ETSI TS 132 326 V13.1.0 (2016-08)

TECHNICAL SPECIFICATION

Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Test management Integration Reference Point (IRP): Solution Set (SS) definitions (3GPP TS 32.326 version 13.1.0 Release 13)

Reference
RTS/TSGS-0532326vd10

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

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (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
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

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 (https://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.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

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; Test management Integration Reference Point (IRP), as identified below:

32.321: "Test management Integration Reference Point (IRP); Requirements"

32.322: "Test management Integration Reference Point (IRP): Information Service (IS)"

32.326: "Test management Integration Reference Point (IRP): Solution Set definitions (SS)"

A 3G telecommunication network is composed of a multitude of different Network Elements (NE). For a successful operation of the network the operator must be provided with mechanisms allowing him to manage the network. These management activities can be grouped into several areas: configuration management, fault management, performance management, accounting management and security management.

A management function assisting in different high level management areas such as fault management and performance management is test management. The purpose of testing is to get information about the functionality and performance of the 3G managed network subject to the test.

The present document is part of a TS-family defining the Telecommunication Management (TM) of 3G systems. The TM principles are described in 3GPP TS 32.101 [2]. The TM architecture is described in 3GPP TS 32.102 [3]. The other specifications define the interface (Itf-N) between the managing system (manager), which is in general the Network Manager (NM) and the managed system (agent), which is either an Element Manager (EM) or the managed NE itself. The Itf-N is composed of a number of integration reference points (IRPs) defining the information in the agent that is visible for the manager, the operations that the manager may perform on this information and the notifications that are sent from the agent to the manager. One of these IRPs is the Test Management IRP.

Each IRP is specified by the requirements part, the IS part and SS part.

1 Scope

The present document contains the Solution Sets for the IRP whose semantics is specified in Test management IRP IS (3GPP TS 32.322 [5]).

This Solution Set specification is related to 3GPP TS 32.322 V13.0.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 32.322: "Telecommunication management; Test management Integration Reference Point (IRP): Information Service (IS)".

[6] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".

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

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

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

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

[11] 3GPP TS 32.306: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Solution Set (SS) definitions".

[12] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management; Information Service (IS)".

[13] 3GPP TS 32.336: "Telecommunication management; Notification Log (NL) Integration Reference Point (IRP): Solution Set (SS) definitions".

[14] 3GPP TS 32.331: "Telecommunication management; Notification Log (NL) Integration Reference Point (IRP): Requirements".

[15] Object Management Group 98 (November 1998): "*Notification Service: Joint Revised Submission OMG TC Document telecom/98-11-01*". http://www.omg.org/technology/documents/

[16] G CORBA Services (November 1996): "Common Object Services Specification" (clause 4 contains the Event Service specification). http://www.omg.org/technology/documents/

[17] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP) management; Requirements".

3 Definitions, symbols 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], 3GPP TS 32.331 [14] and 3GPP TS 32.322 [5] apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Test Category: one or more tests sharing a common purpose and similar characteristics

Tester Object (TO): managed object that is instantiated for the purpose of monitoring and controlling a test invocation Each test invocation has one associated TO. TOs are created and deleted by managed objects with TARR functionality.

IRP document version number string:

The IRP document version number (sometimes called "IRP version" or "version number") string is used to identify the present document. The definition of "IRP document version number string" in 3GPP TS 32.311 [17] provides the rule to derive such a string.

This string is returned in `get_test_management_IRP_versions` method and is carried in the first field of the notification header of all notifications related to Test Management IRP. This string is also returned in `get_notification_categories` method of the Notification IRPAgent, in case that IRPAgent is responsible for emitting notifications related to Test Management IRP.

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].

CORBA	Common Object Request Broker Architecture
EM	Element Manager
GSM	Global System for Mobile communications
IDL	Interface Definition Language
IRP	Integration Reference Point
IOC	Information Object Class
IS	Information Service
NE	Network Element
NL	Notification Log
NM	Network Manager
OMG	Object Management Group
QoS	Quality of Service
SS	Solution Set
TARR	Test Action Request Receiver
TM	Test Management
TMIRP	Test Management IRP
TO	Tester Object
XML	eXtensible Markup Language

4 Solution Set Definitions

This specification defines the following 3GPP Test 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 Test management IRP: Information Service (TS 32.322 [5]).

A.1 Architectural features

The overall architectural feature of Test Management IRP is specified in 3GPP TS 32.322 [5]. This clause specifies features that are specific to the CORBA SS.

A.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [6].

A.1.2 Notification Services

In implementations of CORBA SS, IRPAgent conveys Test Management information to IRPManager via OMG Notification Service (OMG Notification Service [15]).

OMG Event Service [16] provides event routing and distribution capabilities. OMG Notification Service provides, in addition to Event Service, event filtering and Quality of Service (QoS) as well.

A necessary and sufficient sub set of OMG Notification Services shall be used to support `TestManagementIRPNotifications` notifications as specified in 3GPP TS 32.322 [5].

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.

The Notification CORBA SS in 3GPP TS 32.306 [11] 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 TestManagementIRPNotification Interface

OMG CORBA Notification push operation is used to realise the notification of `TestManagementIRPNotifications`. 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 [15]). 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

The Test Management IRP IS in 3GPP TS 32.322 [5] defines semantics of operation and notification visible across the Test Management IRP. Table A.2.1 indicates mapping of these operations and notifications to their equivalents defined in the present SS.

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

IS Operations/ notification TS 32.322 [5]	SS Method	Qualifier
initiateTests	initiate_tests	M
terminateTests	terminate_tests	M
monitorTest	monitor_test	M
getIRPVersion	get_test_management_IRP_versions	M
getOperationProfile (see note)	get_test_management_IRP_operation_profile	O
getNotificationProfile (see note)	get_test_management_IRP_notification_profile	O
notifyTestResult	push_structured_event (See subclause A.1.5)	M
NOTE: This operation is of ManagedGenericIRP IOC specified in 3GPP TS 32.312 [12]. The TestManagementIRP IOC of [5] inherits from it.		

A.2.2 Operation parameter mapping

The Test Management IRP IS in 3GPP TS 32.322 [5] defines semantics of parameters carried in operations across the Test Management IRP. The tables below show the mapping of these parameters, as per operation, to their equivalents defined in this SS.

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

IS Operation parameter	SS Method parameter	Qualifier
testInvocationInitiator	TestManagementIRPConstDefs::TestInvocationInitiator test_invocation_initiator	M
toBeInitiatedTests	TestManagementIRPConstDefs::ToBeInitiatedTestSeq to_be_initiated_test_seq	M
response	TestManagementIRPConstDefs::InitiateTestsResponse Exceptions: InitiateTests, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

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

IS Operation parameter	SS Method parameter	Qualifier
toBeTerminatedTests	TestManagementIRPConstDefs::ToBeTerminatedTestSeq to_be_terminated_test_seq	M
response	TestManagementIRPConstDefs::TerminateTestsResponse Exceptions: TerminateTests, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

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

IS Operation parameter	SS Method parameter	Qualifier
toBeMonitoredTO	TestManagementIRPConstDefs::ToBeMonitoredTO to_be_monitored_TO	M
monitoredAttributeValues	TestManagementIRPConstDefs::TOAttributes tO_attributes	M
error	ManagedGenericIRPConstDefs::Signal Exceptions: MonitorTest, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

A.2.3 Notification parameter mapping

The Test Management IRP IS in 3GPP TS 32.322 [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 [15]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [15], is:

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

Table A.2.3 lists in the second column all OMG Structured Event attributes. The first column identifies notification parameters defined in 3GPP TS 32.322 [5], Test Management: Information Service (IS).

Table A.2.3: Mapping for notifyTestResult

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
There is no corresponding IS parameter.	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 the NOTIFY_TM_TEST_RESULT of module of TestManagementIRPConstDefs.
There is no corresponding IS parameter.	event_name	M	It carries no information.
There is no corresponding IS parameter.	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 TS 32.306 [11] Notification IRP: Solution Set (SS) definitions.
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 TS 32.306 [11] Notification IRP: Solution Set (SS) definitions.
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 TS 32.306 [11] Notification IRP: Solution Set (SS) definitions.
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 TS 32.306 [11] Notification IRP: Solution Set (SS) definitions.
testInvocationInitiator	One NV pair of filterable _body_fields	M	Name of NV pair is the TEST_INVOCATION_INITIATOR of module TestManagementIRPConstDefs. Value of NV pair is a string.
testOutcome	One NV pair of remaining_body	O	Name of NV pair is the TEST_OUTCOME of module TestManagementIRPConstDefs. Value of NV pair is a enum {Inconclusive, Pass, Fail, TimeOut, PrematureTermination}

mORT	One NV pair of remaining_body	O	Name of NV pair is the MORT of module TestManagementIRPConstDefs. Value of NV pair is a string.
proposedRepairActions	One NV pair of remaining_body	O	Name of NV pair is the PROPOSED_REPAIR_ACTIONS of module TestManagementIRPConstDefs. Value of NV pair is a string.
additionalInformation	One NV pair of remaining_body	O	Name of NV pair is the ADDITIONAL_INFORMATION of module TestManagementIRPConstDefs. Value of NV pair is a string.
fileReference	One NV pair of remaining_body	M (Note 1)	Name of NV pair is the FILE_REFERENCE of module TestManagementIRPConstDefs. Value of NV pair is a string.
fileExpiryDate	One NV pair of remaining_body	M (Note 2)	Name of NV pair is the FILE_EXPIRY_DATE of module TestManagementIRPConstDefs. Value of NV pair is a string.
Note 1: It shall contain no information or be absent if there is no test result captured in a file. It shall contain information if the test results are captured in a file. Note 2: It shall contain no information or be absent if fileReference carries no information or absent. Otherwise, it shall contain a valid future date and time.			

A.3 Solution Set definitions

A.3.1 IDL definition structure

Clause A.3.2 defines the constants and types used by the Test management IRP.

Clause A.3.3 defines the operations which are performed by the Test management IRP agent.

Clause A.3.4 defines the notifications which are emitted by the Test management IRP agent.

A.3.2 IDL specification "TestManagementIRPConstDefs.idl"

```
// File: TestManagementIRPConstDefs.idl


#ifndef _TESTMANAGEMENTIRPCONSTDEFS_IDL_
#define _TESTMANAGEMENTIRPCONSTDEFS_IDL_

#include "CosNotification.idl"
#include "ManagedGenericIRPConstDefs.idl"

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

/* ## Module: TestManagementIRPConstDefs
This module contains commonly used definitions for Test Management IRP
======================================================================
*/
module TestManagementIRPConstDefs
{
   /*
   This defines the notification type of this Test Management
      IRP.
   */
   const string NOTIFY_TM_TEST_RESULT = "x1";


   /*
   This enum defines the test state
   */
   enum TestStateType {
      NotInitialized,
      Idle,
      Initializing,
      Testing,
      Terminating,
      Disabled
   };

   /*
   This enum defines the test outcome
   */
   enum TestOutcomeType {
      Inconclusive,
      Pass,
      Fail,
      TimeOut,
      PrematureTermination
   };


   /*
   This block defines notification attributes of this IRP.
   These attribute values should not clash with those used
       in Notification header (see IDL of Notification IRP).
   */
   interface AttributeNameValue
   {
    const string TEST_INVOCATION_INITIATOR = "f";
    const string TEST_INVOCATION_ID = "g";
    const string TEST_ACTUAL_START_TIME = "h";
    const string TEST_ACTUAL_STOP_TIME = "i";
    const string TEST_OUTCOME = "j";
    const string MORT = "k";
    const string PROPOSED_REPAIR_ACTIONS = "l";
    const string ADDITIONAL_INFORMATION = "m";
    const string FILE_REFERENCE = "n";
    const string FILE_EXPIRY_DATE = "o";
    const string TEST_SOURCE_ADDRESS = "p";
    const string TEST_DESTINATION_ADDRESS = "q";
    const string TEST_LOOPBACK_ADDRESS = "r";
    const string TEST_PACKET_INFORMATION = "s";
    };

   typedef string TestInvocationInitiator;
   typedef string ToBeMonitoredTO;
   typedef CosNotification::PropertySeq NVPairs;

   /*
   Define a seq of to-be-initiated-test
   */
   struct ToBeInitiatedTest
   {
       unsigned long max_testing_state_duration;//seconds;0->no limit
       string  toBeTestedMORT; //MORT DN
       string  tOClass;        //Tester object class
       string  tODN;           //Tester object DN
       NVPairs tONVPair;       //Tester object attributes in NV pairs
   };
   typedef sequence <ToBeInitiatedTest> ToBeInitiatedTestSeq;

   /*
   Define the structure returned by initiate_tests
   */
   struct InitiateTestsResponseElement
   {
      // If failureReason is NULL, the test is initiated successfully and
      //     testInvocationId contains the invocation id. In case the tester object name is not
      //     provided in the request, it shall be carried by testerObjectDN. In case the tester
      //     object name is provided in the request tODN shall be NULL.
      // Else, the test initiation fails and failureReason contains
      //     the failure reason and testInvocationId contains garbage.
      string failureReason;
      string testInvocationId;
      string tODN;
   };
   typedef sequence <InitiateTestsResponseElement> InitiateTestsResponse;



   /*
   Define a seq of to-be-terminated-test
   */
   typedef string TestInvocationId;
   typedef sequence <TestInvocationId> ToBeTerminatedTestSeq;

   /*
   Define the structure returned by terminate_tests
   */
   struct TerminateTestsResponseElement
   {
      // If failureReason is NULL, the test has terminated successfully and
      //     testInvocationId identifies the terminated invocation.
      // Else, the test termination fails and failureReason contains
      //     the failure reason and testInvocationId contains garbage.
      string failureReason;
      string testInvocationId;
   };
   typedef sequence <TerminateTestsResponseElement> TerminateTestsResponse;

   /*
   Define the structure of a TOAttributes.
   */
   struct TOAttributes
   {
       TestStateType   testState;
       TestOutcomeType testOutcome;
       NVPairs         attributesInNVPairs;
   };

};
#endif // _TESTMANAGEMENTIRPCONSTDEFS_IDL_
```

A.3.3 IDL specification "TestManagementIRPSystem.idl"

```
// File: TestManagementIRPSystem.idl
```

```
#ifndef _TESTMANAGEMENTIRPSYSTEM_IDL_
#define _TESTMANAGEMENTIRPSYSTEM_IDL_

#include "TestManagementIRPConstDefs.idl"
#include "ManagedGenericIRPSystem.idl"

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

/* ## Module: TestManagementIRPSystem
This module contains the specification of all methods of TestManagement IRP Agent.
================================================================
*/
module TestManagementIRPSystem
{
   /*
   System may fail to complete an operation.  System can provide reason
   to qualify the failed reason.  The semantics carried in reason
   is outside the scope of this IRP.
   */
   exception GetTestManagementIRPVersions { string reason; };
   exception GetTestManagementIRPOperationsProfile { string reason; };
   exception GetTestManagementIRPNotificationProfile { string reason; };
   exception InitiateTests { string reason; };
   exception TerminateTests { string reason; };
   exception MonitorTest { string reason; };


   interface TestManagementIRP
   {
      /*
      Return the list of all supported TestManagement IRP versions.
      */
      ManagedGenericIRPConstDefs::VersionNumberSet
      get_Test_Management_IRP_versions (
      )
      raises (GetTestManagementIRPVersions);


      /*
      Return the list of all supported operations and their supported
      parameters for a specific TestManagement IRP version.
      */
      ManagedGenericIRPConstDefs::MethodList
      get_Test_Management_IRP_operations_profile (
         in ManagedGenericIRPConstDefs::VersionNumber
             test_management_irp_version
      )
      raises (GetTestManagementIRPOperationsProfile,
              ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);

      /*
      Return the list of all supported notifications and their supported
      parameters for a specific TestManagement IRP version.
      */
      ManagedGenericIRPConstDefs::MethodList
      get_Test_Management_IRP_notification_profile (
         in ManagedGenericIRPConstDefs::VersionNumber
            test_management_irp_version
      )
      raises (GetTestManagementIRPNotificationProfile,
              ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);


      /*
      Request to initiate tests.
      */
      TestManagementIRPConstDefs::InitiateTestsResponse
      initiate_tests (
         in TestManagementIRPConstDefs::TestInvocationInitiator
            test_invocation_initiator,
         in TestManagementIRPConstDefs::ToBeInitiatedTestSeq
            to_be_initiated_test_seq
      )
      raises (InitiateTests,
              ManagedGenericIRPSystem::InvalidParameter);


      /*
      Request to terminate tests.
      */
      TestManagementIRPConstDefs::TerminateTestsResponse
      terminate_tests (
         in TestManagementIRPConstDefs::ToBeTerminatedTestSeq
             to_be_terminated_test_seq
      )
      raises (TerminateTests,
              ManagedGenericIRPSystem::InvalidParameter);


      /*
      Request test info (to monitor a test).
      */
      ManagedGenericIRPConstDefs::Signal monitor_test (
         in TestManagementIRPConstDefs::ToBeMonitoredTO
             to_be_monitored_TO,
         out TestManagementIRPConstDefs::TOAttributes tO_attributes
      )
      raises (MonitorTest,
              ManagedGenericIRPSystem::InvalidParameter);




   };
};

#endif // _TESTMANAGEMENTIRPSYSTEM_IDL_
```

A.3.4 IDL specification "TestManagementIRPNotifications.idl"

```
// File: TestManagementIRPNotifications.idl

#ifndef _TESTMANAGEMENTIRPNOTIFICATIONS_IDL
#define _TESTMANAGEMENTIRPNOTIFICATIONS_IDL

#include "TestManagementIRPConstDefs.idl"
#include "NotificationIRPConstDefs.idl"
#include "NotificationIRPNotifications.idl"

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

/* ## Module: TestManagementIRPNotifications
This module contains the specification of all notifications of Test Management IRP Agent.
=============================================================================
*/

module TestManagementIRPNotificationsfDefs
{
   /**
   *  Constant definitions for the notifyTestResult notification
   */
   interface notifyTestResult: NotificationIRPNotifications::Notify
   {
     const string EVENT_TYPE = "notifyTestResult";

     /**
     * This constant defines the name of the  period property,
     * which is transported in the filterable_body fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: TestInvocationInitiator.
     */
     const string TEST_INVOCATION_INITIATOR =
         TestManagementIRPConstDefs::AttributeNameValue::TEST_INVOCATION_INITIATOR;

     /*
     * This constant defines the name of the
     * TestInvocationId property,
     * which is transported in the remaining body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: TestInvocationId.
     */
     const string TEST_INVOCATION_ID =
         TestManagementIRPConstDefs::AttributeNameValue::TEST_INVOCATION_ID;

     /*
     * This constant defines the name of the
     * TestActualStartTime property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: TestActualStartTime.
     */
      const string TEST_ACTUAL_START_TIME =
          TestManagementIRPConstDefs::AttributeNameValue::TEST_ACTUAL_START_TIME;

     /*
     * This constant defines the name of the
     * TestActualStopTime property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: TestActualStopTime.
     */
      const string TEST_ACTUAL_STOP_TIME =
          TestManagementIRPConstDefs::AttributeNameValue::TEST_ACTUAL_STOP_TIME;

     /*
     * This constant defines the name of the
     * testOutcome property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: testOutcome.
     */
      const string TEST_OUTCOME = TestManagementIRPConstDefs::AttributeNameValue::TEST_OUTCOME;

     /*
     * This constant defines the name of the
     * MORT property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs::MORT.
     */
      const string MORT = TestManagementIRPConstDefs::AttributeNameValue::MORT;

     /*
     * This constant defines the name of the
     * ProposedRepairActions property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs::ProposedRepairActions.
     */
      const string PROPOSED_REPAIR_ACTIONS =
          TestManagementIRPConstDefs::AttributeNameValue::PROPOSED_REPAIR_ACTIONS;

     /*
     * This constant defines the name of the
     * AdditionalInformation property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: AdditionalInformation.
     */
      const string ADDITIONAL_INFORMATION =
          TestManagementIRPConstDefs::AttributeNameValue::ADDITIONAL_INFORMATION;

     /*
     * This constant defines the name of the
     * FileReference property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: FdditionalInformation.
     */
      const string FILE_REFERENCE = TestManagementIRPConstDefs::AttributeNameValue::FILE_REFERENCE;

     /*
     * This constant defines the name of the
     * FileExpiryDate property,
     * which is transported in the remaining_body
     * fields.
     * The data type for the value of this property
     * is TestManagementIRPConstDefs:: FileExpiryDate.
     */
      const string FILE_EXPIRY_DATE =
          TestManagementIRPConstDefs::AttributeNameValue::FILE_EXPIRY_DATE;


   };

};

#endif // _TESTMANAGEMENTIRPNOTIFICATIONS_IDL_
```

Annex B (normative): XML Definitions

This annex contains the XML Definitions for the Test management Integration Reference Point (Test management IRP) as it applies to Itf-N, in accordance with Test management IRP IS definitions [5] as well as Notification Log IRP XML Definitions [13].

Apart from being used for the Notification Log, the XML definitions for Test management IRP notifications are also used by the Test management IRP SOAP SS.

B.1 Architectural Features

The overall architectural feature of Test management IRP is specified in 3G TS 32.322 [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 [6].

B.1.2 Notification Services

This annex defines the XML syntax of Test management IRP notifications that is to be used for the Test 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 [13].

B.1.3 IOC Definitions

This annex defines the XML syntax for the IOC definitions of the Test management IRP IS [5], which are used by the XML definitions for the Test management IRP notifications and the Test 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 Test management IRP notifications as defined in [5]. These definitions are to be used for the Test 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 [13], as well as considerations for NL IRP XML File Name Conventions defined therein.

B.3.2 Graphical Representation

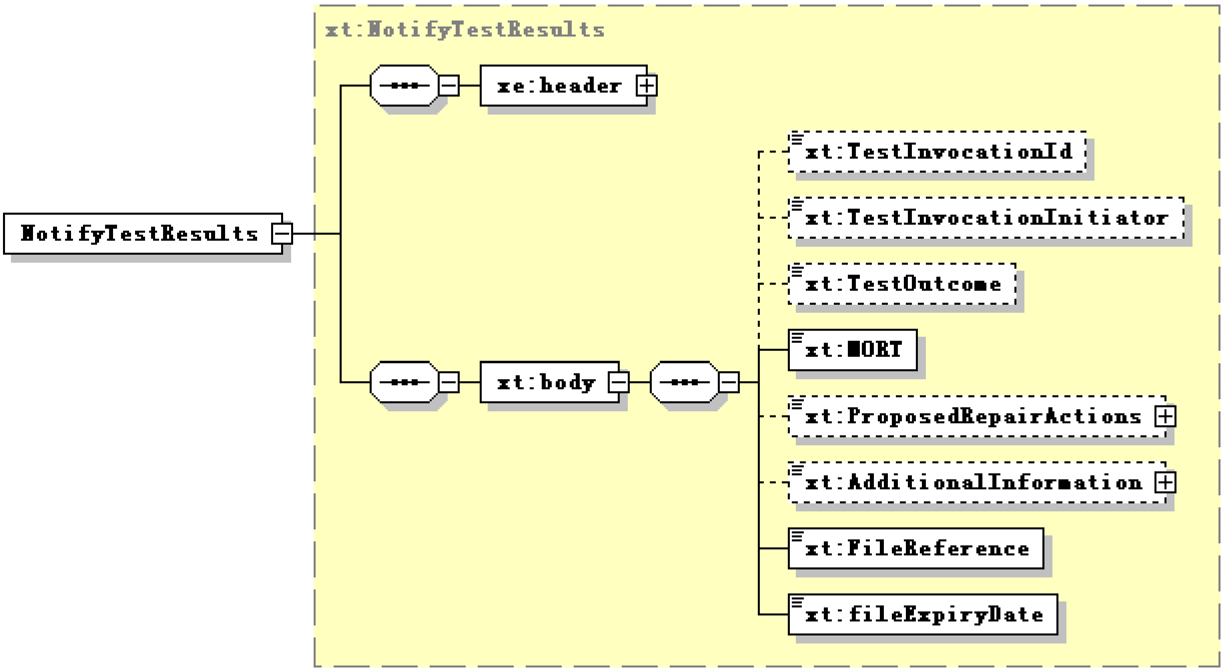

Figure B.3.2

NOTE: The use of XML schema key word "sequence" to support IS-defined set (not sequence) is for the purpose of XML processor efficiency. This shall not imply the use of "sequence" in other technology.

B.3.3 XML Schema 'tMIRPNotif.xsd'

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
    3GPP TS 32.326 TMIRP Notification XML Schema
    tMIRPNotif.xsd
-->
<schema xmlns:xt="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#tMIRPNotif"
xmlns:xe="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification"
xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#tMIRPNotif"
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <import namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification" />
    <simpleType name="TestInvocationId">
        <restriction base="string"/>
    </simpleType>
    <simpleType name="TestOutcome">
        <restriction base="string">
            <enumeration value="pass"/>
            <enumeration value="fail"/>
            <enumeration value="inconclusive"/>
            <enumeration value="timed-out"/>
            <enumeration value="premature-termination"/>
        </restriction>
    </simpleType>
    <complexType name="NotifyTestResults">
        <complexContent>
            <extension base="xe:Notification">
                <sequence>
                    <element name="body">
                        <complexType>
                            <sequence>
                                <element name="fileReference" type="string"/>
                                <element name="fileExpiryDate" type="dateTime"/>
                                <element name="testInvocationId" type="xt:TestInvocationId"
minOccurs="0"/>
                                <element name="testInvocationInitiator" type="string"
minOccurs="0"/>
                                <element name="testOutcome" type="xt:TestOutcome" minOccurs="0"/>
                                <element name="mORT" type="string" minOccurs="0"/>
                                <element name="proposedRepairActions" type="anyType" minOccurs="0"/>
                                <element name="additionalInformation" type="anyType" minOccurs="0"/>
                            </sequence>
                        </complexType>
                    </element>
                </sequence>
            </extension>
        </complexContent>
    </complexType>
    <element name="NotifyTestResults" type="xt:NotifyTestResults"/>
</schema>
```

Annex C (normative): SOAP Solution Set

This annex specifies the SOAP Solution Set for the IRP whose semantics are specified in Test management IRP: Information Service (3GPP TS 32.322 [5]).

C.1 Architectural features

The overall architectural feature of the Test management IRP is specified in 3GPP TS 32.322 [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 [6].

C.1.2 Notification Services

The Test management IRP SOAP SS uses the Notification IRP SOAP SS of 3GPP TS 32.306 [11]. The IRPAgent shall support the push interface model, which means that the IRPAgent sends notifications to the IRPManager as soon as new events occur. The IRPManager does not need to check ("pull") for events.

Relevant definitions are imported from the Test management IRP XML definitions in Annex B.

C.1.3 Supported W3C specifications

The SOAP 1.1 specification [7] and WSDL 1.1 specification [9] are supported.

The SOAP 1.2 specification [10] 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 [8]). IRPAgents may throw a FilterComplexityLimit fault when a given filter is too complex.

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/
tmIRPSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPSystem
tmIRPData	http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPData
genericNrm	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 Test management IRP IS (3GPP TS 32.322 [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 32.322 [5]	SS Operations	SS Port	Qualifier
initiateTests	initiateTests	TMIRPControlOperationsPort	M
terminateTests	terminateTests	TMIRPControlOperationsPort	M
monitorTest	monitorTest	TMIRPMonitorOperationsPort	M
notifyFilePreparationError	notify (note 1)	NotificationIRPNtfPort	M
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 [11].			

C.2.2 Operation parameter mapping

The Test management IRP IS (3GPP TS 32.322 [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 `initiateTests` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
testInvocationInitiator	testInvocationInitiator	M
toBeInitiatedTests	toBeInitiatedTests	M
response	response	M

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

IS Operation parameter	SS Method parameter	Qualifier
toBeTerminatedTests	toBeTerminatedTests	M
response	response	M

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

IS Operation parameter	SS Method parameter	Qualifier
toBeMonitoredTO	toBeMonitoredTO	M
monitoredAttributeValues	monitoredAttributeValues	M
error	error	M

C.2.3 Notification parameter mapping

The Test management IRP Notifications are defined in Annex B.

C.3 Solution Set definitions

C.3.1 WSDL definition structure

Clause C.3.2 provides a graphical representation of the Test management IRP service.

Clause C.3.3 defines the services which are supported the Test 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.

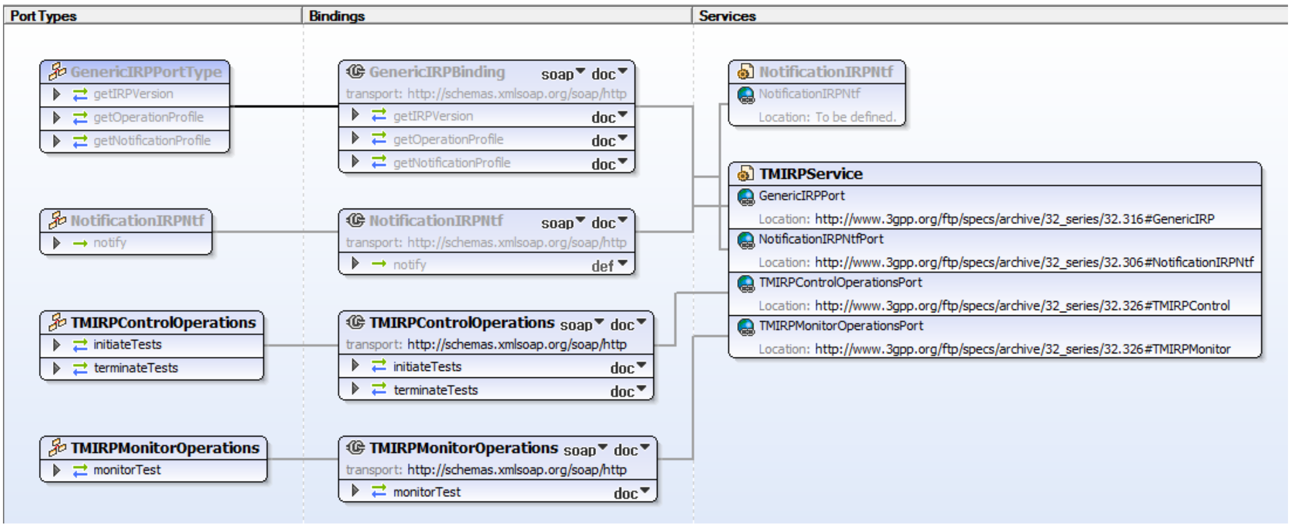
Port Types
Bindings
Services
GenericIRPPortType
getIRPVersion
getOperationProfile
getNotificationProfile
NotificationIRPNtf
notify
TMIRPControlOperations
initiateTests
terminateTests
TMIRPMonitorOperations
monitorTest
GenericIRPBinding soap doc
transport: http://schemas.xmlsoap.org/soap/http
getIRPVersion doc
getOperationProfile doc
getNotificationProfile doc
NotificationIRPNtf soap doc
transport: http://schemas.xmlsoap.org/soap/http
notify def
TMIRPControlOperations soap doc
transport: http://schemas.xmlsoap.org/soap/http
initiateTests doc
terminateTests doc
TMIRPMonitorOperations soap doc
transport: http://schemas.xmlsoap.org/soap/http
monitorTest doc
NotificationIRPNtf
NotificationIRPNtf
Location: To be defined.
TMIRPService
GenericIRPPort
Location: http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRP
NotificationIRPNtfPort
Location: http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPNtf
TMIRPControlOperationsPort
Location: http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPControl
TMIRPMonitorOperationsPort
Location: http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPMonitor

Figure C.3.2: Test management IRP SOAP Solution Set WSDL structure

C.3.3 WSDL specification 'TMIRPSystem.wsdl'

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
   3GPP TS 32.326 Test Management IRP SOAP Solution Set
-->
<definitions
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tmIRPSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPSystem"
xmlns:tmIRPData="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPData"
xmlns:genericNrm="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/32.326#TMIRPSystem">
   <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/32.306#NotificationIRPNtfSystem"/>
   <types>
      <schema targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPData"
xmlns="http://www.w3.org/2001/XMLSchema">
         <import namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm"/>
         <!--  attributeNameValue Type -->
         <complexType name="attributeNameValueType">
            <sequence>
               <element name="attributeName" type="string"/>
               <element name="attributeValue" type="string"/>
            </sequence>
         </complexType>
         <!--  operationResult Type -->
         <simpleType name="operationResultType">
            <restriction base="string">
               <enumeration value="operationFailedEntirely"/>
               <enumeration value="operationFailedPartly"/>
               <enumeration value="operationSucceeded"/>
            </restriction>
         </simpleType>
         <!-- initiateTests Request -->
         <element name="initiateTestsRequest">
            <complexType>
               <sequence>
                  <element name="testInvocationInitiator" type="genericNrm:dn"/>
                  <element name="toBeInitiatedTests">
                     <complexType>
                        <sequence maxOccurs="unbounded">
                           <choice minOccurs="0">
                              <element name="maxTestingStateDuration" type="duration"/>
                              <element name="noLimitTestingStateDuration"/>
                           </choice>
                           <element name="toBeTestedMORT" type="genericNrm:dn" minOccurs="0"/>
                           <element name="testerObjectClass" type="genericNrm:dn"/>
                           <element name="testerObjectName" type="string" minOccurs="0"/>
                           <element name="testerObjectInitialAttributeList" minOccurs="0">
                              <complexType>
                                 <sequence maxOccurs="unbounded">
                                    <element name="initialAttribute"
type="tmIRPData:attributeNameValueType"/>
                                 </sequence>
                              </complexType>
                           </element>
                        </sequence>
                     </complexType>
                  </element>
               </sequence>
            </complexType>
         </element>
         <!-- initiateTests Response -->
         <element name="initiateTestsResponse">
            <complexType>
               <sequence>
                  <element name="initiateTestsResult" type="tmIRPData:operationResultType"/>
                  <element name="response">
                     <complexType>
                        <sequence maxOccurs="unbounded">
                           <choice>
```

```
                              <element name="testInitiated">
                                 <complexType>
                                    <sequence>
                                       <element name="testInvocationId" type="string"/>
                                       <element name="testerObjectName" type="string"
minOccurs="0"/>
                                    </sequence>
                                 </complexType>
                              </element>
                              <element name="testNotInitiated">
                                 <complexType>
                                    <sequence>
                                       <element name="failureReason">
                                          <simpleType>
                                             <restriction base="string">
                                                <enumeration value="TOClassNotExisting"/>
                                                <enumeration value="MORTNotExisting"/>
                                                <enumeration value="MORTNotAvailable"/>
                                                <enumeration
value="operation_failed_invalid_input_parameter"/>
                                                <enumeration
value="operation_failed_unsupported_optional_input_parameter_maxTestingStateDuration"/>
                                                <enumeration
value="operation_failed_unsupported_optional_input_parameter_noLimitTestingStateDuration"/>
                                                <enumeration
value="operation_failed_unsupported_optional_input_parameter_toBeTestedMORT"/>
                                                <enumeration
value="operation_failed_unsupported_optional_input_parameter_testerObjectName"/>
                                                <enumeration
value="operation_failed_unsupported_optional_input_parameter_testerObjectInitialAttributeList"/>
                                                <enumeration
value="operation_failed_internal_problem"/>
                                             </restriction>
                                          </simpleType>
                                       </element>
                                    </sequence>
                                 </complexType>
                              </element>
                           </choice>
                        </sequence>
                     </complexType>
                  </element>
               </sequence>
            </complexType>
         </element>
         <!-- initiateTests Fault -->
         <element name="initiateTestsFault">
            <simpleType>
               <restriction base="string">
                  <enumeration value="OperationFailed"/>
               </restriction>
            </simpleType>
         </element>
         <!-- terminateTests Request -->
         <element name="terminateTestsRequest">
            <complexType>
               <sequence maxOccurs="unbounded">
                  <element name="testInvocationId" type="string"/>
               </sequence>
            </complexType>
         </element>
         <!-- terminateTests Response -->
         <element name="terminateTestsResponse">
            <complexType>
               <sequence>
                  <element name="terminateTestsResult" type="tmIRPData:operationResultType"/>
                  <element name="response">
                     <complexType>
                        <sequence maxOccurs="unbounded">
                           <choice>
                              <element name="testTerminated">
                                 <complexType>
                                    <sequence>
                                       <element name="testInvocationId" type="string"/>
                                    </sequence>
                                 </complexType>
                              </element>
                              <element name="testNotTerminated">
```

```
                                 <complexType>
                                    <sequence>
                                       <element name="testInvocationId" type="string"/>
                                       <element name="failureReason">
                                          <simpleType>
                                             <restriction base="string">
                                                <enumeration value="testInvocationIdNotExisting"/>
                                                <enumeration
value="operation_failed_invalid_input_parameter"/>
                                                <enumeration
value="operation_failed_internal_problem"/>
                                             </restriction>
                                          </simpleType>
                                       </element>
                                    </sequence>
                                 </complexType>
                              </element>
                           </choice>
                        </sequence>
                     </complexType>
                  </element>
               </sequence>
            </complexType>
         </element>
         <!-- terminateTests Fault -->
         <element name="terminateTestsFault">
            <simpleType>
               <restriction base="string">
                  <enumeration value="OperationFailed"/>
               </restriction>
            </simpleType>
         </element>
         <!-- monitorTest Request -->
         <element name="monitorTestRequest">
            <complexType>
               <sequence>
                  <element name="toBeMonitoredTO" type="genericNrm:dn"/>
               </sequence>
            </complexType>
         </element>
         <!-- monitorTest Response -->
         <element name="monitorTestResponse">
            <complexType>
               <sequence>
                  <element name="monitorTestResult" type="tmIRPData:operationResultType"/>
                  <element name="monitoredAttributeValues" minOccurs="0">
                     <complexType>
                        <sequence>
                           <element name="testState">
                              <simpleType>
                                 <restriction base="string">
                                    <enumeration value="notInitialized"/>
                                    <enumeration value="idle"/>
                                    <enumeration value="initializing"/>
                                    <enumeration value="testing"/>
                                    <enumeration value="terminating"/>
                                    <enumeration value="disabled"/>
                                 </restriction>
                              </simpleType>
                           </element>
                           <element name="testOutcome">
                              <simpleType>
                                 <restriction base="string">
                                    <enumeration value="pass"/>
                                    <enumeration value="fail"/>
                                    <enumeration value="inconclusive"/>
                                    <enumeration value="timed-out"/>
                                    <enumeration value="premature-termination"/>
                                 </restriction>
                              </simpleType>
                           </element>
                           <sequence minOccurs="0" maxOccurs="unbounded">
                              <element name="otherAttribute"
type="tmIRPData:attributeNameValueType"/>
                           </sequence>
                        </sequence>
                     </complexType>
                  </element>
                  <element name="error">
                     <complexType>
                        <sequence>
                           <element name="failureReason">
                              <simpleType>
                                 <restriction base="string">
                                    <enumeration value="TOInstanceNotExisting"/>
                                    <enumeration value="errorReadingAttribute"/>
                                    <enumeration value="operation_failed_invalid_input_parameter"/>
                                    <enumeration value="operation_failed_internal_problem"/>
                                 </restriction>
                              </simpleType>
                           </element>
                           <element name="errorInfo" type="string"/>
                        </sequence>
                     </complexType>
                  </element>
               </sequence>
            </complexType>
         </element>
         <!-- monitorTest Fault -->
         <element name="monitorTestFault">
            <simpleType>
               <restriction base="string">
                  <enumeration value="OperationFailed"/>
               </restriction>
            </simpleType>
         </element>
      </schema>
   </types>
   <message name="initiateTestsRequest">
      <part name="parameter" element="tmIRPData:initiateTestsRequest"/>
   </message>
   <message name="initiateTestsResponse">
      <part name="parameter" element="tmIRPData:initiateTestsResponse"/>
   </message>
   <message name="initiateTestsFault">
      <part name="parameter" element="tmIRPData:initiateTestsFault"/>
   </message>
   <message name="terminateTestsRequest">
      <part name="parameter" element="tmIRPData:terminateTestsRequest"/>
   </message>
   <message name="terminateTestsResponse">
      <part name="parameter" element="tmIRPData:terminateTestsResponse"/>
   </message>
   <message name="terminateTestsFault">
      <part name="parameter" element="tmIRPData:terminateTestsFault"/>
   </message>
   <message name="monitorTestRequest">
      <part name="parameter" element="tmIRPData:monitorTestRequest"/>
   </message>
   <message name="monitorTestResponse">
      <part name="parameter" element="tmIRPData:monitorTestResponse"/>
   </message>
   <message name="monitorTestFault">
      <part name="parameter" element="tmIRPData:monitorTestFault"/>
   </message>
   <portType name="TMIRPControlOperations">
      <operation name="initiateTests">
         <input message="tmIRPSystem:initiateTestsRequest"/>
         <output message="tmIRPSystem:initiateTestsResponse"/>
         <fault name="initiateTestsFault" message="tmIRPSystem:initiateTestsFault"/>
      </operation>
      <operation name="terminateTests">
         <input message="tmIRPSystem:terminateTestsRequest"/>
         <output message="tmIRPSystem:terminateTestsResponse"/>
         <fault name="terminateTestsFault" message="tmIRPSystem:terminateTestsFault"/>
      </operation>
   </portType>
   <portType name="TMIRPMonitorOperations">
      <operation name="monitorTest">
         <input message="tmIRPSystem:monitorTestRequest"/>
         <output message="tmIRPSystem:monitorTestResponse"/>
         <fault name="monitorTestFault" message="tmIRPSystem:monitorTestFault"/>
      </operation>
   </portType>
   <binding name="TMIRPControlOperations" type="tmIRPSystem:TMIRPControlOperations">
      <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
      <operation name="initiateTests">
         <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#initiateTests" style="document"/>
         <input>
            <soap:body use="literal"/>
         </input>
         <output>
            <soap:body use="literal"/>
         </output>
         <fault name="initiateTestsFault">
            <soap:fault name="initiateTestsFault" use="literal"/>
         </fault>
      </operation>
      <operation name="terminateTests">
         <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#terminateTests"
style="document"/>
         <input>
            <soap:body use="literal"/>
         </input>
         <output>
            <soap:body use="literal"/>
         </output>
         <fault name="terminateTestsFault">
            <soap:fault name="terminateTestsFault" use="literal"/>
         </fault>
      </operation>
   </binding>
   <binding name="TMIRPMonitorOperations" type="tmIRPSystem:TMIRPMonitorOperations">
      <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
      <operation name="monitorTest">
         <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#monitorTest" style="document"/>
         <input>
            <soap:body use="literal"/>
         </input>
         <output>
            <soap:body use="literal"/>
         </output>
         <fault name="monitorTestFault">
            <soap:fault name="monitorTestFault" use="literal"/>
         </fault>
      </operation>
   </binding>
   <service name="TMIRPService">
      <port name="TMIRPControlOperationsPort" binding="tmIRPSystem:TMIRPControlOperations">
         <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPControl"/>
      </port>
      <port name="TMIRPMonitorOperationsPort" binding="tmIRPSystem:TMIRPMonitorOperations">
         <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.326#TMIRPMonitor"/>
      </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	**Meeting**	**TDoc**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**New version**
2010-09	SA#49	SP-100514	--	--		Presentation to SA for Information and Approval	1.0.0
2010-10	--	--	--	--		Publication	10.0.0
2010-12	SA#50	SP-100833	001	1		Correcting the definition of parameter MORT in XML schema - Align with 32.322 IS	10.1.0
09-2012	SA#57	-	-	-		Automatic upgrade from previous Release version 10.1.0	11.0.0
09-2014	SA#65	SP-140559	002	-		Update the link from Solution Set to Information Service due to the end of Release 12	12.0.0
2016-01	-	-	-	-		Update to Rel-13 version (MCC)	**13.0.0**
2016-06	SA#72	SP-160407	0003	-	F	Update the link from IRP Solution Set to IRP Information Service	13.1.0

History

Document history		
V13.0.0	February 2016	Publication
V13.1.0	August 2016	Publication