type="xti:MobilityTechType" minOccurs="0"
maxOccurs="unbounded"/>
            <element name="ReportingTriggerBitmap" type="integer" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <simpleType name="ReportInterval">
        <restriction base="string">
            <enumeration value="250ms"/>
            <enumeration value="500ms"/>
            <enumeration value="1000ms"/>
            <enumeration value="2000ms"/>
            <enumeration value="3000ms"/>
            <enumeration value="4000ms"/>
            <enumeration value="6000ms"/>
            <enumeration value="8000ms"/>
            <enumeration value="12000ms"/>
            <enumeration value="16000ms"/>
            <enumeration value="20000ms"/>
            <enumeration value="24000ms"/>
            <enumeration value="28000ms"/>
            <enumeration value="32000ms"/>
            <enumeration value="64000ms"/>
            <enumeration value="120ms"/>
            <enumeration value="240ms"/>
            <enumeration value="480ms"/>
            <enumeration value="640ms"/>
            <enumeration value="1024ms"/>
            <enumeration value="2048ms"/>
            <enumeration value="5120ms"/>
            <enumeration value="10240ms"/>
            <enumeration value="60000ms"/>
            <enumeration value="360000ms"/>
            <enumeration value="720000ms"/>
            <enumeration value="1800000ms"/>
            <enumeration value="3600000ms"/>
        </restriction>
    </simpleType>
    <simpleType name="ReportAmount">
        <restriction base="string">
            <enumeration value="1"/>
            <enumeration value="2"/>
            <enumeration value="4"/>
            <enumeration value="8"/>
            <enumeration value="16"/>
            <enumeration value="32"/>
            <enumeration value="64"/>
            <enumeration value="INFINITY"/>
        </restriction>
    </simpleType>
    <simpleType name="LoggingInterval">
        <restriction base="string">
            <enumeration value="1.28s"/>
            <enumeration value="2.56s"/>
            <enumeration value="5.12s"/>
            <enumeration value="10.24s"/>
            <enumeration value="20.48s"/>
            <enumeration value="30.72s"/>
            <enumeration value="40.96s"/>
            <enumeration value="61.44s"/>
        </restriction>
    </simpleType>
    <simpleType name="LoggingDuration">
        <restriction base="string">
            <enumeration value="600s"/>
            <enumeration value="1200s"/>
            <enumeration value="2400s"/>
            <enumeration value="3600s"/>
            <enumeration value="5400s"/>
            <enumeration value="7200s"/>
        </restriction>
    </simpleType>
    <simpleType name="AnonymizationOfMDTData">
        <restriction base="string">
            <enumeration value="NO_IDENTITY"/>
            <enumeration value="TAC_OF_IMEI"/>
        </restriction>
    </simpleType>

<complexType name='EventThreshold'>
    <choice>
        <element name='EventThresholdRSRP' type='integer'>
        <element name='EventThresholdRSRQ' type='integer'>
        <element name='EventThreshold1F' type='integer'>
        <element name='EventThreshold1l' type='integer'>
    </choice>
</complexType>


    <complexType name="UnsupportedList">
        <sequence>
            <element name="UnsupportedItem" type="xti:UnsupportedItem" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
    </complexType>
    <!-- Attributes of the TraceJob IOC -->
    <element name="traceReference" type="unsignedLong"/>
    <element name="listOfInterfaces" type="xti:ListOfInterfaces"/>
    <element name="listOfNeTypes" type="xti:ListOfNeTypes"/>
    <element name="traceDepth" type="xti:TraceDepth"/>
    <element name="traceTarget" type="xti:TraceTarget"/>
    <element name="triggeringEvent" type="xti:TraceTarget"/>
    <element name="traceCollectionEntityAddress" type="string"/>
    <element name="jobType" type="xti:JobType"/>
    <element name="mdtAreaScope" type="xti:DNSet"/>
    <element name="listOfMeasurements" type="xti:ListOfMeasurements"/>
    <element name="reportingTrigger" type="xti:ReportingTrigger"/>
    <element name="reportInterval" type="xti:ReportInterval"/>
    <element name="reportAmount" type="xti:ReportAmount"/>
    <element name="eventThreshold" type="xti:EventThreshold"/>
    <element name="loggingInterval" type="xti:LoggingInterval"/>
    <element name="loggingDuration" type="xti:LoggingDuration"/>
    <element name="anonymizationOfMDTData" type="xti:anonymizationOfMDTData"/>
    <element name="measurementQuantity" type="integer"/>
    <!-- Attributes of the TraceRecord IOC -->
    <element name="fileName" type="string"/>
</schema>
```

Annex C (normative): SOAP Solution Set

This annex specifies the SOAP Solution Set for the IRP whose semantics are specified in Trace Management IRP: Information Service (3GPP TS 32.442 [5]).

C.1 Architectural features

The overall architectural feature of the Trace Management IRP is specified in 3GPP TS 32.442 [5]. This clause specifies features that are specific to the SOAP solution set.

C.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [9].

C.1.2 Notification Services

The Trace Management IRP SOAP SS uses the Notification IRP SOAP SS of 3GPP TS 32.306 [10]. The IRPAgent shall support the push interface model, which means that the IRPAgent sends trace management notifications to the IRPManager as soon as new events occur. The IRPManager does not need to check ("pull") for events.

C.1.3 Supported W3C specifications

The SOAP 1.1 specification [21] and WSDL 1.1 specification [23] are supported.

The SOAP 1.2 specification [24] is supported optionally.

This specification uses "document" style in WSDL file.

This specification uses "literal" encoding style in WSDL file.

The filter language used in the SS is the XPath Language (see W3C XPath 1.0 specification [22]). IRPAgents may throw a FilterComplexityLimit fault when a given filter is too complex.

Relevant definitions are imported from the Trace Management IRP XML definitions of Annex B

C.1.4 Prefixes and namespaces

This specification uses a number of namespace prefixes throughout that are listed in Table C.1.4.

Table C.1.4: Prefixes and Namespaces used in this specification

PREFIX	NAMESPACE
(no prefix)	http://schemas.xmlsoap.org/wsdl/
soap	http://schemas.xmlsoap.org/wsdl/soap/
traceRPSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPSystem
traceIRPData	http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPData
xti	http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPIOCs
xn	http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm
genericIRPSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRPSystem
ntfIRPNtfSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPNtfSystem

C.2 Mapping

C.2.1 Operation and notification mapping

The Trace Management IRP IS (3GPP TS 32.442 [5]) defines semantics of operation and notification visible across the Itf-N. Table C.2.1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table C.2.1: Mapping from IS Operation to SS Equivalents

IS Operations in 3GPP TS 32.442 [5]	SS Operations	SS Port	Qualifier
activateTraceJob	activateTraceJob	TraceIRPManagementPort	M
deactivateTraceJob	deactivateTraceJob	TraceIRPManagementPort	M
listTraceJob	listTraceJob	TraceIRPManagementPort	M
listActivatedTraceJobs	listActivatedTraceJobs	TraceIRPManagementPort	O
notifyTraceRecordingSessionFailure	notify (note 1)	NotificationIRPNtfPort	O
notifyTraceSessionLocalActivation	notify (note 1)	NotificationIRPNtfPort	M
notifyTraceSessionIdentities	notify (note 1)	NotificationIRPNtfPort	CM
NOTE 1: The IS equivalent maps to an XML definition specified in Annex B, and this being an input parameter to the operation notify under the port type ntfIRPNtfSystem:NotificationIRPNtf and under the binding ntfIRPNtfSystem:NotificationIRPNtf of 3GPP TS 32.306 [10].			

C.2.2 Operation parameter mapping

The Trace Management IRP IS (3GPP TS 32.442 [5]) defines semantics of parameters carried in the operations. The tables below show the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table C.2.2.1: Mapping from IS `activateTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
iOCInstance	iOCInstance	M
listOfInterfaces	listOfInterfaces	O
listOfNeTypes	listOfNeTypes	CM
traceDepth	traceDepth	M
traceReference	traceReference	M
traceTarget	traceTarget	M
triggeringEvent	triggeringEvent	CO
traceCollectionEntityAddress	traceCollectionEntityAddress	CM
jobType	jobType	M
mdtAreaScope	mdtAreaScope	CM
listOfMeaurements	listOfMeaurements	CM
reportingTrigger	reportingTrigger	CM
reportInterval	reportInterval	CM
reportAmount	reportAmount	CM
eventThreshold	eventThreshold	CM
loggingInterval	loggingInterval	CM
loggingDuration	loggingDuration	CM
anonymizationOfMDTData	anonymizationOfMDTData	CM
measurementQuantity	measurementQuantity	CM
unsupportedList	unsupportedList	M
status	status	M

Table C.2.2.2: Mapping from IS `deactivateTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReference	traceReference	M
traceTarget	traceTarget	M
traceRecordingSessionReference	traceRecordingSessionReference	CM
status	status	M

Table C.2.2.3: Mapping from IS `listTraceJob` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReference	traceReference	M
iOCInstance	iOCInstance	M
listOfInterfaces	listOfInterfaces	O
traceDepth	traceDepth	M
traceRecordingSessionReference	traceRecordingSessionReference	CM
traceTarget	traceTarget	M
triggeringEvent	triggeringEvent	O
traceCollectionEntityAddress	traceCollectionEntityAddress	CM
jobType	jobType	M
mdtAreaScope	mdtAreaScope	CM
listOfMeaurements	listOfMeaurements	CM
reportingTrigger	reportingTrigger	CM
reportInterval	reportInterval	CM
reportAmount	reportAmount	CM
eventThreshold	eventThreshold	CM
loggingInterval	loggingInterval	CM
loggingDuration	loggingDuration	CM
anonymizationOfMDTData	anonymizationOfMDTData	CM
measurementQuantity	measurementQuantity	CM
status	status	M

Table C.2.2.4: Mapping from IS `listTraceJobs` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
traceReferenceList	traceReferenceList	M
status	status	M

C.2.3 Notification parameter mapping

The Trace Management IRP IS (3GPP TS 32.442 [5]) defines semantics of parameters carried in notifications. The following tables indicate the mapping of these parameters to their SS equivalents.

Table C.2.3.1: Mapping for `notifyTraceRecordingSessionFailure`

IS Parameters	<SS> Parameters	Qualifier	Comment
objectClass	objectClass	M	
objectInstance	objectInstance	M	
eventTime	eventTime	M	
notificationType	notificationType	M	
systemDN	systemDN	M	
notificationID	notificationID	O	
traceRecordingSessionReference	traceRecordingSessionReference	O	
traceReference	traceReference	M	
reason	reason	O	

Table C.2.3.2: Mapping for `notifyTraceSessionLocalActivation`

IS Parameters	<SS> Parameters	Qualifier	Comment
objectClass	objectClass	M	
objectInstance	objectInstance	M	
eventTime	eventTime	M	
notificationType	notificationType	M	
systemDN	systemDN	M	
notificationID	notificationID	O	
traceReference	traceReference	M	
traceTarget	traceTarget	M	
iOCInstance	iOCInstance	M	

Table C.2.3.3: Mapping for **`notifyTraceSessionIdentities`**

IS Parameters	<SS> Parameters	Qualifier	Comment
objectClass	objectClass	M	
objectInstance	objectInstance	M	
eventTime	eventTime	M	
notificationType	notificationType	M	
systemDN	systemDN	M	
notificationID	notificationID	O	
traceReference	traceReference	M	
traceRecordingSessionReference	traceRecordingSessionReference	M	
traceTarget	traceTarget	M	

C.3 Solution Set definitions

C.3.1 WSDL definition structure

Clause C.3.2 provides a graphical representation of the Trace Management IRP service.

Clause C.3.3 defines the services which are supported the Trace Management IRP agent.

C.3.2 Graphical Representation

The WSDL structure is depicted in Figure C.3.2 below, depicting port type, binding and service. The port type contains port type operations, which again contains input, output and fault messages. The binding contains binding operations, which have the same name as the port type operations. The binding connects to a port inside the service.

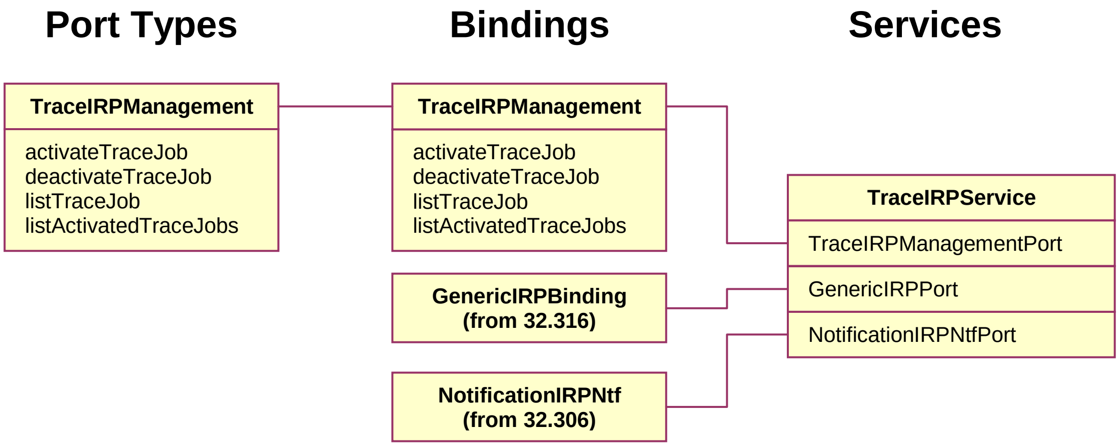


Figure C.3.2: Trace Management IRP SOAP Solution Set WSDL structure

C.3.3 WSDL specification 'TraceIRPSystem.wsdl'

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
   3GPP TS 32.446 Trace Management IRP SOAP Solution Set
-->
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:traceIRPSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPSystem"
xmlns:traceIRPData="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPData"
xmlns:xn="http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm"
xmlns:genericIRPSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRPSystem"
xmlns:ntfIRPNtfSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPNtfSyst
em" targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPSystem">
  <import namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRPSystem"/>
  <import
namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.307/schema/32306#notification/Notifica
tionIRPNtfSystem"/>
  <types>
    <schema targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRPData"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xti="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#tMIRPIOCs">
      <!-- activateTraceJob Request -->
      <element name="activateTraceJobRequest">
        <complexType>
          <sequence>
            <element name="iOCInstance" type="xn:dn"/>
            <element name="listOfInterfaces" type="xti:ListOfInterfaces" minOccurs="0"/>
            <element name="listOfNeTypes" type="xti:ListOfNeTypes" minOccurs="0"/>
            <element name="traceDepth" type="xti:TraceDepth"/>
            <element name="traceReference" type="unsignedLong"/>
            <element name="traceTarget" type="xti:TraceTarget"/>
            <element name="triggeringEvent" type="xti:TriggeringEvent" minOccurs="0"/>
            <element name="traceCollectionEntityAddress" type="string" minOccurs="0"/>
            <element name="jobType" type="xti:JobType" minOccurs="0"/>
            <element name="mdtAreaScope" type="xti:DNSet" minOccurs="0"/>
            <element name="listOfMeasurements" type="xti:ListOfMeasurements" minOccurs="0"/>
            <element name="reportingTrigger" type="xti:ReportingTrigger" minOccurs="0"/>
            <element name="reportInterval" type="xti:ReportInterval" minOccurs="0"/>
            <element name="reportAmount" type="xti:ReportAmount" minOccurs="0"/>
            <element name="eventThreshold" type="xti:EventThreshold" minOccurs="0"/>
            <element name="loggingInterval" type="xti:LoggingInterval" minOccurs="0"/>
            <element name="loggingDuration" type="xti:LoggingDuration" minOccurs="0"/>
            <element name="anonymizationOfMDTData" type="xti:anonymizationOfMDTData" minOccurs="0"/>
            <element name="measurementQuantity" type="integer" minOccurs="0"/>
          </sequence>
        </complexType>
      </element>
      <!-- activateTraceJob Response -->
      <element name="activateTraceJobResponse">
        <complexType>
          <sequence>
            <element name="status">
              <simpleType>
                <restriction base="string">
                  <enumeration value="Success"/>
                  <enumeration value="Failure"/>
                  <enumeration value="PartialSuccess"/>
                </restriction>
              </simpleType>
            </element>
            <element name="unsupportedList" type="xti:UnsupportedList" minOccurs="0"/>
            <element name="failureReason" minOccurs="0">
              <simpleType>
                <restriction base="string">
                  <enumeration value="invalidTraceDepth"/>
                  <enumeration value="invalidListOfInterfaces"/>
                  <enumeration value="invalidTraceTarget"/>
                  <enumeration value="invalidMdtAreaScope"/>
                  <enumeration value="invalidListOfMeasurements"/>
                  <enumeration value="invalidReportingTrigger"/>
                  <enumeration value="invalidReportInterval"/>
                  <enumeration value="invalidReportAmount"/>
                  <enumeration value="invalidEventThreshold"/>
                  <enumeration value="invalidLoggingInterval"/>
                  <enumeration value="invalidLoggingDuration"/>
                  <enumeration value="invalidAnonymizationOfMDTData"/>
                  <enumeration value="notuniqueTraceReference"/>
                  <enumeration value="operation_failed"/>
                  <enumeration value="operation_failed_invalid_input_parameter"/>
                  <enumeration
value="operation_failed_unsupported_optional_input_parameter_listOfInterfaces"/>
                  <enumeration
value="operation_failed_unsupported_optional_input_parameter_listOfNeTypes"/>
                  <enumeration
value="operation_failed_unsupported_optional_input_parameter_triggeringEvent"/>
                  <enumeration
value="operation_failed_unsupported_optional_input_parameter_traceCollectionEntityAddress"/>
                  <enumeration value="operation_failed_internal_problem"/>
                  <enumeration value="invalidMeasurementQuantity"/>
                </restriction>
              </simpleType>
            </element>
          </sequence>
        </complexType>
      </element>
      <!-- activateTraceJob Fault -->
      <element name="activateTraceJobFault">
        <simpleType>
          <restriction base="string">
            <enumeration value="OperationFailed"/>
          </restriction>
        </simpleType>
      </element>
      <!-- deactivateTraceJob Request -->
      <element name="deactivateTraceJobRequest">
        <complexType>
          <sequence>
            <element name="traceReference" type="unsignedLong"/>
            <element name="traceTarget" type="xti:TraceTarget"/>
          </sequence>
        </complexType>
      </element>
      <!-- deactivateTraceJob Response -->
      <element name="deactivateTraceJobResponse">
        <complexType>
          <sequence>
            <element name="status">
              <simpleType>
                <restriction base="string">
                  <enumeration value="Success"/>
                  <enumeration value="Failure"/>
                </restriction>
              </simpleType>
            </element>
            <element name="traceRecordingSessionReference" type="integer" minOccurs="0"/>
            <element name="failureReason" minOccurs="0">
              <simpleType>
                <restriction base="string">
                  <enumeration value="notuniqueTraceReference"/>
                  <enumeration value="operation_failed"/>
                  <enumeration value="operation_failed_internal_problem"/>
                </restriction>
              </simpleType>
            </element>
          </sequence>
        </complexType>
      </element>
      <!-- deactivateTraceJob Fault -->
      <element name="deactivateTraceJobFault">
        <simpleType>
          <restriction base="string">
            <enumeration value="OperationFailed"/>
          </restriction>
        </simpleType>
      </element>
      <!-- listTraceJob Request -->
      <element name="listTraceJobRequest">
        <complexType>
          <sequence>
            <element name="traceReference" type="unsignedLong"/>
          </sequence>
        </complexType>
      </element>
      <!-- listTraceJob Response -->
```

```
<element name="listTraceJobResponse">
  <complexType>
    <sequence>
      <element name="iOCInstance" type="xn:dn"/>
      <element name="listOfInterfaces" type="xti:ListOfInterfaces" minOccurs="0"/>
      <element name="status">
        <simpleType>
          <restriction base="string">
            <enumeration value="Success"/>
            <enumeration value="Failure"/>
          </restriction>
        </simpleType>
      </element>
      <element name="traceDepth" type="xti:TraceDepth"/>
      <element name="traceRecordingSessionReference" type="integer" minOccurs="0"/>
      <element name="traceTarget" type="xti:TraceTarget"/>
      <element name="triggeringEvent" type="xti:TriggeringEvent" minOccurs="0"/>
      <element name="traceCollectionEntityAddress" type="string" minOccurs="0"/>
      <element name="jobType" type="xti:JobType" minOccurs="0"/>
      <element name="mdtAreaScope" type="xti:DNSet" minOccurs="0"/>
      <element name="listOfMeasurements" type=" xti:ListOfMeasurements" minOccurs="0"/>
      <element name="reportingTrigger" type="xti:ReportingTrigger" minOccurs="0"/>
      <element name="reportInterval" type="xti:ReportInterval" minOccurs="0"/>
      <element name="reportAmount" type="xti:ReportAmount" minOccurs="0"/>
      <element name="eventThreshold" type="Integer" minOccurs="0"/>
      <element name="loggingInterval" type="xti:LoggingInterval" minOccurs="0"/>
      <element name="loggingDuration" type="xti:LoggingDuration" minOccurs="0"/>
      <element name="anonymizationOfMDTData" type="xti:anonymizationOfMDTData" minOccurs="0"/>
      <element name="measurementQuantity" type="integer" minOccurs="0"/>
      <element name="failureReason" minOccurs="0">
        <simpleType>
          <restriction base="string">
            <enumeration value="notuniqueTraceReference"/>
            <enumeration value="operation_failed"/>
            <enumeration value="operation_failed_internal_problem"/>
          </restriction>
        </simpleType>
      </element>
    </sequence>
  </complexType>
</element>
<!-- listTraceJob Fault -->
<element name="listTraceJobFault">
  <simpleType>
    <restriction base="string">
      <enumeration value="OperationFailed"/>
    </restriction>
  </simpleType>
</element>
<!-- listActivatedTraceJobs Request -->
<element name="listActivatedTraceJobsRequest">
</element>
<!-- listActivatedTraceJobs Response -->
<element name="listActivatedTraceJobsResponse">
  <complexType>
    <sequence>
      <element name="traceReferenceList">
        <complexType>
          <sequence minOccurs="0" maxOccurs="unbounded">
            <element name="traceReference" type="unsignedLong"/>
          </sequence>
        </complexType>
      </element>
      <element name="status">
        <simpleType>
          <restriction base="string">
            <enumeration value="Success"/>
            <enumeration value="Failure"/>
          </restriction>
        </simpleType>
      </element>
      <element name="failureReason" minOccurs="0">
        <simpleType>
          <restriction base="string">
            <enumeration value="operation_failed"/>
            <enumeration value="operation_failed_internal_problem"/>
          </restriction>
        </simpleType>
```

```
            </element>
          </sequence>
        </complexType>
      </element>
      <!-- listActivatedTraceJobs Fault -->
      <element name="listActivatedTraceJobsFault">
        <simpleType>
          <restriction base="string">
            <enumeration value="OperationFailed"/>
          </restriction>
        </simpleType>
      </element>
    </schema>
  </types>
  <message name="activateTraceJobRequest">
    <part name="parameter" element="traceIRPData:activateTraceJobRequest"/>
  </message>
  <message name="activateTraceJobResponse">
    <part name="parameter" element="traceIRPData:activateTraceJobResponse"/>
  </message>
  <message name="activateTraceJobFault">
    <part name="parameter" element="traceIRPData:activateTraceJobFault"/>
  </message>
  <message name="deactivateTraceJobRequest">
    <part name="parameter" element="traceIRPData:deactivateTraceJobRequest"/>
  </message>
  <message name="deactivateTraceJobResponse">
    <part name="parameter" element="traceIRPData:deactivateTraceJobResponse"/>
  </message>
  <message name="deactivateTraceJobFault">
    <part name="parameter" element="traceIRPData:deactivateTraceJobFault"/>
  </message>
  <message name="listTraceJobRequest">
    <part name="parameter" element="traceIRPData:listTraceJobRequest"/>
  </message>
  <message name="listTraceJobResponse">
    <part name="parameter" element="traceIRPData:listTraceJobResponse"/>
  </message>
  <message name="listTraceJobFault">
    <part name="parameter" element="traceIRPData:listTraceJobFault"/>
  </message>
  <message name="listActivatedTraceJobsRequest">
    <part name="parameter" element="traceIRPData:listActivatedTraceJobsRequest"/>
  </message>
  <message name="listActivatedTraceJobsResponse">
    <part name="parameter" element="traceIRPData:listActivatedTraceJobsResponse"/>
  </message>
  <message name="listActivatedTraceJobsFault">
    <part name="parameter" element="traceIRPData:listActivatedTraceJobsFault"/>
  </message>
  <portType name="TraceIRPManagement">
    <operation name="activateTraceJob">
      <input message="traceIRPSystem:activateTraceJobRequest"/>
      <output message="traceIRPSystem:activateTraceJobResponse"/>
      <fault name="activateTraceJobFault" message="traceIRPSystem:activateTraceJobFault"/>
    </operation>
    <operation name="deactivateTraceJob">
      <input message="traceIRPSystem:deactivateTraceJobRequest"/>
      <output message="traceIRPSystem:deactivateTraceJobResponse"/>
      <fault name="deactivateTraceJobFault" message="traceIRPSystem:deactivateTraceJobFault"/>
    </operation>
    <operation name="listTraceJob">
      <input message="traceIRPSystem:listTraceJobRequest"/>
      <output message="traceIRPSystem:listTraceJobResponse"/>
      <fault name="listTraceJobFault" message="traceIRPSystem:listTraceJobFault"/>
    </operation>
    <operation name="listActivatedTraceJobs">
      <input message="traceIRPSystem:listActivatedTraceJobsRequest"/>
      <output message="traceIRPSystem:listActivatedTraceJobsResponse"/>
      <fault name="listActivatedTraceJobsFault"
message="traceIRPSystem:listActivatedTraceJobsFault"/>
    </operation>
  </portType>
  <binding name="TraceIRPManagement" type="traceIRPSystem:TraceIRPManagement">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="activateTraceJob">
```

```
      <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#activateTraceJob"
style="document"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
      <fault name="activateTraceJobFault">
        <soap:fault name="activateTraceJobFault" use="literal"/>
      </fault>
    </operation>
    <operation name="deactivateTraceJob">
      <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#deactivateTraceJob"
style="document"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
      <fault name="deactivateTraceJobFault">
        <soap:fault name="deactivateTraceJobFault" use="literal"/>
      </fault>
    </operation>
    <operation name="listTraceJob">
      <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#listTraceJob" style="document"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
      <fault name="listTraceJobFault">
        <soap:fault name="listTraceJobFault" use="literal"/>
      </fault>
    </operation>
    <operation name="listActivatedTraceJobs">
      <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#listActivatedTraceJobs"
style="document"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
      <fault name="listActivatedTraceJobsFault">
        <soap:fault name="listActivatedTraceJobsFault" use="literal"/>
      </fault>
    </operation>
  </binding>
  <service name="TraceIRPService">
    <port name="TraceIRPManagementPort" binding="traceIRPSystem:TraceIRPManagement">
      <soap:address location="http://www.3gpp.org/ftp/specs/archive/32_series/32.446#TraceIRP"/>
    </port>
    <port name="GenericIRPPort" binding="genericIRPSystem:GenericIRPBinding">
      <soap:address location="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRP"/>
    </port>
    <port name="NotificationIRPNtfPort" binding="ntfIRPNtfSystem:NotificationIRPNtf">
      <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPNtf"/>
    </port>
  </service>
</definitions>
```

Annex D (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2010-09	SA#49	SP-100508	--	--	Presentation to SA for Information and Approval	---	1.0.0
2010-10	--	--	--	--	Publication	1.0.0	10.0.0
2010-12	SA#50	SP-100833	001	1	Correcting the Identification of IMS Subscriber Tracing - Align with 32.421 and 32.442	10.0.0	10.1.0
2011-03	SA#51	SP-110093	2	-	Add the missing input parameter of activateTraceJob - Align with 32.442 Trace Management IRP Information Service	10.1.0	10.2.0
2011-03	SA#51	SP-110095	3	-	Add mapping of notifyTraceSessionIdentities in SOAP solution set - Align with 32.442 Trace Management IRP Information Service	10.1.0	10.2.0
2011-03	SA#51	SP-110102	30	1	Extend TraceIRP operation to support MDT configuration - Align with 32.442	10.1.0	10.2.0
2011-06	SA#52	SP-110292	31	1	Add areascope parameter as a MDT configuration	10.2.0	10.3.0
2011-12	SA#54	SP-110715	34	--	Support multiple cells in area based MDT –Align with 32.442	10.3.0	10.4.0
2012-03	SA#55	SP-120053	37	--	Inconsistency correction on trace target -Align with 32.442 IS version number	10.4.0	10.5.0
2012-06	SA#56	SP-120368	40	--	Alignment of the Anonymization parameter with TS 32.422 – Solution Set	10.5.0	10.6.0
2012-09	SA#57	SP-120570	46	--	Add missing threshold parameter for UMTS event triggered measurements	10.6.0	10.7.0

History

Document history		
V10.2.0	April 2011	Publication
V10.3.0	July 2011	Publication
V10.4.0	January 2012	Publication
V10.5.0	March 2012	Publication
V10.6.0	July 2012	Publication
V10.7.0	September 2012	Publication