

INTERNATIONAL
STANDARD

ISO/IEC
10164-12

First edition
1994-11-01

Information technology — Open Systems Interconnection — Systems Management: Test Management Function

*Technologies de l'information — Interconnexion de systèmes ouverts
(OSI) — Gestion-systèmes: Fonction de gestion de tests*



Reference number
ISO/IEC 10164-12:1994(E)



Documents

Document Sharing Hub

Contents

	<i>Page</i>
1 Scope	1
2 Normative references	1
3 Definitions.....	3
4 Abbreviations	5
5 Conventions.....	5
6 Requirements.....	6
7 Model for the test function	6
8 Generic definitions	14
9 Service definitions.....	20
10 Systems management functional units	26
11 Protocol	27
12 Relationships with other functions.....	32
13 Conformance	32
Annex A – Definition of management information.....	34
Annex B – Examples	46

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 10164-12 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 21, *Open Systems Interconnection, data management and open distributed processing*, in collaboration with CCITT. The identical text is published as ITU-T Recommendation X.745.

ISO/IEC 10164 consists of the following parts, under the general title *Information technology — Open Systems Interconnection — Systems Management*:

- *Part 1: Object management function*
- *Part 2: State management function*
- *Part 3: Attributes for representing relationships*
- *Part 4: Alarm reporting function*
- *Part 5: Event report management function*
- *Part 6: Log control function*
- *Part 7: Security alarm reporting function*
- *Part 8: Security audit trail function*
- *Part 9: Objects and attributes for access control*
- *Part 10: Accounting metering function*
- *Part 11: Workload monitoring function*
- *Part 12: Test management function*
- *Part 13: Summarisation function*
- *Part 14: Confidence and diagnostic test categories*
- *Part 15: Scheduling function*

Annex A forms an integral part of this part of ISO/IEC 10164. Annex B is for information only.

Introduction

ISO/IEC 10164 is a multipart standard developed according to ISO 7498 and ISO/IEC 7498-4. ISO/IEC 10164 is related to the following International Standards

- ISO/IEC 9595:1991, *Information technology – Open Systems Interconnection – Common management information service definition*;
- ISO/IEC 9596-1:1991, *Information technology – Open Systems Interconnection – Common management information protocol – Part 1: Specification*;
- ISO/IEC 10040:1992, *Information technology – Open Systems Interconnection – Systems management overview*;
- ISO/IEC 10165:1992, *Information technology – Open Systems Interconnection – Structure of management information*.

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SYSTEMS MANAGEMENT: TEST MANAGEMENT FUNCTION

1 Scope

This Recommendation | International Standard defines a Systems Management Function that may be used by an application process in a centralized or decentralized management environment to interact for the purpose of systems management, as defined by CCITT Rec. X.700 | ISO/IEC 7498-4. This Recommendation | International Standard defines a function which consists of generic definitions, services and functional units. This function is positioned in the application layer of CCITT Rec. X.200 | ISO/IEC 7498 and is defined according to the model provided by ISO 9545. The role of systems management functions is described by CCITT Rec. X.701 | ISO/IEC 10040.

This Recommendation | International Standard

- establishes user requirements for this Recommendation | International Standard;
- establishes a test model that relates the service and generic definitions provided by this function to user requirements;
- defines generic object classes, attribute types, action types, notification types, packages and parameters documented in accordance with CCITT Rec. X.722 | ISO/IEC 10165-4;
- specifies compliance requirements placed on other standards that make use of these generic definitions;
- defines the service provided by the function;
- specifies the protocol that is necessary in order to provide the service;
- defines the relationship between this service and SMI operations and notifications;
- specifies the abstract syntax necessary to identify and negotiate the functional unit in protocol;
- defines relationships with other systems management functions;
- specifies conformance requirements to be met by implementations of this Recommendation | International Standard.

This Recommendation | International Standard does not

- define the nature of any implementation intended to provide the Test management function;
- specify the manner in which management is accomplished by the user of the Test management function;
- define the nature of any interactions which result in the use of the Test management function;
- specify the services necessary for the establishment, normal and abnormal release of a management association;
- specify the details of specific tests or test categories;
- specify a framework or methodology for conformance tests.

2 Normative references

The following Recommendations | International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations | Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent editions of the Recommendations | Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunications | Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1993, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services.*
- ISO/TR 8509:1987, *Information processing systems – Open Systems Interconnection – Service conventions.*
- CCITT Recommendation X.701 (1992) | ISO/IEC 10040:1992, *Information technology – Open Systems Interconnection – Systems management overview.*
- CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1992, *Information technology – Open Systems Interconnection – Systems Management: State management function.*
- CCITT Recommendation X.732 (1992) | ISO/IEC 10164-3:1992, *Information technology – Open Systems Interconnection – Systems Management: Attributes for representing relationships.*
- CCITT Recommendation X.734 | ISO/IEC 10164-5:1992, *Information technology – Open Systems Interconnection – Systems Management: Event report management function.*
- CCITT Recommendation X.720 (1992) | ISO/IEC 10165-1:1992, *Information technology – Open Systems Interconnection – Structure of Management information: Management information model.*
- CCITT Recommendation X.721 (1992) | ISO/IEC 10165-2:1992, *Information technology – Open Systems Interconnection – Structure of management information: Definition of management information.*
- CCITT Recommendation X.722 (1992) | ISO/IEC 10165-4:1992, *Information technology – Open Systems Interconnection – Structure of management information: Guidelines for the definition of managed objects.*

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.200 (1988), *Reference Model of Open Systems Interconnection for CCITT Applications.*
ISO 7498:1984, *Information processing systems – Open Systems Interconnection – Basic Reference Model.*
- CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1).*
ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.209 (1988), *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*
ISO 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.700, *Management framework definition for Open Systems Interconnection (OSI) for CCITT applications.*
ISO/IEC 7498-4:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework.*
- CCITT Recommendation X.710 (1991), *Common Management Information Service Definition for CCITT Applications.*
ISO/IEC 9595:1991, *Information technology – Open Systems Interconnection – Common management information service definition.*
- CCITT Recommendation X.290 (1991), *OSI Conformance Testing Methodology and Framework for Protocol Recommendations for CCITT Applications – General Concepts.*
ISO/IEC 9646-1:1991, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts.*

2.3 Additional references

- ISO/IEC 9545:1989, *Information processing systems – Open Systems Interconnection – Application Layer structure.*

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1 Basic reference model definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.200 | ISO 7498:

- a) open system;
- b) systems management.

3.2 Management framework definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.700 | ISO/IEC 7498-4:

- a) managed object;
- b) systems management application-entity.

3.3 CMIS definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.710 | ISO/IEC 9595:

- a) attribute;
- b) Common Management Information Service Element;
- c) Common Management Information Service.

3.4 Remote Operations definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.219 | ISO/IEC 9072-1:

- a) invoker;
- b) performer.

3.5 Systems management overview definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.701 | ISO/IEC 10040:

- a) agent;
- b) agent role;
- c) dependent conformance;
- d) general conformance;
- e) generic definitions;
- f) managed object class;
- g) managed (open) system;
- h) manager;
- i) manager role;
- j) MIS-User;
- k) notification;
- l) notification type;
- m) systems management application protocol;
- n) systems management functional unit.

3.6 Management information model definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.720 | ISO/IEC 10165-1:

- a) attribute type;
- b) containment hierarchy.

3.7 Guidelines for the definition of managed objects definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.722 | ISO/IEC 10165-4:

template.

3.8 Event report management function definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.734 | ISO/IEC 10164-5:

event forwarding discriminator.

3.9 OSI conformance testing definitions

This Recommendation | International Standard makes use of the following term defined in CCITT Rec. X.290 | ISO/IEC 9646-1:

system conformance statement.

3.10 Additional definitions

For the purposes of this Recommendation | International Standard, the following definitions apply:

3.10.1 associated objects (AOs): Managed objects, distinct from MORT(s), TO(s) and managed objects with TARR functionality, that are involved in a test.

3.10.2 controlled test: A test for which one or more test objects having the same value for the Test invocation identifier are created for the purpose of monitoring and control. Test results are provided by notifications emitted by the test objects or are made available as attribute values of the test objects.

3.10.3 intrusive test: A statement made with respect to a test invocation if service user disruption will or may occur as a result of the test.

3.10.4 managed object referring to test (MORT) (plural MORTs, one or more MORT(s)): A managed object which is used to refer to the functionalities that are being tested.

3.10.5 non-intrusive test: A statement made with respect to a test invocation if no service user disruption will occur as a result of the test.

3.10.6 solicited reporting: The reporting of test results by use of TO attributes.

3.10.7 test action request receiver (TARR): A term used to identify the ability of a managed object to act upon a test request. In addition to those required as a TARR, the managed object may also have other attributes, operations and notifications.

A managed object with TARR functionality may create (and subsequently delete) an instance of a particular TO class (as the result of receiving a controlled test request), representing a specific test invocation.

3.10.8 test category: One or more tests which share a common purpose and similar characteristics. A test category documents the additional behaviour and templates necessary to realize a test.

3.10.9 test conductor: A manager that issues test operations.

3.10.10 test execution: That phase of a test during which the test outcome is being determined.

3.10.11 test invocation: A specific instance of test, from the time of initiation to termination.

3.10.12 test object (TO): A managed object that exists only for a controlled test invocation and which has attributes, operations and notifications that pertain to that instance of test.

3.10.13 test object execution: That phase of testing which starts with a transition to the testing state and ends with a transition from the testing state. A TO may exhibit one or more such executions as a result of one test request.

3.10.14 test operations: management operations concerned with tests.

3.10.15 test performer: An agent which receives test operations.

3.10.16 test request: An individual request issued by a test conductor to a test performer, in order to initiate one or more test invocations.

3.10.17 test results: Information generated during test execution and made available to an open system.

3.10.18 test session: A set of test invocations.

3.10.19 uncontrolled test: A test which is not subject to monitoring or control. Test results are provided in one or more replies to the test request.

3.10.20 unsolicited reporting: The reporting of test results by use of test result notifications emitted by a TO.

4 Abbreviations

AO	Associated Object
ASE	Application Service Element
CMIS	Common Management Information Service
Conf	Confirm
DN	Distinguished Name
Id	Identifier
Ind	Indication
MAPDU	Management Application Protocol Data Unit
MORT	Managed Object Referring to Test
PDU	Protocol Data Unit
Req	Request
RDN	Relative Distinguished Name
Rsp	Response
SMAE	Systems Management Application Entity
SMAPM	Systems Management Application Protocol Machine
TARR	Test Action Request Receiver
TO	Test Object

5 Conventions

This Recommendation | International Standard defines services following the descriptive conventions defined in ITU-T Rec. X.210 | ISO/IEC 10731. In clause 9, the definition of each service includes a table that lists the parameters of its primitives. For a given primitive, the presence of each parameter is described by one of the following values:

- M The parameter is mandatory.
- (=) The value of the parameter is equal to the value of the parameter in the column to the left.
- U The use of the parameter is a service-user option.
- The parameter is not present in the interaction described by the primitive concerned.
- C The parameter is conditional. The condition(s) are defined by the text which describes the parameter.
- P Subject to the constraints imposed on the parameter by CCITT Rec. X.701 | ISO/IEC 9595.

NOTE – The parameters which are marked “P” in service tables of this Recommendation | International Standard are mapped directly onto the corresponding parameters of the CMIS service primitive, without changing the semantics or syntax of the parameters. The remaining parameters are used to construct an MAPDU.

6 Requirements

This Recommendation | International Standard is intended to satisfy requirements for the remote control of tests and provides a framework for the specification of tests which exercise resources included in an open system. The requirement for the same test functionality may originate from differing higher level areas such as fault or performance management. For example a particular test may be used to generate information that may be of use either in the verification of correct functionality, diagnosis of a fault or in the generation of performance statistics.

A test is the operation and monitoring of open systems, or parts thereof, within an environment designed to elicit information regarding the functionality and/or the performance of the subject system(s).

Each test may involve creating the environment for the test, control and monitoring of the test operation, and reassertion of the normal environment. Control of a test includes the need to suspend, resume and terminate tests. Each test will require a unique identification so that, for example, data generated by the test can be tracked.

In some cases, there is a requirement to specify tests that may be suspended, resumed and terminated when pre-defined conditions are encountered.

Features of the systems environment which may require alteration for testing are:

- the connections to other open systems;
- the configuration of the subject systems;
- the workloads requested of the subject systems.

In some cases, there is a requirement for scheduling tests. The scheduling of tests must be considered in both a periodic and an aperiodic way. For such tests there is also a requirement to allow modification of the schedule. There is also, in some cases, a requirement for allowing the test to be performed at a time convenient to the system which is to perform the test.

A test may need to be specified such that it becomes active when a preset condition exists (i.e. a threshold is crossed) or when a specific event is detected.

Requirements exist to create more complex tests from simpler ones. For example, to provide the result of many subordinate tests in a single result, or to sequence tests to efficiently diagnose a fault in an entity with a large number of components.

It may be necessary to perform a number of individual tests which together fulfill a specific requirement. In such cases, it is necessary to correlate the results of each test in order to formulate an outcome. There may also be a need for global uniqueness.

This function is seen as being applicable to different test methodologies, for example, loopback tests which configure a resource in such a way that the data sent is then received; fault injection tests in which errors are deliberately introduced in order to verify that such errors are handled properly; or self-tests which simply provide a pass-fail indication.

NOTE – There is a requirement for the test conductor to request the priority of a given test. This Recommendation | International Standard does not meet this requirement.

7 Model for the test function

This clause provides terminology concerned with tests; it defines aspects of tests that are independent of the category of test and identifies aspects of tests that shall be defined for each category of test.

7.1 Environment

The execution of a test involves two or more application processes (MIS-Users). The simplest tests involve just two, a managing process that initiates the test, the *test conductor*, and an agent process that executes the test, the *test performer*. The test performer is requested to perform the test by the test conductor. (Annex B shows examples of tests involving more than two MIS-Users).

Figure 1 illustrates a single instance of a test invocation, showing only a subset of possible protocol exchanges and does not show managed objects. For simplicity, this and succeeding figures do not show all protocol responses, for example where a confirmed event report service is used.

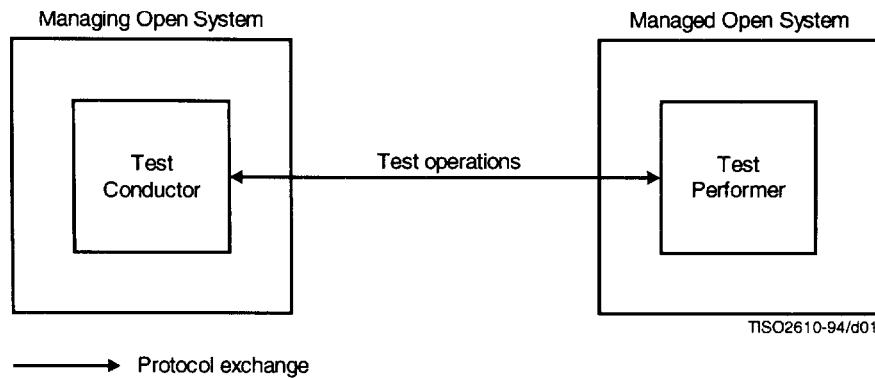


Figure 1 – Test Model

A test request is directed to a managed object, being managed by the test performer, which has functionality to receive and respond to such requests. Such functionality is called the test action request receiver functionality (TARR functionality). Managed objects which refer to functionalities that are the subjects of tests, MORTs, are identified in test requests. Each test shall involve one or more MORT. For any test, the TARR may be part of the functionality of either a MORT or another managed object. For example, the TARR functionality may be part of a managed object which exists expressly for the purpose of receiving test requests.

The execution of the test may include the use of (N)-Layer protocols, (N)-Layer Management protocols or other system specific management mechanisms, as well as Systems Management protocols. Deductions regarding the operability of the resources represented by the MORT(s) may be made if the test results are known.

A test is *uncontrolled* or *controlled*. The general test behaviour is documented by a test category.

7.1.1 Uncontrolled tests

An uncontrolled test is one which is not subject to monitoring or control. Test results are provided in one or more replies to the test request. For uncontrolled tests, the span of test invocation is from the time of the initiation request to the time at which the final response is returned. An uncontrolled test is modeled by using MORT(s) and a managed object with TARR functionality. Figure 2 depicts an example of an uncontrolled test.

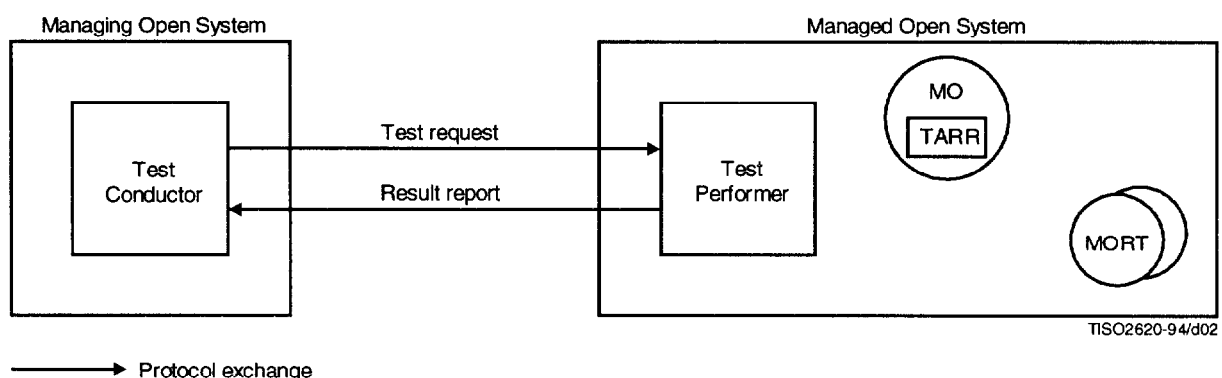


Figure 2 – Example of an uncontrolled test

7.1.2 Controlled tests

A controlled test is one for which one or more test objects, TO(s), having the same value for the test invocation identifier attribute are created for the purpose of monitoring and control. The TO(s) of a controlled test are instantiated as a consequence of a single test request. Results of the test are made available as attribute values of the TO(s) and/or are

returned via notifications issued by the TO(s). The test result notifications may be sent to any open system as event reports by instantiating an appropriate Event forwarding discriminator in the test performer as defined in CCITT Rec. X.734 | ISO/IEC 10164-5. The test results may also be stored as test result records by instantiating an appropriate log in the test performer as defined in CCITT Rec. X.735 | ISO/IEC 10164-6.

A controlled test is modeled by using TO(s), MORT(s), and a managed object with TARR functionality. Figure 3 depicts an example of a controlled test.

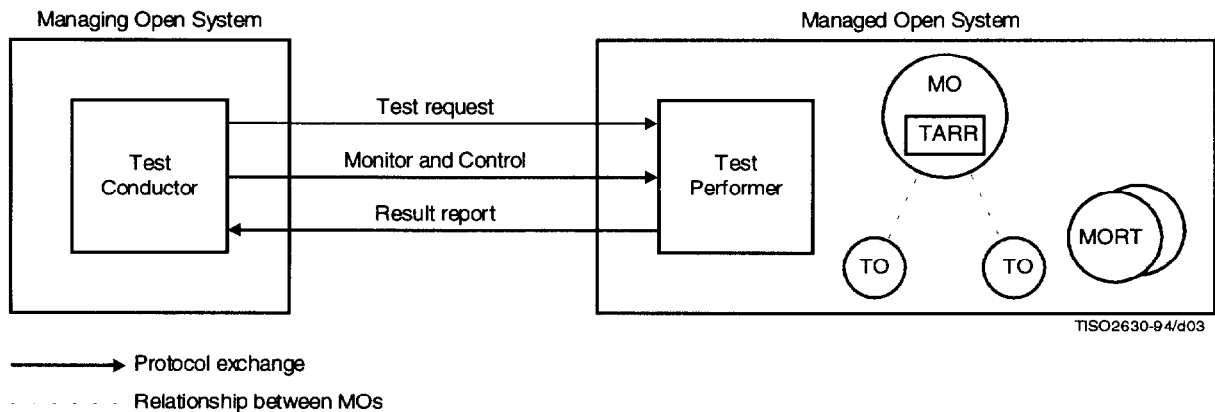


Figure 3 – Example of a controlled test

TO(s) hold information pertaining to the test. A single test request may create any number of TO(s). Relationships may exist between TO(s) and the behaviour of one TO may depend upon the behaviour of other TO(s). The name of the TO may be assigned by either the test conductor or the test performer. The test performer assigns the test invocation identifier which identifies the test invocation.

Requests to suspend, resume or terminate a controlled test are directed to the object with TARR functionality. The affected TO(s) are identified using either a test invocation identifier or a test session identifier. Requests to abort a test, if permitted, may be directed to the TO(s). The specification of the test may include the conditions under which it will be executed, suspended, resumed and terminated. These conditions may be controlled by either a scheduling mechanism, the status of a MORT or detection of a specific event.

7.2 Functional model

7.2.1 Test initiation

The request by the test conductor to the test performer to initiate the test(s) specifies the test input information and may include the following:

- the identification of MORT(s);
- test category specific input information;
- a timeout period;
- the identity of one or more associated objects;
- a test session identifier.

For controlled tests, a test conductor may also:

- indicate whether one or more tests are being requested;
- supply the classes and, optionally, names of required TO(s);
- include information about initial attribute values for the TO(s).

For uncontrolled tests, a response to a successful test request will contain test results. For controlled tests, a successful response will contain the distinguished names of the TO(s), their test invocation identifier(s), and optionally, initial attribute values. A failure response will contain a failure indication and information pertaining to the failure.

The TOs created as a result of a controlled test request may be independent tests or part of one related test. The Controlled test request type parameter is used to indicate whether the TOs are independent or related. An independent test request initiates multiple (one or more) tests each comprised of a single TO. A related test request initiates a single test comprised of more than one related TO. A single controlled test request may initiate one related test, or one or more independent tests. A controlled test with only one TO is always an independent test.

7.2.2 Test scheduling

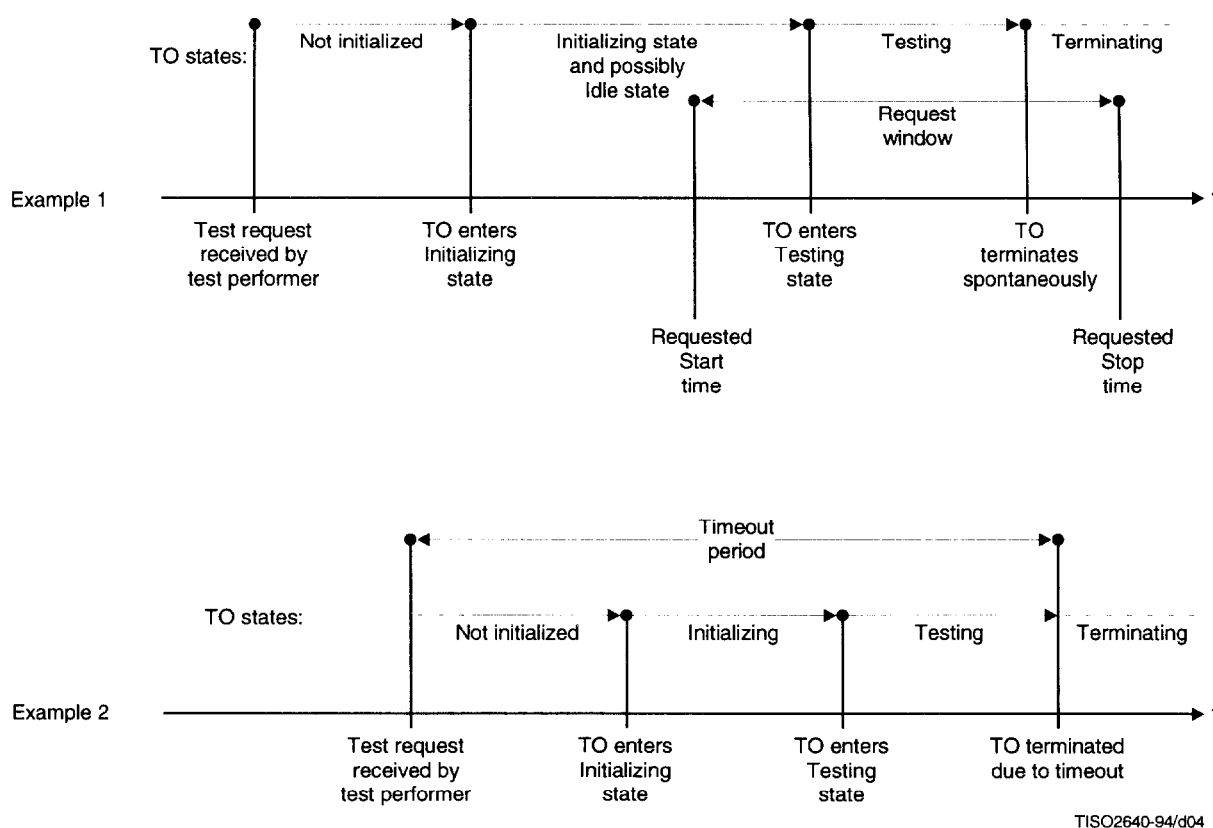
The test conductor may provide a time window in which it would like a TO to execute. If the test performer can schedule TO execution within this time window it will return a successful confirmation and may provide an actual (or expected) start and stop time for the TO, as measured or predicted by the test performer. The test performer may report any changes in the actual start and stop times. If the actual stop time becomes greater than the requested stop time or the actual start time is prior to the requested start time the TO shall cease execution (if executing) and emit a scheduling conflict notification.

The test conductor may provide a requested time window for the TO to be in the Testing state. The start time is the earliest time at which the test conductor wishes a TO to enter the Testing state and the stop time is the latest time at which the test conductor wishes a TO to leave the Testing state. This information is in the Requested window package.

The test conductor may provide a time for the TO to enter the Initializing state. This time can be specified as an absolute time or relative to the time the TO will enter the Testing state. This information is in the Initializing time package.

The test performer may provide the actual start and stop time for a TO. This information can not be directly changed by the test conductor. The actual start time is the time at which the test performer expects a TO to start execution or the time at which a TO has started execution. The actual stop time is the time at which the test performer expects a TO to end execution or the time when the TO ended execution. This information is defined in an Actual test time package.

Figure 4 shows examples of the scheduling model. Example 1 in Figure 4 illustrates the concept of the Request window. Example 2 of Figure 4 illustrates the use of a Timeout period for a test.



TISO2640-94/d04

Figure 4 – Test scheduling examples

7.2.3 Reporting of test results

For uncontrolled tests, the results of the test are reported in one or more confirmations to the test request. The final confirmation indicates that the test was completed and shall contain the test outcome parameter. The test outcome parameter shall not be present in confirmations other than the final one. This parameter may take one of the following values: pass, fail, inconclusive, timed-out or premature termination. The interpretation of this parameter is contingent on the type of test that was requested. In addition, confirmations may contain information pertinent to the type of test. If the test outcome indicates fail, the confirmation primitive may contain parameters indicating the nature of the problem and proposed repair actions.

For controlled tests, the results may be requested of the TO (solicited) by the test conductor, or results may be emitted as notifications (unsolicited) from the TO.

In the case of unsolicited reporting the results are provided by one or more notifications from a TO. A result notification shall contain the test invocation identifier of the test. The test session identifier shall also be present if present in the test request. The TO shall indicate that it is sending no more reports for an execution of a test by including the test outcome parameter in a report. Result reports may contain any other information in accord with the specification of the notifications for the TO.

The test session id may be used for correlation of test results. The algorithm by which correlation is accomplished is outside the scope of this Recommendation | International Standard.

A TO may be defined to hold test results as attribute values instead of or in addition to emitting notifications. In this case, the test conductor may directly retrieve this information by reading the relevant attributes.

Irrespective of whether the test results are solicited or unsolicited, test outcome information may only be reliable on completion of a test execution.

7.2.4 Test suspension and resumption

Only a controlled test may be suspended or resumed by a test suspend/resume request directed to the object which received the test request. Successfully suspending or resuming a test implies that all the TO(s) of the test are suspended/resumed. Tests are identified in suspend/resume requests using either a test invocation id or a test session id.

If a test session id is provided in the suspend/resume request, test invocations identified by the session shall be suspended/resumed in a best effort fashion.

If a test invocation id is provided in the suspend/resume request, all TO(s) identified by the test shall be suspended/resumed, otherwise an error shall be returned.

In the response to the suspend/resume request, the list of all test invocations that have been suspended/resumed as a result of this request shall be returned.

When a test is suspended all the TO(s) for the test are set to the suspended state. The TO definitions determine if the normal operating state of the MORT(s) is restored during suspension. When a test is resumed the TO definition determines at what point in the test lifecycle the test will be resumed. The TO test states are returned in the confirmation of the resumption request.

7.2.5 Test termination

A test may terminate spontaneously or by request. Both uncontrolled and controlled tests may terminate spontaneously, either upon the completion of the test or upon abnormal conditions (including scheduling conflicts for controlled tests). Only controlled tests may be terminated by a test terminate request or aborted by deleting all TO(s) related by the same test invocation identifier.

Spontaneous termination occurs upon the fulfillment of predefined criteria or a fault situation. These predefined criteria and some specific fault situations shall be specified by the test category or TO class. For uncontrolled tests, the final test results or test failure response shall be returned.

A controlled test may be terminated by a test terminate request directed to the object which received the test request. Successfully terminating a test implies that all the TO(s) of the test are terminated. Tests are identified in termination requests using either a test invocation id or a test session id.

If a test session id is provided in the termination request, test invocations identified by the session shall be terminated in a best effort fashion.

If a test invocation id is provided in the termination request, all TO(s) identified by the test shall be terminated otherwise an error shall be returned.

In the response to the termination request, the list of all test invocations that have been terminated as a result of this request shall be returned.

When a test is terminated the TO(s) of the test will execute a termination sequence which may include issuing test result reports and performing any necessary cleanup including ending the test activity of MORT(s) and Associated object(s). If a result report is issued and the test outcome has not been concluded then the test outcome shall indicate a value of premature termination. The temporal order in which the termination sequence(s) are carried out is system specific and not defined by the test.

A controlled test may be aborted by deleting all the TO(s) with that test invocation identifier. Test objects accepting deletion requests shall not emit any further test result reports.

7.3 Information

7.3.1 Test invocation identifier

All controlled tests, whether independent or related, shall have a unique identifier called a test invocation id. The test invocation ids are returned in the response to the test request. All TOs belonging to the same test shall have the same test invocation id. For a related test request, the same unique test invocation id is assigned to all TOs in the related test. For an independent test request, each TO represents a single test. In this case, the test invocation id may be the DN of the TO.

The Test request controlled response returns the test invocation id for each test initiated by the request. For a related test request, only one test invocation id is returned. Also returned is the identifiers of the instantiated TOs followed optionally with their attribute list. For an independent test request, a test invocation id is returned for each instantiated TO, optionally followed by the TO Name and/or its attribute list.

The test invocation id attribute is mandatory in the TO object class.

7.3.2 Test session identifier

A test session is a set of test invocations. A test session identifier identifies a test session. If used, the test session identifier shall be assigned by the test conductor and provided in the test request. The test conductor may use test session identifiers at its discretion. If present in the test request, the same test session identifier shall also be present in any TO(s) instantiated and any result notifications emitted by the TO(s) for the test.

7.3.3 Test states

TO(s) are required to support a subset of the state attributes defined in CCITT Rec. X.722 | ISO/IEC 10164-2 in order to support queries about the current state of a test.

A TO may exhibit seven distinct states:

- a) **Not initialized** – The TO is enabled but waiting to transit to the Initializing state. This transition may be controlled by the Initializing time attribute of the Initializing time package.
- b) **Idle** – The TO has not yet entered the testing state as the test start criterion has not yet been met. The transition to the testing state may be controlled by the Actual start time attribute of the Actual test time package.
- c) **Initializing** – The test environment is being prepared for testing. When the test environment has successfully been prepared, the TO proceeds to either the Idle state or the testing state depending on whether the test start criteria have been met. A test may be in this state for a significant length of time if, for example, it is required to wait for managed objects to be put into a specific administrative or operational state.
- d) **Suspended** – A TO may be suspended by a Suspend request. In this state it is not active but the TO(s) attributes are visible, i.e. read attributes may be accessed and write attributes may be modified. For example, while a test is suspended result attributes may be accessed or scheduling parameters may be changed. The test may be resumed, as determined by the TO behaviour, from the point at which it was suspended by a Resume request.
- e) **Testing** – The testing state reflects that phase of the TO during which the testing algorithms and measurements are taking place.
- f) **Terminating** – The test environment is being taken down. This may include the activities that are necessary to restore test resources or MORT(s) to their pre-test condition. A test environment may take an appreciable amount of time to be taken down.

- g) **Disabled** – The TO is disabled when it becomes totally inoperable due to failure conditions. This may be temporary or may lead to abnormal test termination.

Some classes of TO(s) exhibit a subset of these states, however all TOs shall at least support the testing state. For example, a test which requires no particular testing environment might not need any set up time and may not exhibit the initializing state nor the terminating state. Likewise, a test which does not support scheduling would not exhibit the idle state. Table 1 specifies the mapping between the states defined above and the attributes defined in CCITT Rec. X.722 | ISO/IEC 10164-2. The Control status attribute is necessary only if the TO supports the suspended state. The availability status attribute is necessary only if scheduling is supported.

Table 1 – Test state mapping

Test state	Operational state	Procedural status	Control status	Availability Status
Not initialized	Enabled	Not initialized	Not suspended	Off-duty
Idle	Enabled	(empty) or reporting	Not suspended	Off-duty
Initializing	Enabled	Initializing	Not suspended	(empty) or Off-duty
Testing	Enabled	(empty) or reporting	Not suspended	(empty)
Terminating	Enabled	Terminating (may also be reporting)	Not suspended	(empty)
Suspended	Enabled	(any value)	Suspended	(any value)
Disabled	Disabled	(any value)	(any value)	(any value)

Figure 5 shows an example of a state transition diagram for a TO that supports scheduling. Only a single TO execution is shown.

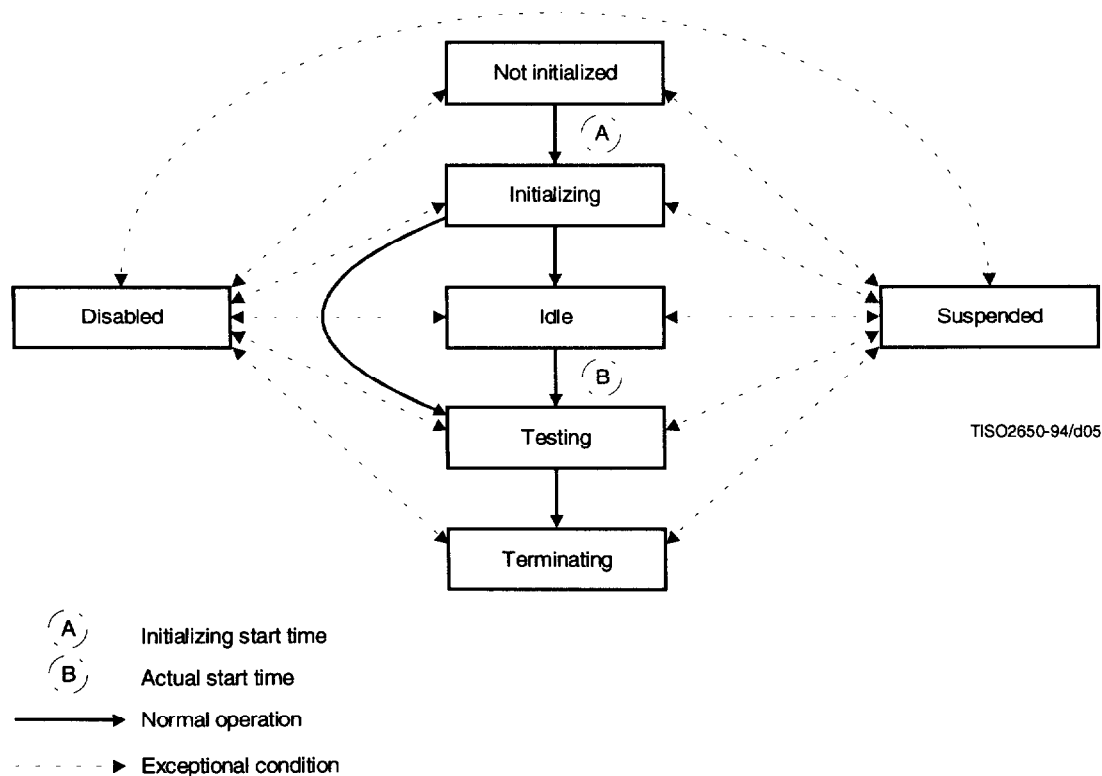


Figure 5 – Test state transition diagram

7.3.4 Test steps

A TO may have a test step package which consists of a test step attribute and a possible qualifier for a step. The test step attribute can reflect steps within each TO test state. These attributes are read only. The test steps, when used, will be defined by the TO class. The qualifier provides textual information about the test steps such as enabling conditions ('awaiting external event' or 'awaiting management input'), a procedural statement ('stop on error', 'loop on error' or 'loop at end') or status information ('fault detected'). This allows a test conductor to monitor the progress of the test and any special conditions.

7.4 Specification of test information

This form documents the management information which, taken together, specify a test. Controlled and uncontrolled tests may be in the same test category but are documented separately.

Items which do not apply to the test category being documented may be omitted.

7.4.1 Test category name

The name of this test category

7.4.2 Test category purpose

A description of the test and possible reasons for invoking it.

7.4.3 MORT requirements

A description of the MORTs to which this test is applicable. (This may include a list of managed objects).

7.4.4 Associated object requirements

A description of the Associated objects to which this test is applicable. (This may include a list of managed objects).

7.4.5 TARR requirements

Requirements upon the managed object with TARR functionality.

Subclauses 7.4.5.1 through 7.4.5.5 may be iterated for each test request service that is applicable to this test category.

7.4.5.1 Test request service type

The name of the test request action template. (Indicate the Controlled/Uncontrolled package to be used).

7.4.5.2 Specific errors

List each specific error parameter template which specifies an error response for this test and service.

7.4.5.3 Test category information parameter

The names of the parameter templates which specify the possible values for this parameter.

7.4.5.4 Additional information parameter

The names of the parameter templates which specify the possible values for this parameter (uncontrolled only).

7.4.5.5 Test control

List each of the possible control services, directed to the TARR managed object, applicable to the category (controlled only).

7.4.5.5.1 Test control specific errors

List each specific error parameter template which specifies an error response for this service.

7.4.6 TO requirements

List each TO class involved in the test (controlled only).