Technical Specification Group Services and System Aspects
Meeting #17, Biarritz, France, 9-12 September 2002

TSGS#17(02)0458

Source:	**SA5 (Telecom Management)**
Title:	**Rel-5 TS 32.323-200 (Test Management IRP: CORBA Solution Set)**
Document for:	**Approval**
Agenda Item:	**7.5.3**

3GPP TSG-SA5 (Telecom Management) S5-026732
Meeting #30, Tampere, Finland, 19-23 August 2002

Presentation of Technical Specification to TSG SA

Presentation to:	**TSG SA Meeting #17**
Document for presentation:	**TS 32.323, Version 2.0.0** **Test Management IRP: CORBA Solution Set**
Presented for:	**Approval**

Abstract of document:
This is a Technical Specification defining the CORBA Solution Set of the Test Management IRP. Work done against the WID contained in SP-010461 (Work Item ID: OAM-NIM).

Purpose of These Specifications:
This CORBA Solution Set Specification is intended for Release 5 and is part of the Test Management IRP, which consists of the four specifications, a Requirement Specification (32.321), an Information Service Specification (32.322) and a Stage 3 CORBA Solution Set (32.323) and CMIP Solution Set (32.324) specification.

The purpose of this set of specifications is to define an IRP that provides a framework for Test Management. It does not specify specific tests.

Status of these Specifications:

Jun 2002	S_16	SP-020328	--	--	Submitted to TSG SA #16 for Information	1.0.0	

The Test Management IRP: CORBA Solution Set Specification (32.323) is complete and ready as v2.0.0 for Approval as Release 5.

Outstanding Issues:
None.

Contentious Issues:
None.

3GPP TS 32.323 V2.0.0 (2002-09)

Technical Specification

3rd Generation Partnership Project;
Technical Specification Group Services and System Aspects;
Telecommunication Management;
Test Management Integration Reference Point (IRP);
CORBA solution set
(Release 5)

Keywords
Test Management, IRP, test object, MORT

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

Contents

Foreword

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

The present document is part of the 32.32x-series 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: "Requirements";

32.322: "Information service";

32.323: "CORBA solution set";

32.324: "CMIP solution set".

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.

1 Scope

The present document specifies the CORBA Solution Set (SS) for the IRP whose semantics is specified in Test Management IRP: Information Service (IS) (3GPP TS 32.322 [6]).

Clause 1 to 3 provides background information. Clause 4 provides key architectural features supporting the SS. Clause 5 defines the mapping of operations, notification, parameters and attributes defined in IS to their SS equivalents. Clause 6 describes the notification interface containing the push method. Annex A contains the IDL specification.

This Solution Set specification is related to 3GPP TS 32.322 (V5.0.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] Object Management Group 98 (November 1998): "*Notification Service: Joint Revised Submission OMG TC Document telecom/98-11-01*".

[2] OMG CORBA Services (November 1996): "Common Object Services Specification" (clause 4 contains the Event Service specification).

[3] 3GPP TS 32.300: "Name Convention for Managed Objects".

[4] 3GPP TS 32.302: "Telecommunication management; Configuration Management; Notification Integration Reference Point; Information Service version 1".

[5] 3GPP TS 32.303: "Telecommunication management; Configuration Management; Notification Integration Reference Point; CORBA solution set version 1:1".

[6] 3GPP TS 32.322-2: "Telecommunication management; Test management Integration Reference Point (IRP); Information service".

[7] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management".

[8] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP) management; Requirements".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions defined in 3GPP TS 32.322 [6] apply. There are no additional definitions applicable to the present document.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CORBA	Common Object Request Broker Architecture
IDL	Interface Definition Language
IRP	Integration Reference Point
IOC	Information Object Class
IS	Information Service
NE	Network Element
OMG	Object Management Group
QoS	Quality of Service
SS	Solution Set

3.3 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 [8] 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.

4 Architectural Features

The overall architectural feature of Test Management IRP is specified in 3GPP TS 32.322-2 [6]. The present document specifies features that are specific to the CORBA SS.

4.1 Notification Services

In implementations of CORBA SS, IRPAgent conveys Test Management information to IRPManager via OMG Notification Service (OMG Notification Service [1]).

OMG Event Service [2] 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 32.322-2 [6].

4.2 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 Notification CORBA SS [5] specifies that support of Push style is Mandatory (M) and that support of Pull style is Optional (O).

4.3 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.

5 Mapping

5.1 Operation and Notification mapping

Test Management IRP: IS 3GPP TS 32.322-2 [6] defines semantics of operation and notification visible across the Test Management IRP. Table 1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table 1: Mapping from IS Operations and Notification to SS equivalents

IS Operations/ notification 3G TS 32.322 [6]	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 clause 6.1)	M
NOTE: This operation is of ManagedGenericIRP IOC specified in [7]. The TestManagementIRP IOC of [6] inherits from it.		

5.2 Operation parameter mapping

The Test Management IRP: IS [6] defines semantics of parameters carried in operations across the Test Management IRP. The following tables indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table 2: Mapping from IS `initiateTests` parameters to SS equivalents

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

Table 3: 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 4: Mapping from IS `monitorTest` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
toBeMonitoredTO	TestManagementIRPConstDefs::ToBeMonitoredTO to_be_monitored_TO	M
attributeList	TestManagementIRPConstDefs::TOAttributeList tO_attribute_list	M
error	ManagedGenericIRPConstDefs::Signal Exceptions: MonitorTest, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::ParameterNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

5.3 Notification parameter mapping

The Test Management IRP: IS [6] 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 [1]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [1], is:

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

The following table lists all OMG Structured Event attributes in the second column. The first column identifies the Test Management: IS [6] defined notification parameters.

Table 5: Mapping for `notifyTestResult`

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 3.3. 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 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 [5].
notificationId	One NV pair of filterable_ body_fields	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 [5].
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 [5].
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 [5].
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.
testInvocationId	One NV pair of filterable_ body_fields	M	Name of NV pair is the TEST_INVOCATION_ID of module TestManagementIRPConstDefs. Value of NV pair is a string.
testActualStartTime	One NV pair of filterable_ body_fields	O	Name of NV pair is the TEST_ACTUAL_START_TIME of module TestManagementIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS [5].
testActualStopTime	One NV pair of filterable_ body_fields	O	Name of NV pair is the TEST_ACTUAL_STOP_TIME of module TestManagementIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS [5].
testOutcome	One NV pair of filterable_ body_fields	O	Name of NV pair is the TEST_OUTCOME of module TestManagementIRPConstDefs. Value of NV pair is the enum TestOutcomeType of TestManagementIRPConstDefs.
mORT	One NV pair of filterable_ body_fields	O	Name of NV pair is the MORT of module TestManagementIRPConstDefs. Value of NV pair is a string.

IS Parameters	OMG CORBA Structured Event attribute	Qualifier	Comment
proposedRepairActions	One NV pair of filterable_body_fields	O	Name of NV pair is the PROPOSED_REPAIR_ACTIONS of module TestManagementIRPConstDefs. Value of NV pair is a string.
additionalInformation	Two NV pair of filterable_ body_fields	O	Name of one NV pair is the TEST_ADDITIONAL_INFORMATION of module TestManagementIRPConstDefs. Value of NV pair is a string.
fileReference	One NV pair of filterable_ body_fields	O	Name of NV pair is FILE_REFERENCE of module TestManagementIRPConstDefs. Value of NV pair is a string.
fileExpiryDate	One NV pair of filterable_ body_fields	O	Name of NV pair is the FILE_EXPIRY_DATE of module TestManagementIRPConstDefs. Value of NV pair is IRPTime. See corresponding table in Notification IRP: CORBA SS [5].
There is no corresponding IS attribute.	remaining_body		

6 `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.

6.1 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 [1]). 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.

Annex A (normative): IDL specifications

A.1 IDL specification (file name "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).
   */

   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";
```

```
   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 vSTOC;          //MORT class
       NVPairs vSTONVPair;    //MORT 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.
      // Else, the test initiation fails and failureReason contains
      //     the failure reason and testInvocationId contains garbage.
      string failureReason;
      string testInvocationId;
   };
   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
```

A.2 IDL specification (file name "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
```

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Jun 2002	S_16	SP-020328	--	--	Submitted to TSG SA #16 for Information	1.0.0	
Sep 2002	S_17	SP-020458	--	--	Submitted to TSG SA #17 for Approval	2.0.0	