

INTERNATIONAL  
STANDARD

ISO/IEC/  
IEEE  
29119-2

First edition  
2013-09-01

---

---

**Software and systems engineering —  
Software testing —**

**Part 2:  
Test processes**

*Ingénierie du logiciel et des systèmes — Essais du logiciel —  
Partie 2: Processus des essais*



Reference number  
ISO/IEC/IEEE 29119-2:2013(E)

© ISO/IEC 2013  
© IEEE 2013



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2013  
© IEEE 2013

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from ISO, IEC or IEEE at the respective address below.

ISO copyright office  
Case postale 56  
CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax + 41 22 749 09 47  
E-mail [copyright@iso.org](mailto:copyright@iso.org)  
Web [www.iso.org](http://www.iso.org)

IEC Central Office  
3, rue de Varembé  
CH-1211 Geneva 20  
Switzerland  
E-mail [inmail@iec.ch](mailto:inmail@iec.ch)  
Web [www.iec.ch](http://www.iec.ch)

Institute of Electrical and Electronics Engineers, Inc.  
3 Park Avenue, New York  
NY 10016-5997, USA  
E-mail [stds.ipr@ieee.org](mailto:stds.ipr@ieee.org)  
Web [www.ieee.org](http://www.ieee.org)

Published in Switzerland

# Contents

Page

Foreword .....	v
Introduction.....	vi
1 Scope .....	1
2 Conformance .....	1
2.1 Intended Usage .....	1
2.1.1 Full Conformance .....	1
2.1.2 Tailored Conformance .....	1
3 Normative References.....	2
4 Terms and Definitions.....	2
5 Multi-Layer Test Process Model .....	10
6 Organizational Test Process .....	11
6.1 Introduction.....	11
6.2 Organizational Test Process .....	12
6.2.1 Overview.....	12
6.2.2 Purpose .....	13
6.2.3 Outcomes .....	13
6.2.4 Activities and tasks .....	13
6.2.5 Information items .....	14
7 Test Management Processes .....	15
7.1 Introduction.....	15
7.2 Test Planning Process .....	16
7.2.1 Overview.....	16
7.2.2 Purpose .....	17
7.2.3 Outcomes .....	17
7.2.4 Activities and tasks .....	17
7.2.5 Information items .....	21
7.3 Test Monitoring and Control Process .....	21
7.3.1 Overview.....	21
7.3.2 Purpose .....	22
7.3.3 Outcomes .....	22
7.3.4 Activities and tasks .....	23
7.3.5 Information Items .....	24
7.4 Test Completion Process .....	25
7.4.1 Overview.....	25
7.4.2 Purpose .....	25
7.4.3 Outcomes .....	25
7.4.4 Activities and tasks .....	26
7.4.5 Information Items .....	27
8 Dynamic Test Processes .....	27
8.1 Introduction.....	27
8.2 Test Design & Implementation Process .....	29
8.2.1 Overview.....	29
8.2.2 Purpose .....	30
8.2.3 Outcomes .....	30
8.2.4 Activities and tasks .....	31
8.2.5 Information Items .....	33
8.3 Test Environment Set-Up & Maintenance Process .....	34
8.3.1 Overview.....	34

8.3.2	Purpose.....	34
8.3.3	Outcomes .....	34
8.3.4	Activities and tasks .....	34
8.3.5	Information Items.....	35
8.4	Test Execution Process .....	36
8.4.1	Overview .....	36
8.4.2	Purpose.....	36
8.4.3	Outcomes .....	36
8.4.4	Activities and tasks .....	37
8.4.5	Information Items.....	37
8.5	Test Incident Reporting Process.....	38
8.5.1	Overview .....	38
8.5.2	Purpose.....	38
8.5.3	Outcomes .....	38
8.5.4	Activities and tasks .....	39
8.5.5	Information Items.....	39
Annex A	(informative) Partial Example Test Design Process.....	40
Annex B	(normative) ISO/IEC/IEEE 29119-2 and ISO/IEC 12207:2008 Process Alignment .....	42
B.1	Overview .....	42
B.2	ISO/IEC 12207:2008 to ISO/IEC/IEEE 29119-2 Mapping .....	42
Annex C	(informative) ISO/IEC/IEEE 29119-2 and ISO/IEC 15288:2008 process alignment .....	53
Annex D	(informative) ISO/IEC/IEEE 29119-2 and ISO/IEC 17025:2005 process alignment .....	54
Annex E	(informative) ISO/IEC/IEEE 29119-2 and ISO/IEC 25051:2006 process alignment.....	55
Annex F	(informative) ISO/IEC/IEEE 29119-2 and BS 7925-2:1998 process alignment.....	56
Annex G	(informative) ISO/IEC/IEEE 29119-2 and IEEE Std 1008-2008 process alignment .....	57
Bibliography	.....	59

## 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.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of ISO/IEC JTC 1 is to prepare International Standards. 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.

Attention is called to the possibility that implementation of this standard may require the use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISO/IEEE is not responsible for identifying essential patents or patent claims for which a license may be required, for conducting inquiries into the legal validity or scope of patents or patent claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance or a Patent Statement and Licensing Declaration Form, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from ISO or the IEEE Standards Association.

ISO/IEC/IEEE 29119-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*, in cooperation with the Software & Systems Engineering Standards Committee of the IEEE Computer Society, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

ISO/IEC 29119 consists of the following standards, under the general title *Software and systems engineering — Software testing*:

- *Part 1: Concepts and definitions*
- *Part 2: Test processes*
- *Part 3: Test documentation*
- *Part 4: Test techniques*

## Introduction

The purpose of the ISO/IEC/IEEE series of software testing standards is to define a generic process model for software testing that can be used by any organization when performing any form of software testing. It comprises test process descriptions that define the software testing processes at the organizational level, test management level and dynamic test levels. Supporting informative diagrams describing the processes are also provided. ISO/IEC/IEEE 29119 supports dynamic testing, functional and non-functional testing, manual and automated testing, and scripted and unscripted testing. The processes defined in this series of international standards can be used in conjunction with any software development lifecycle model. Each process is defined using the generic process template that is provided in ISO/IEC TR 24774:2010 Guidelines for Process Description, and covers the purpose, outcomes, activities, tasks and information items of each test process.

Testing is a key approach to risk mitigation in software development. This part of ISO/IEC/IEEE 29119 follows a risk-based approach to testing. Risk-based testing is a best-practice approach to strategizing and managing testing, as it allows testing to be prioritized and focused on the most important features and quality attributes.

The concepts and vocabulary that support this series of international standards are defined in ISO/IEC/IEEE 29119-1 Concepts and definitions. Templates and examples of test documentation that are produced during the testing process are defined in ISO/IEC/IEEE 29119-3 Test documentation. Software test design techniques that can be used during testing are defined in ISO/IEC/IEEE 29119-4 Test techniques.

This series of international standards aims to provide those responsible for software testing with the information required to manage and perform software testing in any organization.

# Software and systems engineering — Software testing —

## Part 2: Test processes

### 1 Scope

This part of ISO/IEC/IEEE 29119 specifies test processes that can be used to govern, manage and implement software testing for any organization, project or smaller testing activity. It comprises generic test process descriptions that define the software testing processes. Supporting informative diagrams describing the processes are also provided.

This part of ISO/IEC/IEEE 29119 is applicable to testing in all software development lifecycle models.

This part of ISO/IEC/IEEE 29119 is intended for, but not limited to, testers, test managers, developers and project managers, particularly those responsible for governing, managing and implementing software testing.

### 2 Conformance

#### 2.1 Intended usage

The requirements in this part of ISO/IEC/IEEE 29119 are contained in Clauses 6 to 8. This part of ISO/IEC/IEEE 29119 provides requirements for a number of test processes suitable for use during the complete software lifecycle. It is recognized that particular projects or organizations may not need to use all of the processes defined by this part of ISO/IEC/IEEE 29119. Therefore, implementation of this part of ISO/IEC/IEEE 29119 typically involves selecting a set of processes suitable for the organization or project. There are two ways that an organization can claim to conform to the provisions of this part of ISO/IEC/IEEE 29119.

The organization shall assert whether it is claiming full or tailored conformance to this part of ISO/IEC/IEEE 29119:

##### 2.1.1 Full conformance

Full conformance is achieved by demonstrating that all of the requirements (i.e. shall statements) of the full set of processes defined in this part of ISO/IEC/IEEE 29119 have been satisfied.

##### 2.1.2 Tailored conformance

When this part of ISO/IEC/IEEE 29119 is used as a basis for establishing a set of processes that do not qualify for full conformance, the subset of processes for which tailored conformance is claimed, is recorded. Tailored conformance is achieved by demonstrating that all of the requirements (i.e. shall statements) for the recorded subset of processes have been satisfied.

Where tailoring occurs, justification shall be provided (either directly or by reference), whenever a process defined in Clauses 6, 7 and 8 of this part of ISO/IEC/IEEE 29119 is not followed. All tailoring decisions shall be recorded with their rationale, including the consideration of any applicable risks. Tailoring decisions shall be agreed by the relevant stakeholders.

**EXAMPLE** Where organizations follow information item management processes in standards such as ISO 15489 (Information and documentation - Records management) or ISO 9001 (Quality management systems - Requirements) or

use similar internal organizational processes, they can decide to use those processes in place of the information item management tasks defined in this part of ISO/IEC/IEEE 29119.

### 3 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC/IEEE 29119-1, *Software and systems engineering — Software testing — Part 1: Concepts and definitions*

ISO/IEC/IEEE 29119-3, *Software and systems engineering — Software testing — Part 3: Test documentation*

ISO/IEC/IEEE 29119-4, *Software and systems engineering — Software testing — Part 4: Test techniques*<sup>1</sup>

ISO/IEC 12207:2008, *Systems and software engineering — Software life cycle processes*

Other standards useful for the implementation and interpretation of this document are listed in the Bibliography.

### 4 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC/IEEE 24765 and the following apply.

**NOTE** Use of the terminology in this part of ISO/IEC/IEEE 29119 is for ease of reference and is not mandatory for conformance with this part of ISO/IEC/IEEE 29119. The following terms and definitions are provided to assist with the understanding and readability of this part of ISO/IEC/IEEE 29119. Only terms critical to the understanding of this part of ISO/IEC/IEEE 29119 are included. This clause is not intended to provide a complete list of testing terms. The Systems and Software Engineering Vocabulary ISO/IEC/IEEE 24765 can be referenced for terms not defined in this clause. This source is available at the following web site: <http://www.computer.org/sevocab>. All terms defined in this clause are also intentionally included in ISO/IEC/IEEE 29119-1, as that international standard includes all terms that are used in ISO/IEC/IEEE 29119-1, -2, -3 and -4.

#### 4.1

##### **actual results**

set of behaviours or conditions of a test item, or set of conditions of associated data or the test environment, observed as a result of test execution

**EXAMPLE** Outputs to screen, outputs to hardware, changes to data, reports and communication messages sent.

#### 4.2

##### **completion criteria**

conditions under which the testing activities are considered complete

#### 4.3

##### **coverage item**

see test coverage item (4.33)

#### 4.4

##### **dynamic testing**

testing that requires the execution of program code

---

<sup>1</sup> To be published.



**4.5****equivalence partition**

subset of the range of values of a variable, or set of variables, within a test item or at its interfaces such that all the values in the partition can reasonably be expected to be treated similarly by the test item (i.e. they may be considered "equivalent")

**4.6****equivalence partition coverage**

proportion of identified equivalence partitions of a test item that are covered by a test set

Note 1 to entry: In many cases, the identification of equivalence partitions is subjective (especially in the sub-partitioning of "invalid" partitions), so a definitive count of the number of equivalence partitions in a test item can be impossible.

**4.7****equivalence partitioning**

test design technique in which test cases are designed to exercise equivalence partitions by using one or more representative members of each partition

**4.8****expected result**

observable predicted behaviour of the test item under specified conditions based on its specification or another source

**4.9****exploratory testing**

type of unscripted experience-based testing in which the tester spontaneously designs and executes tests based on the tester's existing relevant knowledge, prior exploration of the test item (including the results of previous tests), and heuristic "rules of thumb" regarding common software behaviours and types of failure

Note 1 to entry: Exploratory testing hunts for hidden properties (including hidden behaviours) that, while quite possibly benign by themselves, can interfere with other properties of the software under test, and so constitute a risk that the software will fail.

**4.10****feature set**

logical subset of the test item(s) that could be treated independently of other feature sets in the subsequent test design activities

Note 1 to entry: This could be the set of all features for the item (its full feature set), or a subset identified for a specific purpose (the functional feature set, etc.).

**4.11****Incident Report**

documentation of the occurrence, nature, and status of an incident

Note 1 to entry: Incident reports are also known as anomaly reports, bug reports, defect reports, error reports, issues, problem reports and trouble reports, amongst other terms.

**4.12****performance testing**

type of testing conducted to evaluate the degree to which a test item accomplishes its designated functions within given constraints of time and other resource

**4.13****Organizational Test Process**

test process for developing and managing organizational test specifications

#### 4.14

##### **Organizational Test Policy**

see Test Policy

#### 4.15

##### **Organizational Test Specification**

document that provides information about testing for an organization, i.e. information that is not project-specific

EXAMPLE The most common examples of organizational test specifications are the Organizational Test Policy and Organizational Test Strategy.

#### 4.16

##### **Organizational Test Strategy**

document that expresses the generic requirements for the testing to be performed on all the projects run within the organization, providing detail on how the testing is to be performed

Note 1 to entry: The Organizational Test Strategy is aligned with the Organizational Test Policy.

Note 2 to entry: An organization can have more than one Organizational Test Strategy to cover markedly different project contexts.

#### 4.17

##### **product risk**

risk that a product can be defective in some specific aspect of its function, quality, or structure

#### 4.18

##### **project risk**

risk related to the management of a project

EXAMPLE Lack of staffing, strict deadlines, changing requirements.

#### 4.19

##### **regression testing**

testing following modifications to a test item or to its operational environment, to identify whether regression failures occur

Note 1 to entry: The sufficiency of a set of regression test cases depends on the item under test and on the modifications to that item or its operational environment.

#### 4.20

##### **retesting**

re-execution of test cases that previously returned a "fail" result, to evaluate the effectiveness of intervening corrective actions

Note 1 to entry: Retesting is often combined with regression testing.

Note 2 to entry: Retesting is also known as confirmation testing.

#### 4.21

##### **risk-based testing**

testing in which the management, selection, prioritisation, and use of testing activities and resources is consciously based on corresponding types and levels of analyzed risk

#### 4.22

##### **security testing**

type of testing conducted to evaluate the degree to which a test item, and associated data and information, are protected so that unauthorized persons or systems cannot use, read, or modify them, and authorized persons or systems are not denied access to them

**4.23****scripted testing**

testing performed based on a documented test script

Note 1 to entry: This term normally applies to manually executed testing, rather than the execution of an automated script.

**4.24****static testing**

testing in which a test item is examined against a set of quality or other criteria without code being executed

EXAMPLE      Reviews or static analysis.

**4.25****stress testing**

type of performance efficiency testing conducted to evaluate a test item's behaviour under conditions of loading above anticipated or specified capacity requirements, or of resource availability below minimum specified requirements

**4.26****test basis**

body of knowledge used as the basis for the design of tests and test cases

Note 1 to entry: The test basis can take the form of documentation, such as a requirements specification, design specification, or module specification, but can also be an undocumented understanding of the required behaviour.

Note 2 to entry: For specification-based testing the test basis is used to derive both test inputs and expected results, whereas for structure-based testing, the test basis is used solely for deriving expected results.

**4.27****test case**

set of test case preconditions, inputs (including actions, where applicable), and expected results, developed to drive the execution of a test item to meet test objectives, including correct implementation, error identification, checking quality, and other valued information

Note 1 to entry: A test case is the lowest level of test input (i.e. test cases are not made up of other test cases).

**4.28****Test Case Specification**

documentation of a set of one or more test cases

**4.29****Test Completion Process**

Test Management Process for ensuring that useful test assets are made available for later use, test environments are left in a satisfactory condition, and the results of testing are recorded and communicated to relevant stakeholders

**4.30****Test Completion Report**

report that provides a summary of the testing that was performed

**4.31****test condition**

testable aspect of a component or system, such as a function, transaction, feature, quality attribute, or structural element identified as a basis for testing

Note 1 to entry: Test conditions can be used to derive coverage items, or can themselves constitute coverage items.

**4.32**

**test coverage**

degree, expressed as a percentage, to which specified test coverage items have been exercised by a test case or test cases

**4.33**

**test coverage item**

attribute or combination of attributes that is derived from one or more test conditions by using a test design technique that enables the measurement of the thoroughness of the test execution

**4.34**

**test data**

data created or selected to satisfy the input requirements for executing one or more test cases, which can be defined in the Test Plan, test case or test procedure

Note 1 to entry: Test data can be stored within the product under test (e.g., in arrays, flat files, or a database), or can be available from or supplied by external sources, such as other systems, other system components, hardware devices, or human operators.

**4.35**

**Test Data Readiness Report**

document describing the status of each test data requirement

**4.36**

**Test Design and Implementation Process**

test process for deriving and specifying test cases and test procedures

**4.37**

**Test Design Specification**

document specifying the features to be tested and their corresponding test conditions

**4.38**

**test design technique**

activities, concepts, processes, and patterns used to construct a test model that is used to identify test conditions for a test item, derive corresponding test coverage items, and subsequently derive or select test cases

**4.39**

**test environment**

facilities, hardware, software, firmware, procedures, and documentation intended for or used to perform testing of software

**4.40**

**test environment readiness report**

document that describes the status of each environment requirement

**4.41**

**Test Environment Requirements**

description of the necessary properties of the test environment

Note 1 to entry: All or parts of the test environment requirements can reference where the information can be found, e.g. in the appropriate Organizational Test Strategy, Test Plan, and/or Test Specification.

**4.42**

**Test Environment Set-up Process**

dynamic test process for establishing and maintaining a required test environment

**4.43****test execution**

process of running a test on the test item, producing actual results

**4.44****Test Execution Log**

document that records details of the execution of one or more test procedures

**4.45****Test Execution Process**

dynamic test process for executing test procedures created in the Test Design and Implementation Process in the prepared test environment, and recording the results

**4.46****Test Incident Reporting Process**

dynamic test process for reporting to the relevant stakeholders issues requiring further action that were identified during the test execution process

**4.47****test item**

work product that is an object of testing

EXAMPLE      System, software item, requirements document, design specification, user guide.

**4.48****test level**

specific instantiation of a test sub-process

EXAMPLE      The following are common test levels that can be instantiated as test sub-processes: component testing, integration testing, system testing and acceptance testing.

Note 1 to entry: Test levels are also known as test phases.

**4.49****test management**

planning, estimating, monitoring, reporting, control and completion of test activities

**4.50****Test Management Process**

test process for containing the sub-processes that are required for the management of a test project

Note 1 to entry: See test planning process, test monitoring and control process, test completion process.

**4.51****Test Monitoring and Control Process**

test management process for ensuring that testing is performed in line with a Test Plan and with organizational test specifications

**4.52****test phase**

specific instantiation of test sub-process

Note 1 to entry: Test phases are synonymous with test levels, therefore examples of test phases are the same as for test levels (e.g. system test phase/sub-process).

**4.53****Test Plan**

detailed description of test objectives to be achieved and the means and schedule for achieving them, organised to coordinate testing activities for some test item or set of test items

Note 1 to entry: A project can have more than one test plan, for example there can be a project test plan (also known as a master test plan) that encompasses all testing activities on the project; further detail of particular test activities can be defined in one or more test sub-process plans (i.e. a system test plan or a performance test plan).

Note 2 to entry: Test plans can also be written for non-project activities, for example a maintenance test plan.

#### **4.54**

##### **Test Planning Process**

Test Management Process used to complete test planning and develop Test Plans

#### **4.55**

##### **Test Policy**

an executive-level document that describes the purpose, goals, principles and scope of testing within an organization

Note 1 to entry: The Test Policy defines what testing is performed and what it is expected to achieve but does not detail how testing is to be performed.

Note 2 to entry: The Test Policy can provide a framework for establishing, reviewing and continually improving the organisations testing.

#### **4.56**

##### **test procedure**

sequence of test cases in execution order, and any associated actions that may be required to set up the initial preconditions and any wrap up activities post execution

Note 1 to entry: Test procedures include detailed instructions for how to run a set of one or more test cases selected to be run consecutively, including set up of common preconditions, and providing input and evaluating the actual results for each included test case.

#### **4.57**

##### **Test Procedure Specification**

document specifying one or more test procedures, which are collections of test cases to be executed for a particular objective

Note 1 to entry: The test cases in a test set are listed in their required order in the test procedure

Note 2 to entry: Also known as a manual test script. A test procedure specification for an automated test run is usually called a test script.

#### **4.58**

##### **test process**

used to provide information on the quality of a software product, often comprised of a number of activities, grouped into one or more test sub-processes

#### **4.59**

##### **test result**

indication of whether or not a specific test case has passed or failed, i.e. if the actual results corresponds to the expected results or if deviations were observed

#### **4.60**

##### **test requirement**

see test condition (4.31)

#### **4.61**

##### **test script**

test procedure specification for manual or automated testing

**4.62****test set**

collection of test cases for the purpose of testing a specific test objective

Note 1 to entry: The test sets will typically reflect the feature sets, but they could contain test cases for a number of feature sets.

Note 2 to entry: Test cases for a test set could be selected based on the identified risks, test basis, retesting and/or regression testing.

**4.63****test specification**

complete documentation of the test design, test cases and test procedures for a specific test item

Note 1 to entry: A test specification can be detailed in one document, in a set of documents, or in other ways, for example in a mixture of documents and database entries.

**4.64****test specification technique**

see test design technique (4.38)

**4.65****Test Status Report**

report that provides information about the status of the testing that is being performed in a specified reporting period

**4.66****test strategy**

part of the Test Plan that describes the approach to testing for a specific test project or test sub-process or sub-processes

Note 1 to entry: The test strategy is a distinct entity from the Organizational Test Strategy.

Note 2 to entry: The test strategy usually describes some or all of the following; the test sub-processes to be implemented; the retesting and regression testing to be employed; the test design techniques and corresponding test completion criteria to be used; test data; test environment and testing tool requirements; and expectations for test deliverables.

**4.67****test sub-process**

test management and dynamic (and static) test processes used to perform a specific test level (e.g. system testing, acceptance testing) or test type (e.g. usability testing, performance testing) normally within the context of an overall test process for a test project

Note 1 to entry: A test sub-process can comprise one or more test types. Depending on the life cycle model used, test sub-processes are also typically called test phases, test levels, test stages or test tasks.

**4.68****test technique**

see test design technique (4.38)

**4.69****test type**

group of testing activities that are focused on specific quality characteristics

Note 1 to entry: A test type can be performed in a single test sub-process or can be performed across a number of test sub-processes (e.g. performance testing completed at a component test sub-process and also completed at a system test sub-process).

EXAMPLE Security testing, functional testing, usability testing, and performance testing.

#### 4.70

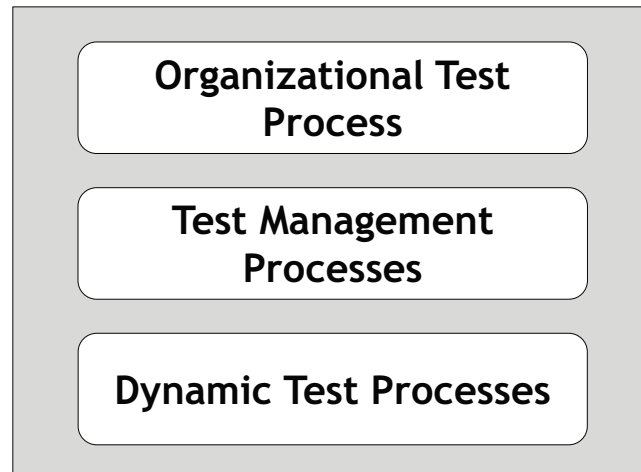
#### testing

set of activities conducted to facilitate discovery and/or evaluation of properties of one or more test items

Note 1 to entry: Testing activities include planning, preparation, execution, reporting, and management activities, insofar as they are directed towards testing.

## 5 Multi-Layer Test Process Model

This standard groups the testing activities that may be performed during the life cycle of a software system into three process groups, as shown in Figure 1. Each of the processes within those groups is described in terms of its purpose and desired outcomes and activities and tasks which need to be performed are listed.



**Figure 1 — The multi-layer testing processes**

The aim of each layer is as follows:

- a) Organizational Test Process (clause 6)
  - 1) Defining a process for the creation and maintenance of organizational test specifications, such as organizational test policies, strategies, processes, procedures and other assets.
- b) Test Management Processes (clause 7)
  - 1) Defining processes that cover the management of testing for a whole test project or any test phase (e.g. system testing) or test type (e.g. performance testing) within a test project (e.g. project test management, system test management, performance test management).
  - 2) The Test Management Processes are:
    - i) Test Planning Process (clause 7.2);
    - ii) Test Monitoring and Control Process (clause 7.3);
    - iii) Test Completion Process (clause 7.4)



## c) Dynamic Test Processes (clause 8)

- 1) Defining generic processes for performing dynamic testing. Dynamic testing may be performed at a particular phase of testing (e.g. unit, integration, system, and acceptance) or for a particular type of testing (e.g. performance testing, security testing, and functional testing) within a test project.
- 2) The Dynamic Test Processes are:
  - i) Test Design and Implementation Process (clause 8.2);
  - ii) Test Environment Set-Up and Maintenance Process (clause 8.3);
  - iii) Test Execution Process (clause 8.4); and
  - iv) Test Incident Reporting Process (clause 8.5).

NOTE In IEEE 1012, the Dynamic Test Process is referred to as "the Test Process".

The layers of the test process model comprise varying numbers of test process, as shown in Figure 2.

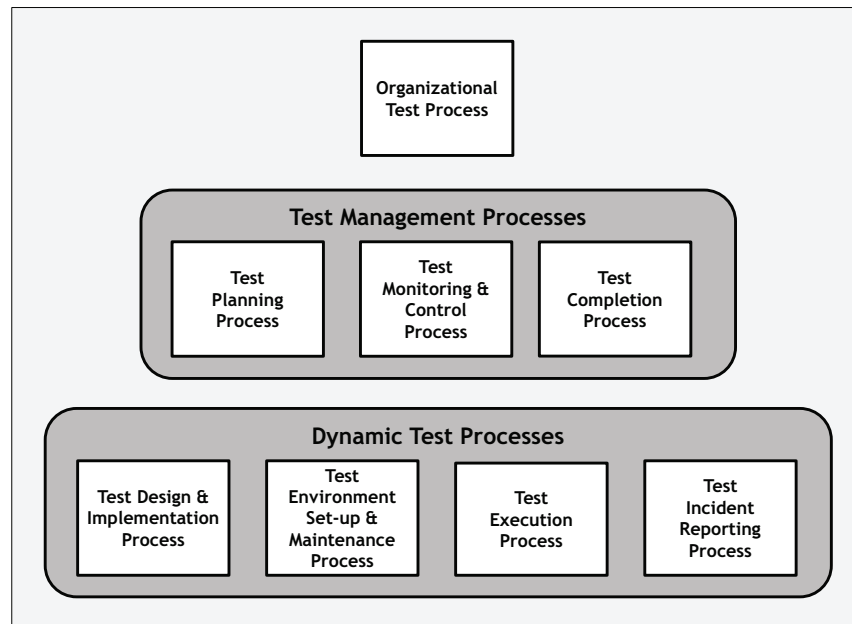


Figure 2 — The multi-layer model showing all test processes

## 6 Organizational Test Process

### 6.1 Introduction

The Organizational Test Process is used to develop and manage organizational test specifications. These specifications typically apply to testing across the whole organization (i.e. they are not project-based). The Organizational Test Policy and Organizational Test Strategy are examples of organizational test specifications. The organizational test process is generic and can be used to develop and manage other non-project specific test documents, such as a Programme Test Strategy that applies to a number of related projects.

The Organizational Test Policy is an executive-level document that describes the purpose, goals, and overall scope of testing within the organization. It also establishes organizational testing practices and provides a

framework for establishing, reviewing and continually improving the organization's Test Policy, Test Strategy and approach to project test management.

The Organizational Test Strategy is a detailed, technical document that defines how testing is carried out within the organization. It is a generic document that provides guidelines for a number of projects in the organization and is not project-specific.

In Figure 3 the organizational test process is shown in a typical situation where it has been applied to create and maintain both an organization's Test Policy and Test Strategy. As Figure 3 illustrates, the two instances of the organization-level processes communicate with each other. The Organizational Test Strategy needs to align with the Organizational Test Policy and feedback from this activity is provided back to the Test Policy for possible process improvement. Similarly, the test management processes being used on each of the projects within the organization need to align with the Organizational Test Strategy (and Policy) and feedback from the management of these projects is used to improve the organizational test process which formulates and maintains the organizational test specifications.

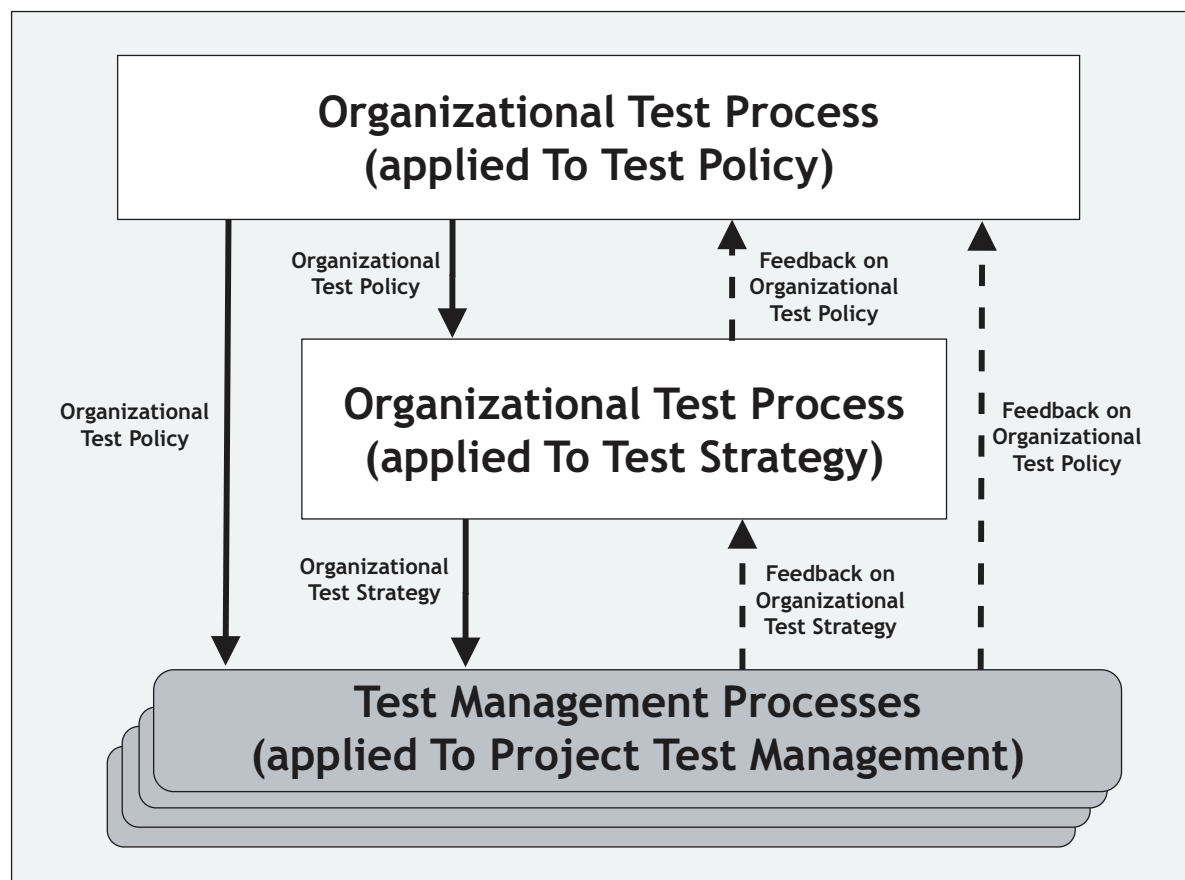


Figure 3 — Example Organizational Test Process implementation

## 6.2 Organizational Test Process

### 6.2.1 Overview

The *Organizational Test Process* comprises activities for the creation, review and maintenance of organizational test specifications. It also covers the monitoring of organizational compliance with them (see Figure 4).

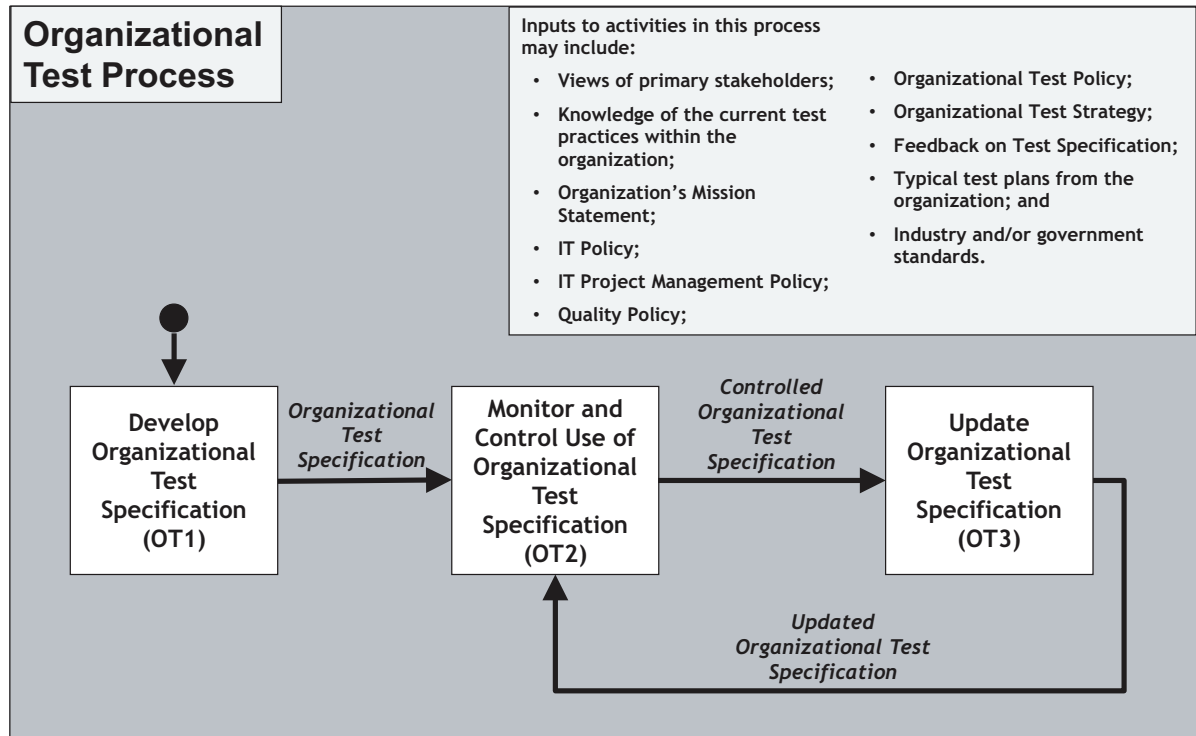


Figure 4 — Organizational Test Process

### 6.2.2 Purpose

The purpose of the *Organizational Test Process* is to develop, monitor conformance and maintain organizational test specifications, such as the Organizational Test Policy and Organizational Test Strategy.

### 6.2.3 Outcomes

As a result of the successful implementation of the *Organizational Test Process*:

- The requirements for organizational test specifications are identified;
- The organizational test specifications are developed;
- The organizational test specifications are agreed to by stakeholder(s);
- The organizational test specifications are made accessible;
- Conformance to the organizational test specifications is monitored;
- Updates to organizational test specifications are agreed to by stakeholder(s); and
- Updates to the organizational test specifications are made.

### 6.2.4 Activities and tasks

The person responsible for organizational test specifications shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Organizational Test Process*.

#### 6.2.4.1 Develop Organizational Test Specification (OT1)

This activity consists of the following tasks:

- a) Requirements for the organizational test specifications shall be identified from the current testing practices within the organization, from stakeholders and/or will be developed by other means.

NOTE This can be achieved by analyzing relevant source documents, through workshops, interviews or other suitable means.

- b) The organizational test specification requirements shall be used to create the organizational test specification.
- c) Approval on the content of the organizational test specification shall be obtained from the stakeholders.
- d) The availability of the Organization Test Specification shall be communicated to the stakeholders in the organization.

#### 6.2.4.2 Monitor and Control Use of Organizational Test Specification (OT2)

This activity consists of the following tasks:

- a) Usage of the Organizational Test Specification shall be monitored to determine whether it is being used effectively within the organization.
- b) Appropriate actions shall be taken to encourage alignment of stakeholders to the organizational test specification.

#### 6.2.4.3 Update Organizational Test Specification (OT3)

This activity consists of the following tasks:

- a) Feedback on use of the organizational test specification should be reviewed.
- b) The effectiveness of the use and management of the organizational test specification should be considered and any feedback and changes to improve its effectiveness should be determined and approved.

NOTE This can be achieved by reviewing feedback, through workshops, interviews and other suitable means.

- c) Where changes to the organizational test specification have been identified and approved, these changes shall be implemented.
- d) All changes to the organizational test specification shall be communicated throughout the organization including to all stakeholders.

#### 6.2.5 Information items

As a result of carrying out this process, the following information item shall be produced:

- a) Organizational Test Specification

EXAMPLE Organizational Test Policy, Organizational Test Strategy.

## 7 Test Management Processes

### 7.1 Introduction

There are three test management processes:

- a) test planning;
- b) test monitoring and control; and
- c) test completion.

These generic test management processes may be applied at the project level (project test management), for test management at different test phases (e.g. system test management, acceptance test management) and for managing various test types (e.g. performance test management, usability test management).

When applied at the project test management level, these test management processes are used to manage the testing for the whole project, based on a project test plan. For many projects, each of the individual test phases and types will also require the test management processes to be applied to their management separately; these will typically be based on separate test plans, such as the system test plan, reliability test plan and acceptance test plan.

Figure 5 illustrates the relationships between the three test management processes, and how they interact with the organizational test process, other applications of the test management processes and the dynamic test processes.

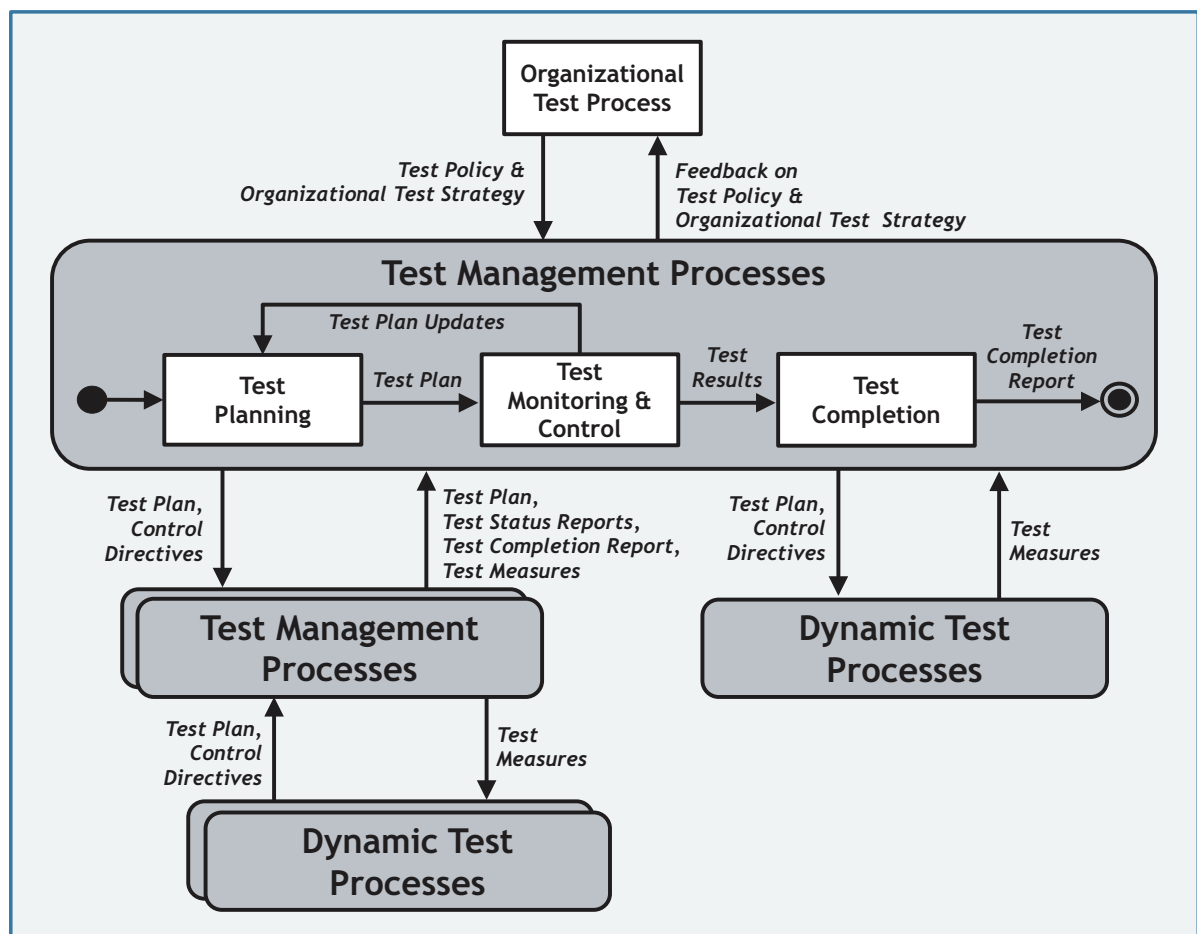


Figure 5 — Example test management process relationships

Test management processes need to align with the outputs of the organizational test process, such as the Organizational Test Policy and Organizational Test Strategy. Based on the practical implementation of these outputs, the test management processes may produce feedback on the organizational test process.

## 7.2 Test Planning Process

### 7.2.1 Overview

The *Test Planning Process* is used to develop the Test Plan. Depending on where in the project this process is implemented this may be a Project Test Plan or a test plan for a specific phase, such as a System Test Plan, or a test plan for a specific type of testing, such as a Performance Test Plan.

To create a test plan, the activities shown in Figure 6 shall be performed. As content for the test plan becomes available through performing the defined activities, a draft test plan will gradually be elaborated until the complete test plan is recorded. Due to the iterative nature of the process, a number of the activities shown in Figure 6 may need to be re-visited before the complete test plan can be made available. Most typically the activities TP3, TP4, TP5 and TP6 will need to be performed iteratively in order to achieve an acceptable test plan.

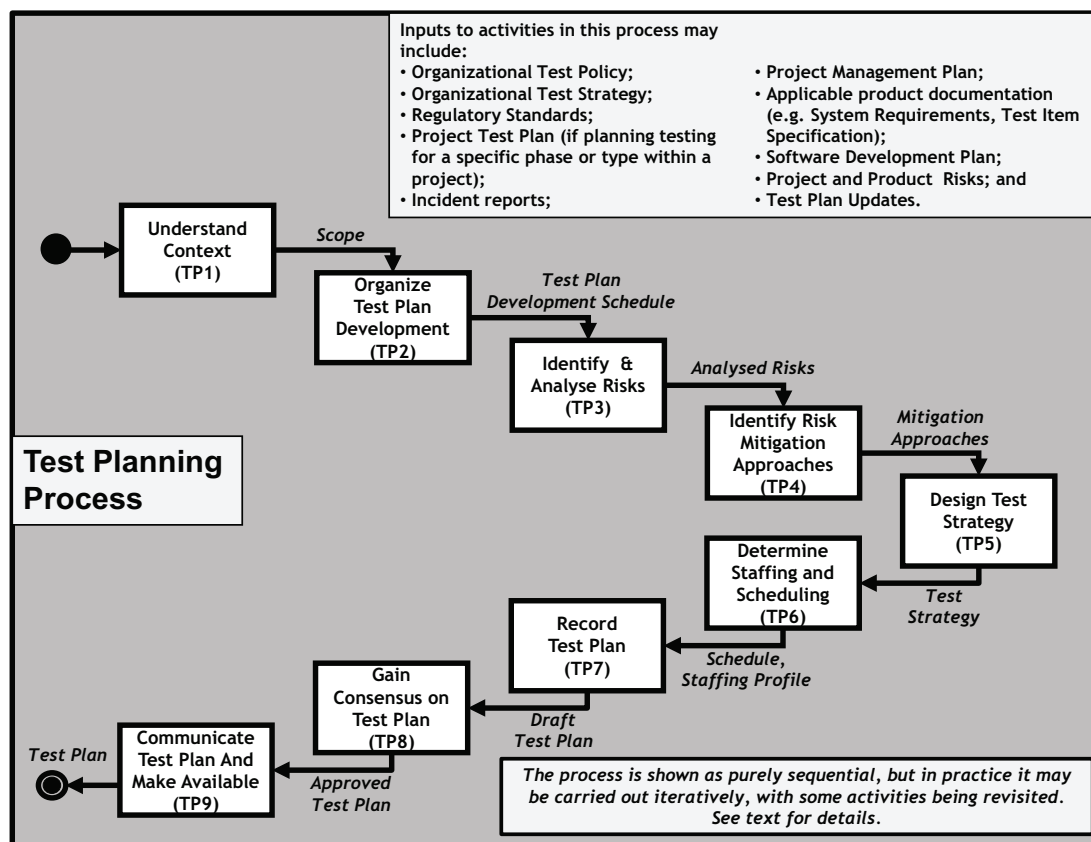


Figure 6 — Test Planning Process

During the course of testing, the test plan may need to be modified in response to the results of implementing the plan and new information becoming available. Depending on the scale and nature of the changes, a variety of activities in figure 6 will need to be re-visited to maintain the test plan.

For example, if, after the test plan is initially made available, it is realised that new risks threaten either the project or the deliverable product, or that the threat from existing risks has changed, then the process should be re-entered at *Identify & Analyse Risks* (TP3).

Alternatively, if it is believed to be necessary to change the test strategy for reasons other than risk (e.g. a different test environment is to be used) then the process should be re-entered at *Design Test Strategy* (TP5).

Or, if it is believed to be necessary to change either the staffing or scheduling of the testing for reasons other than risk (e.g. the availability of test items from development has changed) then the process should be re-entered at *Determine Staffing and Scheduling* (TP6).

### 7.2.2 Purpose

The purpose of the *Test Planning Process* is to develop, agree, record and communicate to relevant stakeholders the scope and approach that will be taken to testing, enabling early identification of resources, environments and other requirements of testing.

### 7.2.3 Outcomes

As a result of the successful implementation of the *Test Planning Process*:

- a) The scope of work of the test project is analysed and understood;
- b) The stakeholders who will participate in the test planning are identified and informed;
- c) Risks that can be treated by testing are identified, analysed and classified with an agreed level of risk exposure;
- d) Test strategy, test environment, test tool and test data needs are identified;

EXAMPLE 1 Tools, special equipment, test environment, office space.

- e) Staffing and training needs are identified;
- f) Each activity is scheduled;
- g) Estimates are calculated and evidence to justify the estimates is recorded;

EXAMPLE 2 Cost, staff, and timeline estimates.

- h) The Test Plan is agreed to and distributed to all stakeholders.

### 7.2.4 Activities and tasks

The person(s) responsible for test planning shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Planning Process*.

#### 7.2.4.1 Understand Context (TP1)

This activity consists of the following tasks:

- a) Understanding of the context and the software testing requirements shall be obtained to support the preparation of the Test Plan.

NOTE 1 The software testing requirements includes identification of test item(s).

NOTE 2 The following documentation can be used:

- 1) organizational test specifications, such as the Organizational Test Policy and the Organizational Test Strategy;
- 2) project management plan, for information that will affect the testing, such as the allocated testing budget and resources;

- 3) higher level test plans (e.g. project test plan if managing a lower level of testing, such as system testing) for requirements and constraints on this level of testing, such as testing estimates, staffing and expected deliverables and their timing;
  - 4) applicable regulatory standards for information on regulations that may impact the testing;
  - 5) applicable product documentation, such as system requirement specifications, quality objectives described by system quality characteristics, and test item specifications, for information that relates to possible testing requirements for this phase or type of testing;
  - 6) quality characteristics are defined in ISO/IEC 25010 Systems and software engineering — Systems and software product Quality Requirements and Evaluation (SQuaRE) — System and software quality models;
  - 7) software development plan, for information that may impact on testing timelines or cycles such as expected development deliverables and their timing;
  - 8) project risk register for information on identified project and product risks;
  - 9) verification and validation plan.
- b) An understanding of the context and the software testing requirements should be obtained by identifying and interacting with the relevant stakeholders.
- c) A communication plan should be initiated and lines of communication recorded.

NOTE 3 The activity, Understand Context, will be on-going throughout the lifetime of the project. The tasks in this activity can, in principle, be carried out in any order.

#### 7.2.4.2 Organize Test Plan Development (TP2)

This activity consists of the following tasks:

- a) Based on the testing requirements identified in the *Understand Context* activity (TP1), those activities that need to be performed to complete test planning, shall be identified and scheduled.
- b) The stakeholders required to participate in these activities should be identified.
- c) Approval of the activities, schedule and participants shall be obtained from the relevant stakeholders.

EXAMPLE 1 The Project Manager and/or Project Test Manager.

NOTE This could require repeating tasks a) and b).

- d) Stakeholder involvement should be organized.

EXAMPLE 2 Request project manager to schedule a meeting for review of the test strategy

#### 7.2.4.3 Identify and Analyze Risks (TP3)

This activity consists of the following tasks:

- a) Any risks that have been previously identified shall be reviewed to identify those that relate to and/or can be treated by software testing.

EXAMPLE 1 Risks held in the project risk register

- b) Additional risks that relate to and/or can be treated by software testing shall be identified.

NOTE 1 Any identified risks that are not related to software testing should be communicated to the relevant stakeholders.

NOTE 2 This could be achieved by reviewing product specifications and other appropriate documentation, through workshops, interviews or by other suitable means.



- c) The risks shall be classified using an appropriate classification scheme that, at a minimum, discriminates between project and product risks.
- d) Each risk shall be assigned a level of exposure (such as by considering its impact and likelihood).
- e) Approval shall be obtained for the results of this risk assessment from the stakeholders.
- f) The results of this risk assessment shall be recorded.

EXAMPLE 2 In the test plan, in the project risk register.

#### 7.2.4.4 Identify Risk Mitigation Approaches (TP4)

This activity consists of the following tasks:

- a) Appropriate means of treating the risks shall be identified, based on the risk type, classification and level of risk exposure.

NOTE Appropriate means could include test phases, test types, test design techniques, test completion criteria, etc.,. Practitioners could consider the concept of software criticality covered in ISO/IEC 15026 or IEEE 1012:2012. Where constraints (such as time and cost) on testing are known, the mitigations for risks with low risk exposure levels that are not expected to be treatable within these constraints will be identified as being out of scope for that reason.

- b) The results of the risk mitigation shall be recorded.

EXAMPLE In the test plan, in the project risk register.

#### 7.2.4.5 Design Test Strategy (TP5)

This activity consists of the following tasks:

- a) An initial estimate of the resources required to implement the requirements defined by organizational test specifications, such as the Organizational Test Strategy and the Organizational Test Policy, should be produced. Constraints imposed by higher level test strategies on the project should also be considered.

NOTE 1 Of particular importance are estimates of the effort and elapsed time required.

- b) An initial estimate of the resources required to perform the individual mitigation actions identified in the *Identify Risk Mitigation Approaches* activity (TP4) should be produced, starting with those corresponding to risks with the highest exposure levels as determined in the *Identify and Analyze Risks* activity (TP3).

NOTE 2 Of particular importance are estimates of the effort and elapsed time required.

- c) A Test Strategy (comprising choices including test phases, test types, features to be tested, test design techniques, test completion criteria, and suspension and resumption criteria) shall be designed that considers test basis, risks, and organizational, project and product constraints.

NOTE 3 This takes into consideration the level of risk exposure to prioritise the test activities, the initial test estimates, the resources needed to perform actions (e.g. skills, tool support and environment needs), and organizational, project and product constraints, such as:

- a) regulatory standards;
- b) the requirements of the Organizational Test Policy, Organizational Test Strategy and the Project Test Plan (if designing a test strategy for a lower level of testing);
- c) contractual requirements;
- d) project time and cost constraints;
- e) availability of appropriately-skilled testers;
- f) availability of tools and environments;
- g) technical, system or product limitations.

Where it is not possible to design a test strategy that implements all the requirements of the Organizational Test Strategy and the recommendations for treating all identified risks while still meeting the project and product constraints, then a judgement is made to arrive at a test strategy that best meets these conflicting requirements. How this compromise is achieved will vary dependent on the project and on the organization, and could require the constraints to be relaxed and the Identify Risk Mitigation Approaches activity (TP4) and tasks a) to c) to be repeated until an acceptable test strategy is achieved. Where it is decided to deviate from the Organizational Test Strategy, this should be recorded in the test strategy.

NOTE 4 A test strategy would typically address static testing (e.g. reviews, inspections, static analysis) as well as dynamic testing.

d) Metrics to be used for Test Monitoring and Control (see activities TMC1 to TMC4) shall be identified.

e) Test data shall be identified.

EXAMPLE Factors to consider when identifying test data include regulations on data confidentiality (it could require data masking or encryption), volume of data required and data clean-up upon completion.

f) Test environment requirements and test tool requirements shall be identified.

g) Test deliverables shall be identified and their degree of formality and frequency of communication should be recorded.

h) An initial estimate of the required resources to perform the complete set of actions described in the test strategy shall be produced.

NOTE 5 The initial test estimate that is produced in this step is finalised in Record Test Plan (TP7).

i) The test strategy shall be recorded.

NOTE 6 The test strategy will normally be a section of the test plan, but in some cases it could be recorded as a separate document.

j) Approval on the test strategy shall be obtained from the stakeholders.

NOTE 7 This could require repeating earlier tasks in this activity.

#### 7.2.4.6 Determine Staffing and Scheduling (TP6)

This activity consists of the following tasks:

a) The roles and skills of staff to carry out the testing described in the test strategy should be identified.

NOTE 1 This could require identification of staff recruitment and/or training needs.

b) Each required test activity in the Test Strategy shall be scheduled based on the estimates, dependencies and staff availability.

c) Approval on staffing and scheduling shall be obtained from the relevant stakeholders.

NOTE 2 This could require repeating tasks a) and b), and if the test strategy needs revision, then the Design Test Strategy activity (TP5) will need to be revisited.

#### 7.2.4.7 Record Test Plan (TP7)

This activity consists of the following tasks:

a) Final estimates for the testing shall be calculated based on the test strategy designed in the *Design Test Strategy* activity (TP5) and the staffing and scheduling agreed in the *Determine Staffing and Scheduling* activity (TP6).

NOTE Where these disagree with previous initial estimates it could be necessary to revisit the Determine Staffing and Scheduling (TP6) and/or the Design Test Strategy (TP5) activities.

- b) The Test Strategy identified in the *Design Test Strategy* activity (TP5), the staffing profile and schedule agreed in the *Determine Staffing and Scheduling* activity (TP6), and the final estimates calculated in the previous task shall be incorporated in the test plan.

#### 7.2.4.8 Gain Consensus on Test Plan (TP8)

This activity consists of the following tasks:

- a) The views of the stakeholders on the test plan shall be gathered.

NOTE 1 This could be achieved through workshops, interviews or by other suitable means.

- b) Conflicts between the test plan and stakeholders' views shall be resolved.
- c) The test plan shall be updated to take into account feedback from stakeholders.

NOTE 2 This could require repeating earlier activities in the Test Planning Process.

- d) Approval on the test plan shall be obtained from the stakeholders.

NOTE 3 This could require repeating tasks a) to c).

#### 7.2.4.9 Communicate Test Plan and Make Available (TP9)

This activity consists of the following tasks:

- a) The Test Plan shall be made available.
- b) The availability of the Test Plan shall be communicated to the stakeholders.

NOTE This could require a communication plan to be developed.

#### 7.2.5 Information items

As a result of carrying out this process, the following information item shall be produced:

- a) Test Plan.

### 7.3 Test Monitoring and Control Process

#### 7.3.1 Overview

The *Test Monitoring and Control Process* as shown in Figure 7 scrutinizes whether testing progresses in accordance with the Test Plan and the organizational test specifications, such as the Organizational Test Policy and the Organizational Test Strategy. If there are significant departures from planned progress, activities, or other aspects of the test plan, activities will be initiated to correct or compensate for the resultant variances.

This process can be applied to the management of a whole test project (normally made up of a number of test phases and test types) or to the management of the testing of a single test phase (e.g. system testing) or test type (e.g. performance testing). In the latter case it is applied as part of the monitoring and control of dynamic testing described by the *Dynamic Test Processes*. When applied as part of the monitoring and control of the testing for a complete project then it will interact directly with the test management processes being used to manage the individual test phases and test types of the project.

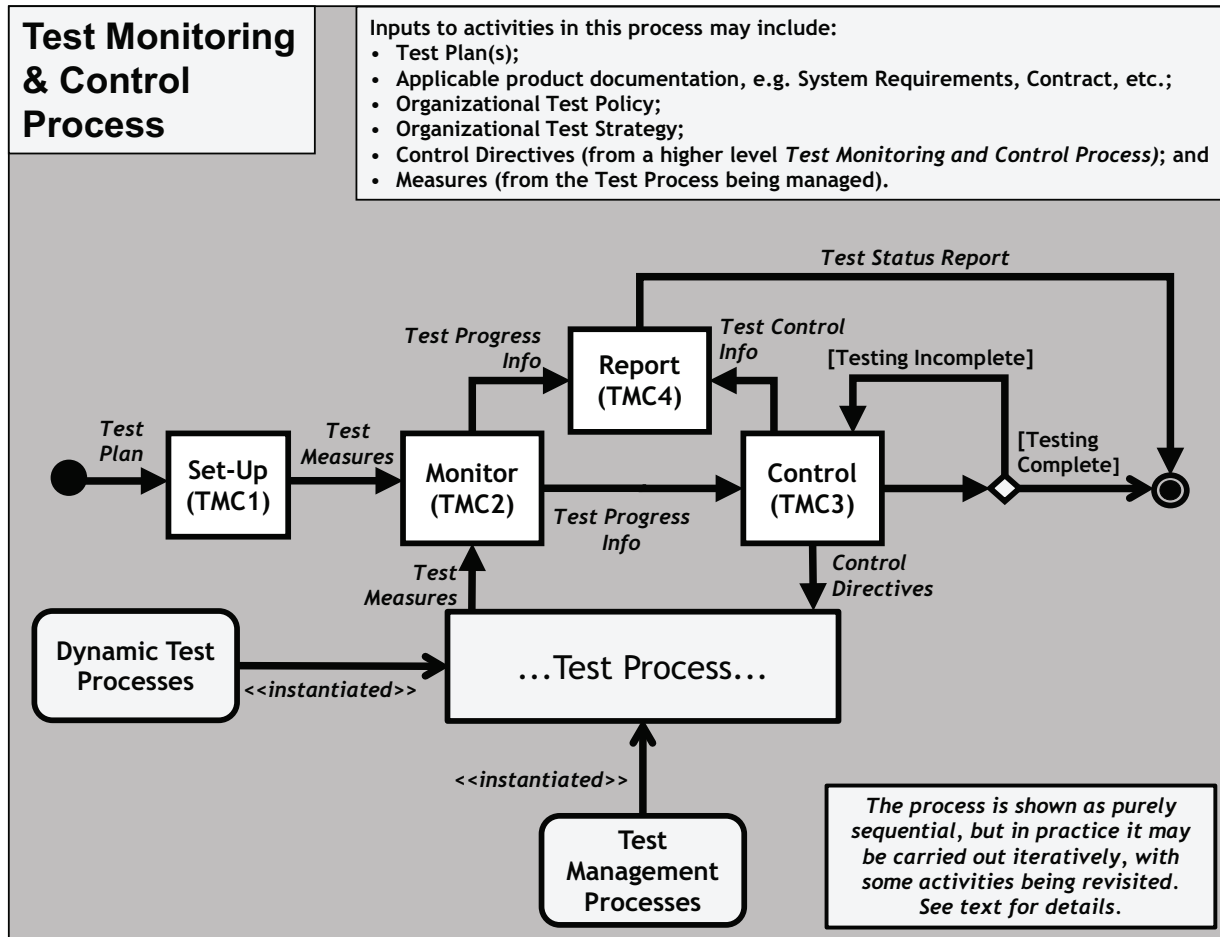


Figure 7 — Test monitoring and control process

### 7.3.2 Purpose

The purpose of the *Test Monitoring and Control Process* is to determine whether testing progresses in accordance with the Test Plan and with organizational test specifications (e.g. the Organizational Test Policy and the Organizational Test Strategy). It also initiates control actions as necessary and identifies necessary updates to the Test Plan (e.g. revise completion criteria or identify new actions to compensate for deviations from the Test Plan).

The process is also used to determine whether testing progresses in accordance with higher level test plans, such as the Project Test Plan, and to manage the testing performed at specific test phases (e.g. system testing) or for specific test types (e.g. performance testing).

### 7.3.3 Outcomes

As a result of the successful implementation of the Test Monitoring and Control Process:

- The means of collecting suitable measures to monitor test progress and changing risk are set up;
- Progress against the test plan is monitored;
- New and changed test-related risks are identified, analysed and necessary action(s) invoked;
- Necessary control actions are identified;

- e) Necessary control actions are communicated to the relevant stakeholders;
- f) The decision to stop testing is approved;
- g) Test progress and changes to the risks are reported to stakeholders.

#### 7.3.4 Activities and tasks

The person(s) responsible for test monitoring and control shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Monitoring and Control Process*.

##### 7.3.4.1 Set-Up (TMC1)

This activity consists of the following tasks:

- a) Suitable measures for monitoring progress against the Test Plan should be identified, if these measures are not already defined in the Test Plan or the Organizational Test Strategy.
- b) Suitable means of identifying new and changing risks should be identified, if these are not already defined in the Test Plan or the Organizational Test Strategy.
- c) Monitoring activities, such as test status reporting and test metrics collection, shall be put in place to collect the measures identified in tasks a) and b) above, and in the Test Plan and the Organizational Test Strategy.

##### 7.3.4.2 Monitor (TMC2)

This activity consists of the following tasks:

- a) The test measures shall be collected and recorded.
- b) Progress against the Test Plan shall be monitored using the collected test measures.

EXAMPLE 1 By examining test status reports, analysing test measures and meeting with stakeholders.

- c) Divergence from planned testing activities shall be identified and any factors blocking progress recorded.
- d) New risks shall be identified and analysed to identify those that require mitigation by testing and those that need to be communicated to other stakeholders.
- e) Changes to known risks shall be monitored to identify those that require mitigation by testing and those that need to be communicated to other stakeholders.

EXAMPLE 2 Communicate risks that require testing as mitigation to the Project Manager.

NOTE Tasks a) to e) above are repeated on a regular basis, until it is determined that the testing specified in the Test Plan can be terminated or is complete, which would typically be by checking whether the completion criteria have been achieved.

##### 7.3.4.3 Control (TMC3)

This activity consists of the following tasks:

- a) Those actions necessary to implement the test plan shall be performed.

EXAMPLE 1 Assigning responsibility for testing activities to testers.

- b) Those actions necessary to implement control directives received from higher level management processes shall be performed.

EXAMPLE 2 Actions from the project test manager if a specific phase of testing is being managed.

- c) Those actions necessary to manage the divergence of actual testing from planned testing shall be identified.

NOTE 1 These control actions could require changes to the testing, the test plan, test data, test environment, staffing and/or changes in other areas, such as development.

- d) Means of treating newly-identified and changed risks shall be identified.

NOTE 2 This could include assigning more staff to specific tasks and changing test completion criteria.

- e) As appropriate:

- 1) control directives shall be issued to make changes to the way testing is performed;
- 2) changes to the test plan shall be in the form of test plan updates; and
- 3) recommended changes shall be communicated to the relevant stakeholders.

EXAMPLE 3 IT Support for test environments.

- f) Readiness for commencing any assigned test activity shall be established before commencing that activity, if not already done.

NOTE 3 This could typically be performed by checking against entry criteria described in the test plan.

NOTE 4 Assigned test activity could be test execution.

NOTE 5 Readiness could have been established in the test design and implementation process and/or the test environment set-up process.

- g) Approval shall be granted at the completion of assigned test activities.

EXAMPLE 4 Completion of a lower level of testing.

NOTE 6 This will typically be performed by checking against exit criteria described in the test plan.

- h) When the testing has met its completion criteria, approval for the test completion decision shall be obtained.

#### 7.3.4.4 Report (TMC4)

This activity consists of the following tasks:

- a) Testing progress against the Test Plan shall be communicated to stakeholders in a test status report for the specified reporting period.
- b) New risks and changes to existing risks shall be updated in the risk register and communicated to the relevant stakeholders.

#### 7.3.5 Information Items

As a result of carrying out this process, the following information items shall be produced:

- a) Test status reports;
- b) Test plan updates;

- c) Control directives (e.g. changes to the testing, the test plan, test data, test environment and staffing);
- d) Project and product risk information.

NOTE Risk information could be held in the project risk register or locally in the Test Plan.

## 7.4 Test Completion Process

### 7.4.1 Overview

The *Test Completion Process* as shown in Figure 8 is performed when agreement has been obtained that the testing activities are complete. It will be performed to complete the testing carried out at a specific test phase (e.g. system testing) or test type (e.g. performance testing) and to complete the testing for a complete project.

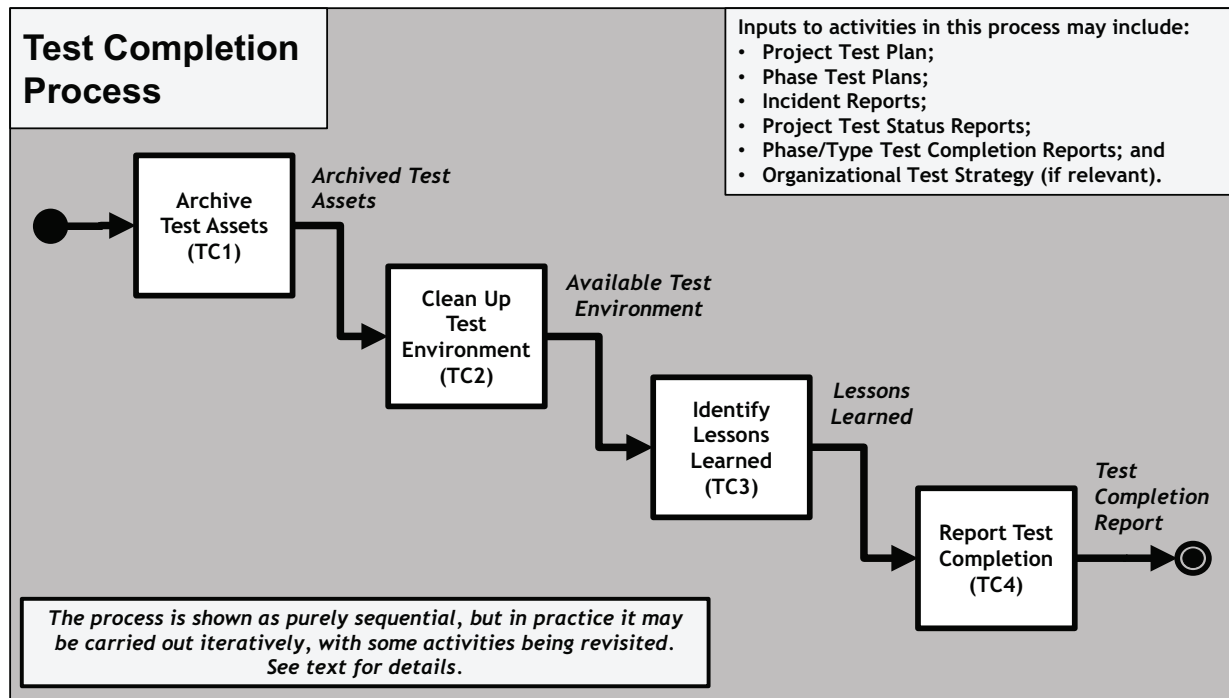


Figure 8 — Test completion process

### 7.4.2 Purpose

The purpose of the *Test Completion Process* is to make available useful test assets for later use, leave the test environment in a satisfactory condition, and record and communicate the results of the testing to relevant stakeholders. Test assets include Test Plans, Test Case Specifications, test scripts, test tools, test data and test environment infrastructure.

### 7.4.3 Outcomes

As a result of the successful implementation of the *Test Completion Process*:

- a) Test assets are either archived or passed directly to the relevant stakeholders;
- b) The test environment is in its agreed state (e.g. so that it is available for the next test project);
- c) All test requirements are satisfied and verified;
- d) The Test Completion Report is recorded;

- e) The Test Completion Report is approved;
- f) The Test Completion Report is communicated to relevant stakeholders.

#### 7.4.4 Activities and tasks

The person(s) responsible for test completion shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Completion Process*.

##### 7.4.4.1 Archive Test Assets (TC1)

This activity consists of the following tasks:

- a) Those test assets which may be of use at a later date should be identified and made available using appropriate means.

EXAMPLE 1 The test assets to be reused (e.g. for regression testing) are appropriately labelled within the configuration management system.

- b) Those test assets which may be reused on other projects should be identified and archived.

EXAMPLE 2 Test plans, manual and/or automated test procedures, test environment infrastructure.

- c) The availability of reusable test assets shall be recorded in the Test Completion Report and communicated to the relevant stakeholders.

EXAMPLE 3 Those responsible for maintenance testing (to achieve successful transition) and the project test manager.

##### 7.4.4.2 Clean Up Test Environment (TC2)

This activity consists of the following task:

- a) The test environment shall be restored to a pre-defined state on completion of all testing activities.

EXAMPLE Restore settings and hardware to original state.

##### 7.4.4.3 Identify Lessons Learned (TC3)

This activity consists of the following tasks:

- a) Lessons learned during the project execution shall be recorded.

NOTE This could be achieved by recording:

- 1) what went well during testing and associated activities;
- 2) what did not go well during testing and associated activities;
- 3) recommended improvements to the testing and other processes, such as the development process.

- b) The outcomes shall be recorded for inclusion in the Test Completion Report and communicated to the relevant stakeholders.

##### 7.4.4.4 Report Test Completion (TC4)

This activity consists of the following tasks:

- a) Relevant information shall be collected from the following documents, but not limited to:
  - 1) Test Plans (e.g. project test plan, system test plan, or performance test plan);



- 2) Test Results;
- 3) Test Status Reports;
- 4) Test Completion Reports from test phase or test type; and

EXAMPLE From unit testing, performance testing, acceptance testing, etc., if this is reporting on the completion of testing for the whole project.

- 5) Incident Reports.
- b) The collected information shall be evaluated and summarized in the Test Completion Report.
- c) Approval for the Test Completion Report shall be obtained from the responsible stakeholder(s).
- d) The approved Test Completion Report shall be distributed to the relevant stakeholders.

#### 7.4.5 Information Items

As a result of carrying out this process, the following information item shall be produced:

- a) Test Completion Report.

## 8 Dynamic Test Processes

### 8.1 Introduction

The Dynamic Test Processes are used to carry out dynamic testing within a particular phase of testing (e.g. unit, integration, system and acceptance) or type of testing (e.g. performance testing, security testing, usability testing). The processes for the management of this dynamic testing are described in clause 7, Test Management Processes.

There are four dynamic test processes (as show in Figure 9):

- a) test design & implementation;
- b) test environment set-up & maintenance;
- c) test execution; and
- d) test incident reporting.

Figure 9 illustrates how the dynamic test processes interact and the relationship with the test management processes. These dynamic test processes would normally be invoked as part of the implementation of the test strategy documented within the test plan for the test phase (e.g. system testing) or test type (e.g. performance testing) being performed.

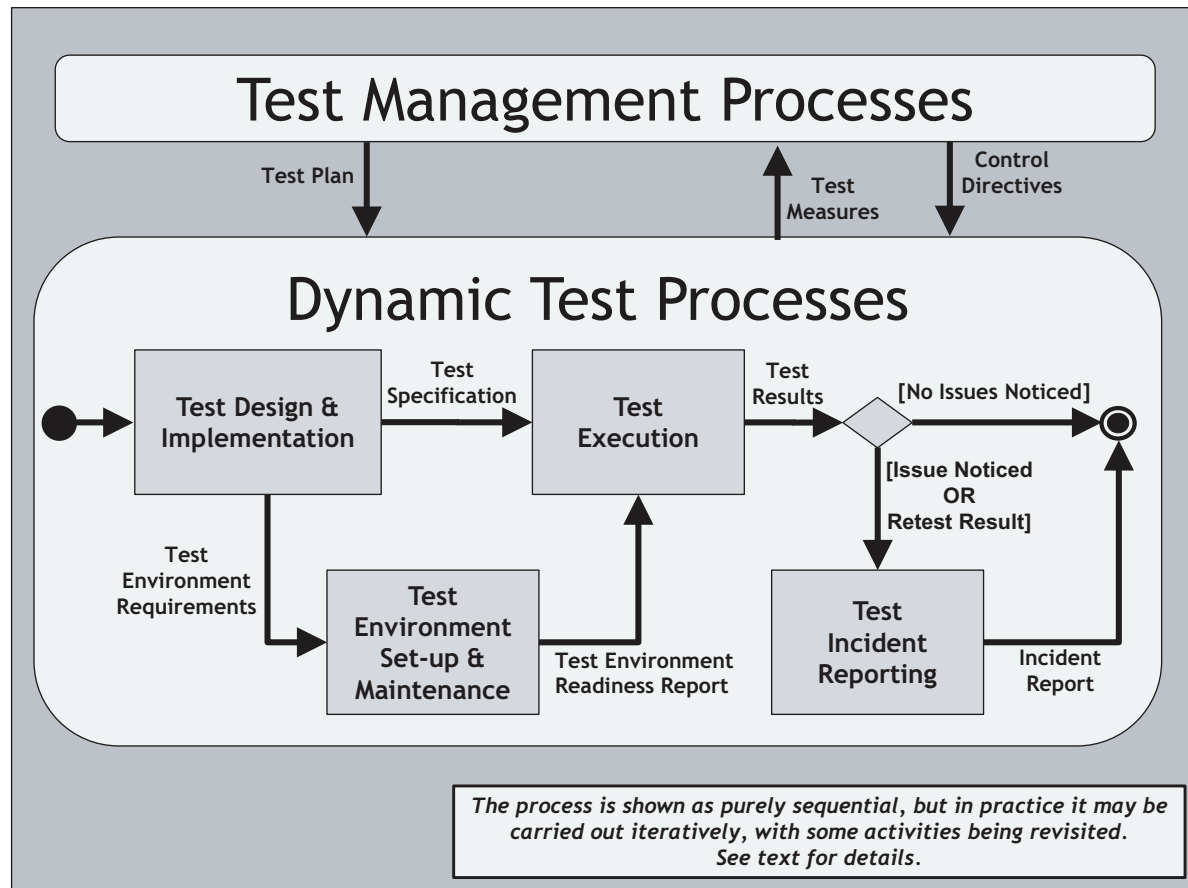


Figure 9 — Dynamic test processes

For any particular test, the dynamic test processes will execute in the order shown in Figure 9, but these processes will typically be invoked a number of times to complete the testing for a given test phase (e.g. system testing) or test type (e.g. performance testing). This is because as tests are designed and run, the overseeing test management process (*Test Monitoring and Control*) monitors test progress (through test measures) and may require further tests (through control directives) to be designed and run until the test completion criterion for this test activity is achieved.

Test measures, which are an output of the dynamic test processes and an input into the *Test Monitoring and Control Process* (see Figure 7), can be produced during any activity of the dynamic test processes. Test measures are used to report the status and progress of testing to test management staff. For example, test measures could be used to indicate to test management how many test cases have been designed by the testing team.

Similarly, control directives are an output of the test management process and an input to the dynamic test process (see Figure 7), and can be acted upon during any activity of the dynamic test processes. Control directives correspond to instructions from test management staff that dictate how dynamic testing should be conducted by the test team. For example, a control directive could be given to a test team that instructs them to design additional test cases for new program features that have been allocated to their team by their test manager.

Since test measures can be produced during any activity of the dynamic test processes, and since control directives can be acted upon during any activity of these processes, the production of measures and handling of directives are not shown as tasks in any specific activity of these processes.

## 8.2 Test Design & Implementation Process

### 8.2.1 Overview

The *Test Design & Implementation Process* is used to derive test cases and test procedures; these are normally documented in a test specification, but may be immediately executed, for instance, if performing exploratory testing, in which case they are unlikely to be documented in advance. In Figure 10 the activities are shown in a logical sequence, but in practice iteration will occur between many of the activities, often with activities TD3 to TD5 occurring in parallel for substantial periods.

This process is used to derive test cases and test procedures, but it should be noted that in some cases it may be possible to reuse previously designed test assets, especially in the situation where regression tests are being developed.

The *Test Design & Implementation Process* may also be exited and re-entered for a number of reasons, for example, if, after executing a test procedure or reporting an incident, it is realized that additional test cases are required in order to meet the required test completion criteria. Thus, it is possible that only a subset of all the test cases that are required for a test item may be derived during any one implementation of this process.

This process requires testers to apply one or more test design techniques to derive test cases and test procedures with the ultimate aim of achieving the test completion criteria, typically described in terms of test coverage measures. Those test design techniques and test completion criteria to use are specified in the Test Plan. Test design techniques and measures are defined in ISO/IEC/IEEE 29119-4: Test Techniques.

A number of situations can cause iteration between activities in this process. Such situations include stakeholders failing to agree with the results of an activity, such as the identification of test conditions. Similarly, situations can occur where the results of an activity make it clear that test planning decisions, such as the choice of test completion criteria, are incompatible with constraints on project timelines, requiring Test Management processes to be revisited.

NOTE An example application of activities TD2 to TD5 is provided in Annex A.

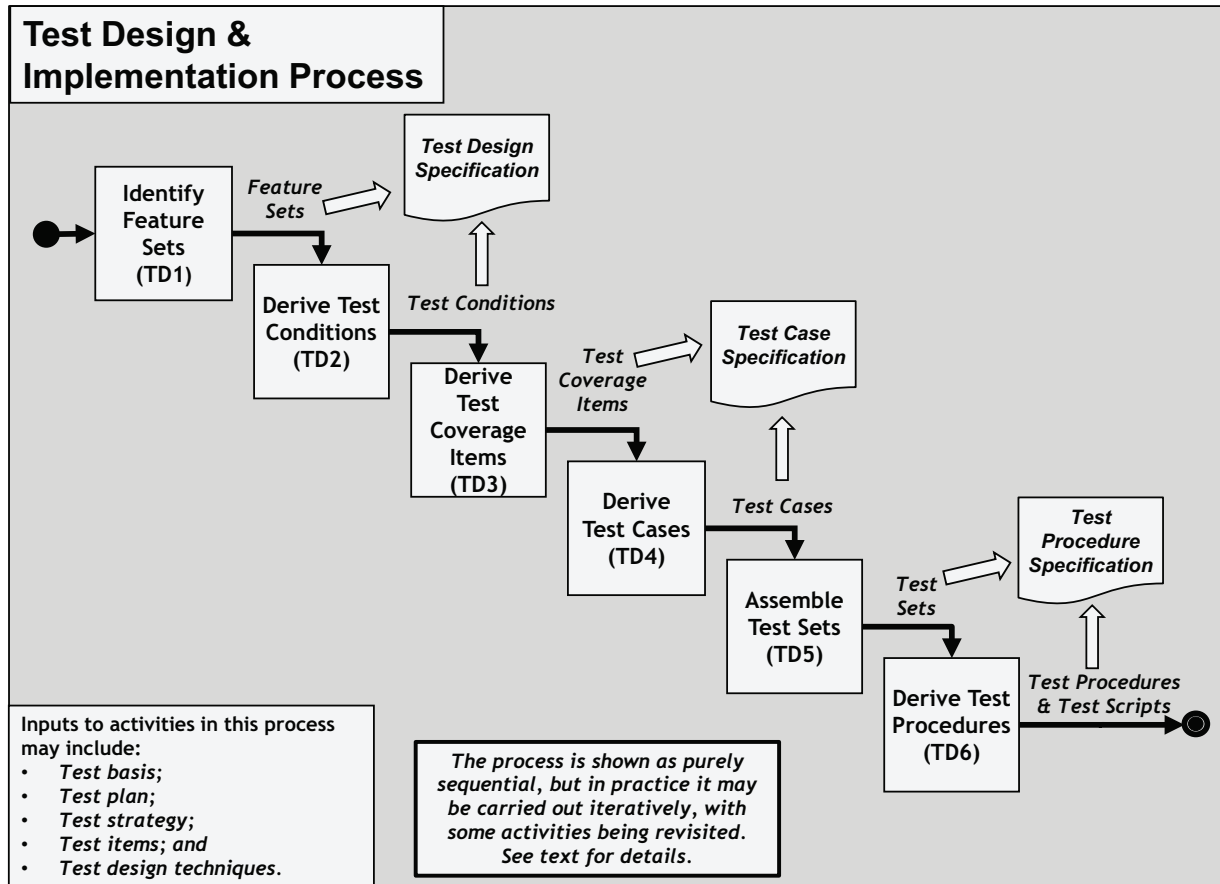


Figure 10 — Test Design and Implementation Process

### 8.2.2 Purpose

The purpose of the *Test Design & Implementation Process* is to derive test procedures that will be executed during the *Test Execution Process*. As part of this process the test basis is analysed, features are combined into feature sets, test conditions, test coverage items, test cases, test procedures are derived and test sets are assembled.

### 8.2.3 Outcomes

As a result of the successful implementation of the *Test Design & Implementation Process*:

- The test basis for each test item is analysed;
- The features to be tested are combined into Feature Sets;
- The Test Conditions are derived;
- The Test Coverage Items are derived;
- Test Cases are derived;
- Test Sets are assembled;
- Test Procedures are derived.

## 8.2.4 Activities and tasks

The person(s) responsible for test design and implementation shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Design & Implementation Process*.

### 8.2.4.1 Identify Feature Sets (TD1)

This activity consists of the following tasks:

- a) The test basis shall be analyzed to understand the requirements for the test item.

NOTE 1 If defects in the test basis are uncovered during this analysis these are reported using an appropriate incident management system.

- b) The features to be tested should be combined into feature sets.

NOTE 2 A feature set could be tested independently of other feature sets.

NOTE 3 For component testing/unit, there could be only a single feature set; for higher levels of testing (e.g. system testing) there could be a number of feature sets that would typically mirror the architecture of the test item.

NOTE 4 Where multiple feature sets have not been identified the set of features will be treated as a single feature set.

- c) The testing of the feature sets shall be prioritized using the risk exposure levels documented in the Identify and Analyze Risks activity (TP3).

- d) The composition and prioritization of feature sets should be agreed with the stakeholders.

NOTE 5 Where necessary, tasks a), b) and c) are revisited.

- e) The feature set(s) shall be documented in the test design specification.

- f) The traceability between the test basis and feature set(s) shall be recorded.

NOTE 6 Tasks c) to f) are applicable if feature sets were identified in task b).

### 8.2.4.2 Derive Test Conditions (TD2)

This activity consists of the following tasks:

- a) Based on the test completion criteria specified in the Test Plan, the test conditions for each feature shall be determined.

NOTE 1 A test condition is a testable aspect of a component or system, such as a function, transaction, feature, quality attribute, or structural element identified as a basis for testing. This determination can be performed simply by agreeing on those attributes of specific interest to stakeholders that are to be tested or by applying one or more systematic techniques (techniques for deriving test conditions are often described as the modelling or analysis part of test design techniques, which are defined in ISO/IEC/IEEE 29119-4 Test Techniques). For instance, if a test completion criterion related to state coverage was specified then the test conditions would be the states the test item could be in. Other examples of test conditions may be equivalence classes (and boundaries between them) and decisions in the code.

- b) The test conditions shall be prioritized using the risk exposure levels documented in the *Identify and Analyze Risks* activity (TP3).

- c) The test conditions shall be recorded in the test design specification.

NOTE 2 When performing exploratory testing the test design specification can take different forms including a test charter.

- d) The traceability between the test basis, feature sets and test conditions shall be recorded.
- e) The test design specification shall be approved by the stakeholders.

NOTE 3 This could require repeating tasks a), b) and c), or first repeating the Identify Feature Sets activity (TD1).

#### 8.2.4.3 Derive Test Coverage Items (TD3)

This activity consists of the following tasks:

- a) The test coverage items to be exercised by the testing shall be derived by applying test design techniques to the test conditions to achieve the test completion coverage criteria specified in the Test Plan.

NOTE 1 The test coverage items are attributes of each test condition. For instance, where a boundary was identified as a test condition then the corresponding test coverage items could be the boundary itself, and either side of the boundary; thus a single test condition could be the basis for one or more test coverage items.

NOTE 2 Where a test completion criterion for the test item is specified as less than 100% of a test coverage measure, a subset of the test coverage items required to achieve 100 % coverage needs to be selected to be exercised by the testing.

NOTE 3 There could be criteria provided to aid in this selection in the Test Plan or Organizational Test Strategy (e.g. discard test coverage items associated with lower risk exposures). This selection could need to be revisited based on the results of later activities.

NOTE 4 The set of test coverage items can be optimized by combining coverage of multiple test conditions into a single test coverage item; thus a single test coverage item can exercise more than one test condition.

- b) The test coverage items shall be prioritized using the risk exposure levels documented in the *Identify and Analyze Risks* activity (TP3).
- c) The test coverage items shall be recorded in the test case specification.
- d) The traceability between the test basis, feature sets, test conditions and test coverage items shall be recorded.

#### 8.2.4.4 Derive Test Cases (TD4)

This activity consists of the following tasks:

- a) One or more test cases shall be derived by determining pre-conditions, selecting input values and, where necessary, actions to exercise the selected test coverage items, and by determining the corresponding expected results.

NOTE 1 When deriving the test cases, one test case could exercise more than one test coverage item and thus there is the opportunity to combine coverage of multiple test coverage items in a single test case. This could reduce test execution times, but could also increase debugging times.

- b) The test cases shall be prioritized using the risk exposure levels documented in the *Identify and Analyze Risks* activity (TP3).
- c) The test cases shall be recorded in the test case specification.
- d) The traceability between the test basis, feature sets, test conditions, test coverage items and test cases shall be recorded.
- e) The test case specification shall be approved by the stakeholders.

NOTE 2 This could require repeating tasks a) and b), and in some cases, first repeating the Derive Test Conditions (TD2) and/or Derive Test Coverage Items (TD3) activities.

#### 8.2.4.5 Assemble Test Sets (TD5)

This activity consists of the following tasks:

- a) The test cases may be distributed into one or more test sets based on constraints on their execution.

NOTE Where multiple test sets have not been identified the set of test cases will be treated as a single test set.

EXAMPLE For instance, some test sets could require a specific test environment set-up, or some could be appropriate for manual test execution while others are more appropriate for automated test execution, or some could require specific domain knowledge.

- b) The test sets shall be recorded in the test procedure specification.
- c) The traceability between the test basis, feature sets, test conditions, test coverage items, test cases and test sets shall be recorded.

#### 8.2.4.6 Derive Test Procedures (TD6)

This activity consists of the following tasks:

- a) Test procedures shall be derived by ordering test cases within a test set according to dependencies described by pre-conditions and post-conditions and other testing requirements.

EXAMPLE Risks to be treated.

NOTE 1 Any other required actions could be included in the test procedure, such as those necessary to set up the pre-conditions for a test case.

NOTE 2 Where test procedures are to be executed using tools, it could be necessary to further elaborate them by adding extra detail to create automated test scripts.

- b) Any test data and test environment requirements that are not already included in the Test Plan shall be identified.

NOTE 3 Although this activity might not be finalized until the derivation of test procedures is complete, this task could often start far earlier in the process, sometimes even as early as when test conditions are agreed.

- c) The test procedures shall be prioritized using the risk exposure levels documented in the *Identify and Analyze Risks* activity (TP3).
- d) The test procedures shall be recorded in the test procedure specification.
- e) The traceability between the test basis, feature sets, test conditions, test coverage items, test cases, test sets and test procedures (and/or automated test scripts) shall be recorded.
- f) The test procedure specification shall be approved by the stakeholders.

NOTE 4 This could require repeating tasks a) to e).

#### 8.2.5 Information Items

As a result of carrying out this process, the following information items shall be produced:

- a) Test Specifications (test design specifications, test case specifications and test procedure specifications) and related traceability information;
- b) Test data requirements;
- c) Test environment requirements.

### 8.3 Test Environment Set-Up & Maintenance Process

#### 8.3.1 Overview

The *Test Environment Set-Up & Maintenance Process* is used to establish and maintain the environment in which tests are executed. Maintenance of the test environment may involve changes based on the results of previous tests. Where change and configuration management processes exist, changes to the test environments may be managed using these processes.

The requirements for a test environment will initially be described in the Test Plan, but the detailed composition of the test environment will normally only become clear once the *Test Design & Implementation Process* has started.

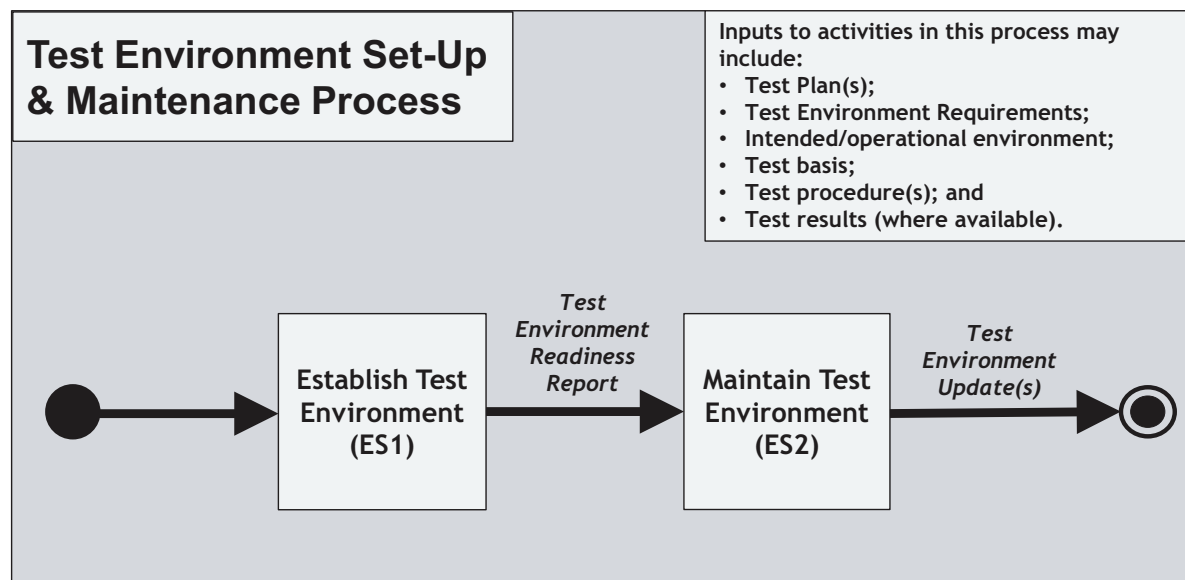


Figure 11 — Test Environment Set-Up & Maintenance process

#### 8.3.2 Purpose

The purpose of the *Test Environment Set-Up & Maintenance Process* is to establish and maintain the required test environment and to communicate its status to all relevant stakeholders.

#### 8.3.3 Outcomes

As a result of the successful implementation of the *Test Environment Set-Up & Maintenance Process*:

- The test environment is set-up in a state ready for testing;
- The status of the test environment is communicated to all relevant stakeholders;
- The test environment is maintained.

#### 8.3.4 Activities and tasks

The person(s) responsible for test environment set-up and maintenance (such as IT support technicians) shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Environment Set-Up & Maintenance Process*.



#### 8.3.4.1 Establish Test Environment (ES1)

This activity consists of the following tasks:

- a) Based on the Test Plan, the detailed requirements generated as a result of the *Test Design & Implementation Process*, the requirement for test tools, and the scale/ formality of the testing, the following shall be performed:

- 1) Plan the set-up of the test environment;

EXAMPLE Requirements, interfaces, schedules and costs.

- 2) Design the test environment;
- 3) Determine the degree of configuration management to be applied (where appropriate);
- 4) Implement the test environment;

NOTE 1 This could include hardware and software items, as appropriate.

- 5) Set up test data to support the testing (where appropriate);
- 6) Set up test tools to support the testing (where appropriate);
- 7) Install and configure the test item on the test environment;
- 8) Verify that the test environment meets the test environment requirements; and
- 9) Where required, provide assurance that the test environment meets the defined requirements.

- b) The status of the test environment and test data shall be recorded and communicated through the Test Environment Readiness Report and Test Data Readiness Report to the relevant stakeholders.

NOTE 2 Relevant stakeholders could include testers and the test manager.

- c) The Test Environment Readiness report shall include a description of the known differences between the test environment and the operational environment.

#### 8.3.4.2 Maintain Test Environment (ES2)

This activity consists of the following tasks:

- a) The test environment shall be maintained as defined by the test environment requirements.

NOTE This could require making changes based on the results of previous tests.

- b) Changes to the status of the test environment shall be communicated to the relevant stakeholders.

EXAMPLE The testers and the test manager.

#### 8.3.5 Information Items

As a result of carrying out this process, the following information items shall be produced:

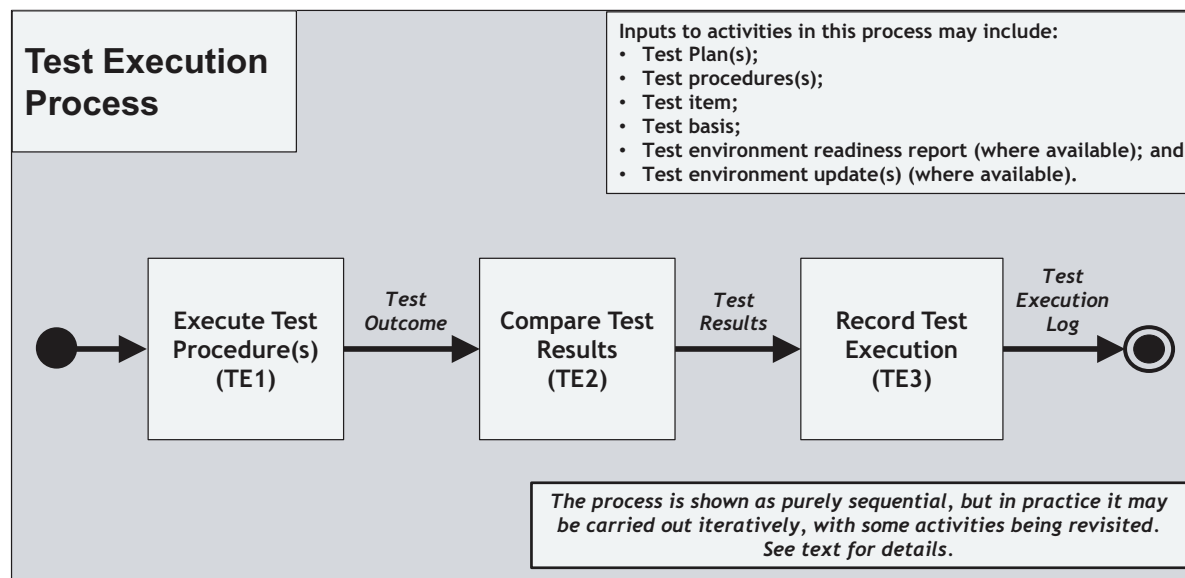
- a) Test Environment;
- b) Test Data;
- c) Test Environment Readiness Report;

- d) Test Data Readiness Report;
- e) Test Environment Updates (where applicable);

## 8.4 Test Execution Process

### 8.4.1 Overview

The *Test Execution Process* is used to run the test procedures generated as a result of the *Test Design & Implementation Process* on the test environment established by the *Test Environment Set-Up & Maintenance Process*. The *Test Execution Process* may need to be performed a number of times as all the available test procedures may not be executed in a single iteration. If an issue is fixed it should be retested by re-entering the *Test Execution Process*.



**Figure 12 — Test Execution process**

In figure 12 the activities are shown in a logical sequence, but in practice iteration will occur between many of the activities. Comparison of test results and recording of test execution details will normally be interleaved with the execution of test procedures.

### 8.4.2 Purpose

The purpose of the *Test Execution Process* is to execute the test procedures created in the *Test Design & Implementation Process* in the prepared test environment and record the results.

### 8.4.3 Outcomes

As a result of the successful implementation of the *Test Execution Process*:

- a) The test procedure(s) is/are executed;
- b) The actual results are recorded;
- c) The actual and expected results are compared; and
- d) The test results are determined.

#### 8.4.4 Activities and tasks

The person(s) responsible for test execution shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Execution Process*.

NOTE 1 The Test Execution Process can be interrupted by circumstances such as when a defect is discovered in a test case, a problem is discovered in the test environment or changes are made to the Test Plan (e.g. due to project cost or time modifications) or circumstances that have been specified under suspension criteria. In these circumstances the process is resumed at the appropriate task or cancelled altogether.

NOTE 2 The Test Execution Process will be re-entered if, after executing one or more test cases, it is realized that additional test cases need to be executed in order to meet the required test completion criteria. Thus, only a subset of all test cases for a test item could be executed during any one iteration of this process.

##### 8.4.4.1 Execute Test Procedure(s) (TE1)

This activity consists of the following tasks:

- a) One or more test procedures shall be executed in the prepared test environment.

NOTE 1 The test procedures could have been scripted for automated execution, or could have been recorded in a test specification for manual test execution, or could be executed immediately they are designed as in the case of exploratory testing.

- b) The actual results for each test case in the test procedure shall be observed.
- c) The actual results shall be recorded.

NOTE 2 This could be in a test tool or manually, as indicated in the test case specification.

NOTE 3 Where exploratory testing is performed, actual results can be observed, and not recorded.

##### 8.4.4.2 Compare Test Results (TE2)

This activity consists of the following tasks:

- a) The actual and expected results for each test case in the test procedure shall be compared.

NOTE 1 Expected results could have been recorded in the test specification, or could be an undocumented expectation in the case of exploratory testing. In the case of automated testing the expected results are normally embedded in the automated test script (or in an associated file) and the test tool performs the comparison.

- b) The test result of executing the test cases in the test procedure shall be determined. If a retest is passed, this will require an update to an incident report by the *Test Incident Reporting Process*.

NOTE 2 Failures and unexpected changes to the test environment will result in issues (potential incidents) being passed to the Test Incident Reporting process.

##### 8.4.4.3 Record Test Execution (TE3)

This activity consists of the following task:

- a) Test execution shall be recorded, as specified in the Test Plan.

NOTE This would normally be the execution log.

#### 8.4.5 Information Items

As a result of carrying out this process, the following information items shall be produced:

- a) Actual results;
- b) Test results;

EXAMPLE Test passes, failures and instances where something unusual or unexpected occurred.

- c) Test execution log.

## 8.5 Test Incident Reporting Process

### 8.5.1 Overview

The *Test Incident Reporting Process* is used for the reporting of test incidents. This process will be entered as a result of the identification of test failures, instances where something unusual or unexpected occurred during test execution, or when a retest passes.

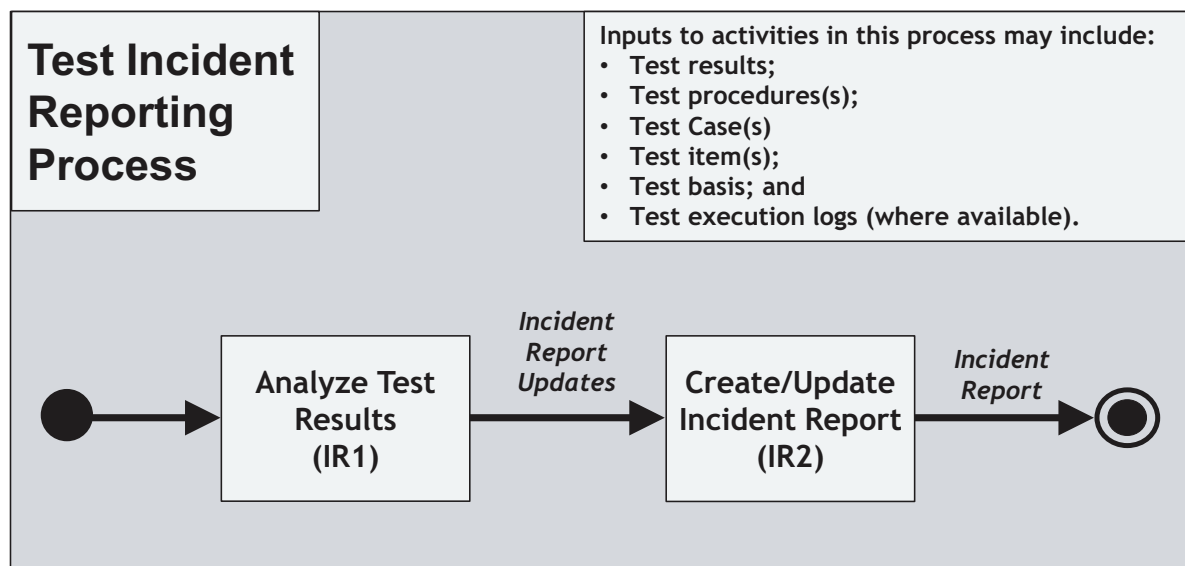


Figure 13 — Test Incident Reporting process

### 8.5.2 Purpose

The purpose of the *Test Incident Reporting Process* is to report to the relevant stakeholders incidents requiring further action identified as a result of test execution. In the case of a new test this will require an incident report to be created. In the case of a retest, this will require the status of a previously-raised incident report to be updated, but may also require a new incident report to be raised where further incidents are identified.

### 8.5.3 Outcomes

As a result of the successful implementation of the *Test Incident Reporting Process*:

- a) Test results are analysed;
- b) New incidents are confirmed;
- c) New incident report details are created;
- d) The status and details of previously-raised incidents are determined;

- e) Previously-raised incident report details are updated as appropriate;
- f) New and/or updated incident reports are communicated to the relevant stakeholders.

#### 8.5.4 Activities and tasks

The person(s) responsible for test incident reporting shall implement the following activities and tasks in accordance with applicable organization policies and procedures with respect to the *Test Incident Reporting Process*.

##### 8.5.4.1 Analyze Test Result(s) (IR1)

This activity consists of the following tasks:

- a) Where a test result relates to a previously-raised incident, the test result shall be analyzed and the incident details shall be updated.
- b) Where a test result indicates that a new issue has been identified, the test result shall be analyzed and it will be determined whether it is an incident that requires reporting, an action item that will be resolved without incident reporting, or requires no further action to be taken.

NOTE Where appropriate, the decision not to raise an incident report is discussed with the originator to aid mutual understanding of this decision.

- c) Action items shall be assigned to an appropriate person for resolution.

##### 8.5.4.2 Create/Update Incident Report (IR2)

This activity consists of the following task:

- a) The information that needs to be recorded about the incident shall be identified and reported/updated.

NOTE 1 Incident reports can be raised against test items and other items such as test procedures, the test basis and the test environment.

NOTE 2 Following a successful re-test, the incident report can be updated and closed.

- b) The status of new and/or updated incidents shall be communicated to the relevant stakeholders.

#### 8.5.5 Information Items

As a result of carrying out this process, the following information item shall be produced:

- a) Incident Report.

## Annex A

(informative)

## Partial Example Test Design Process

The following is an example application of activities TD2 to TD5 of the *Test Design and Implementation* process.

### Fragment of Test Basis

“The system shall accept insurance applicants over the age of 18 and under the age of 80 years on the day of application based on their input age in whole years; all others shall be rejected.

Accepted applicants of 70 and over shall receive a warning that in the event of a claim they shall pay an excess of \$1000."

### Test Completion Criterion

"The Test Completion Criterion is that 100 % Equivalence Partition Coverage is achieved and all test cases must result in a "pass" status on execution."

### Test Conditions (TD2)

Based on the test completion criterion, the Test Conditions are the equivalence partitions for the described system behaviour.

Considering the inputs, the following valid partition is derived:

TCOND-1.  $18 \leq \text{Age} \leq 80$

Similarly, from the inputs, the following two invalid partitions are derived

TCOND-2. Age < 18.

TCOND-3. Age > 80.

Less obvious invalid input partitions could include other input types, such as non-integer inputs and non-numeric inputs. So, we could also generate the following invalid input equivalence partitions:

TCOND-4. Age = alphabetic

TCOND-5. Age = special character

Note that depending on the necessary rigour, further invalid (non-integer) partitions could be derived, such as numeric non-integers (e.g. 33.67 years). The set of potential invalid outputs is an infinitely large set.

Considering the valid (specified) outputs, the following equivalence partitions are identified:

TCOND-6.      Accept                                  (is induced by  $18 \leq \text{Age} \leq 80$ )

TCOND-7.	Reject	(is induced by (Age < 18) OR (Age > 80))
----------	--------	--

TCOND-8. Excess warning (is induced by  $70 \leq \text{Age} \leq 80$ )

An invalid output would be any output from the test item other than one of those specified. It can be challenging to identify unspecified outputs, however, they must be considered because if we can cause one to occur, then we have identified a defect in the test item, its test basis, or both. For this example just one unspecified output was identified and is shown below. Note that other testers may well derive quite different invalid outputs.

TCOND-9. Discount message (is induced by  $40 \leq \text{Age} \leq 55$ )

Note that it is also subjective how this (unspecified) message can be induced and the partition guessed here as the most likely to cause a discount message to be output may be completely different from that guessed by another tester.

#### Test Coverage Items (TD3)

Using Equivalence Partitioning (equivalence partitioning simply requires that each partition is exercised) then the following seven test coverage items are derived:

TCI-1.	$18 \leq \text{Age} \leq 80$	(covers TCOND-1/TCOND-6)
TCI-2.	$\text{Age} < 18$	(covers TCOND-2/TCOND-7)
TCI-3.	$\text{Age} > 80$	(covers TCOND-3/TCOND-7)
TCI-4.	$\text{Age} = w$	(covers TCOND-4)
TCI-5.	$\text{Age} = \&$	(covers TCOND-5)
TCI-6.	$70 \leq \text{Age} \leq 80$	(covers TCOND-8)
TCI-7.	$40 \leq \text{Age} \leq 55$	(covers TCOND-9)

#### Test Cases (TD4)

As long as we generate test cases that exercise each of the seven test coverage items then we will achieve 100% Equivalence Partition Coverage.

When generating test cases we can see that a single test case can sometimes exercise more than one test coverage item. There is an obvious benefit in minimizing the number of test cases as this reduces the test execution time, but this benefit can sometimes be counteracted by the extra time required to determine the minimal set and the potentially more complex debugging required when test cases target multiple test coverage items.

In this example two of the test cases exercise more than one test coverage item as shown below:

CASE#1.	Input: 'Age =53'	Expected Result: 'Accept'.	(exercises TCI-1 & TCI-7)
CASE#2.	Input: 'Age = 15'	Expected Result: 'Reject'.	(exercises TCI-2)
CASE#3.	Input: 'Age = 89'	Expected Result: 'Reject'.	(exercises TCI-3)
CASE#4.	Input: 'Age = w'	Expected Result: 'Reject'.	(exercises TCI-4)
CASE#5.	Input: 'Age =&'	Expected Result: 'Reject'.	(exercises TCI-5)
CASE#6.	Input: 'Age =77'	Expected Result: 'Accept+Warn'.	(exercises TCI-6 & TCI-1)

These six test cases will demonstrate that all the test coverage items were exercised, so achieving the test completion criterion.

#### Test Set (TD5)

If we assume that it is possible for test automation to handle integer inputs, but that non-integer inputs have to be handled manually, then we can generate two test sets; one for manual testing and one for automated testing.

TS1: CASES # 4 and 5	- manual testing.
TS2: CASES # 1, 2, 3, and 6	- automated testing.

## Annex B (normative)

### ISO/IEC/IEEE 29119-2 and ISO/IEC 12207:2008 Process Alignment

#### B.1 Overview

ISO/IEC 12207 provides a common framework of life cycle processes, a number of which include activities and tasks related to software testing. This normative annex explains at a high level how ISO/IEC/IEEE 29119-2 maps to the testing-related processes of ISO/IEC 12207.

The high level mapping below explains how users of ISO/IEC 12207 shall make use of ISO/IEC/IEEE 29119-2. For users of ISO/IEC/IEEE 29119-2 that do not use ISO/IEC 12207, this annex is optional.

A mapping of testing-related terms that are used in ISO/IEC 12207 is provided in ISO/IEC/IEEE 29119-1 Concepts and definitions.

#### B.2 ISO/IEC 12207:2008 to ISO/IEC/IEEE 29119-2 Mapping

The table below provides a mapping from the clauses, sub-clauses and tasks (column 1) of ISO/IEC 12207:2008, to corresponding processes of ISO/IEC/IEEE 29119-2 (column 2). The table includes a brief explanation of how the two sets of clauses map (column 3).

**Table B.1 — ISO/IEC 12207:2008 to ISO/IEC/IEEE 29119-2 high-level mapping**

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
<b>6.1.1 Acquisition Process</b>		
6.1.1.3 Activities and tasks 6.1.1.3.1 Acquisition preparation Task 6.1.1.3.1.2	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which provides a generic process for planning testing, which can include the identification and documentation of testing requirements.</i>
6.1.1.3 Activities and tasks 6.1.1.3.6 Acquirer acceptance Task 6.1.1.3.6	8 Dynamic Test Processes 8.2 Test Design & Implementation Process and 8.3 Test Environment Set-Up & Maintenance Process	<i>This is supported by 8.2 Test Design &amp; Implementation Process, which supports the preparation of test cases, test data and test procedures for acceptance testing. This is also supported by 8.3 Test Environment Set-Up &amp; Maintenance Process, which supports preparation of the test environment.</i>
6.1.1.3 Activities and tasks 6.1.1.3.6 Acquirer acceptance Task 6.1.1.3.6.2 (includes subclause 6.1.1.3.1.9)	8 Dynamic Test Processes 8.4 Test Execution Process	<i>This is supported by 8.4 Test Execution Process, which provides a generic process for executing test cases, including test cases for acceptance testing.</i>
<b>6.1.2 Supply Process</b>		
6.1.2.3 Activities and tasks 6.1.2.3.4 Contract execution Task 6.1.2.3.4.5	7 Test Management Processes 7.2 Test Planning Process	<i>For the planning of testing-related work, this clause is supported by clause 7 Test Management Processes, which provides activities for supporting test planning and test strategy development.</i>
6.1.2.3 Activities and tasks 6.1.2.3.4 Contract execution Task 6.1.2.3.4.8	7 Test Management Processes 7.3 Test Monitoring and Control Process	<i>This is supported by clause 7.3 Test Monitoring and Control Process, which supports monitoring testing progress and defect management process.</i>
6.1.2.3 Activities and tasks 6.1.2.3.4 Contract execution Task 6.1.2.3.4.10	<i>Not applicable</i>	<i>Not applicable, as ISO/IEC 29119-2 does not discuss interfaces between suppliers and testers.</i>



ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
6.1.2.3 Activities and tasks 6.1.2.3.4 Contract execution Task 6.1.2.3.4.15	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.4 Report (TMC4)	<i>This is supported by clause 7.3.4.4 Report (TMC4), which provides an ability to report testing progress and communicate new risks to stakeholders.</i>
	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.4 Report Test Completion (TC4)	<i>This is also supported by clause 7.4.4.4 Report Test Completion (TC4), which provides the ability to report the outcomes of testing to stakeholders.</i>
<b>6.2.1 Life Cycle Model Management Process</b>		
6.2.1.3 Activities and tasks 6.2.1.3.2 Process assessment Task 6.2.1.3.2.2	6 Organizational Test Process	<i>This is supported by clause 6 Organizational Test Process, which supports the periodic review of organizational test documentation, which can include the definition of processes for supporting testing.</i>
<b>6.3.1 Project Planning Process</b> (note: the mapping for this clause is only applicable for the planning of testing)		
6.3.1.3 Activities and tasks 6.3.1.3.2 Project planning Task 6.3.1.3.2.1	7 Test Management Processes 7.2 Test Planning Process	<i>The planning of testing is supported by clause 7.2 Test Planning Process, which provides generic processes for planning any phase or type of testing.</i>
<b>6.3.2 Project Assessment and Control Process</b> (note: the mapping for this clause is only applicable for the monitoring and control of testing)		
6.3.2.3 Activities and tasks 6.3.2.3.1 Project monitoring Task 6.3.2.3.1.1	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4.2 Monitor (TMC2)	<i>The monitoring of testing is supported by clause 7.3.4.2 Monitor (TMC2), which provides generic tasks for monitoring progress within any phase or type of testing.</i>
6.3.2.3 Activities and tasks 6.3.2.3.2 Project monitoring Task 6.3.2.3.2.1	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4.3 Control (TMC3)	<i>The management control of testing is supported by clause 7.3.4.3 Control (TMC3), which provides generic tasks for controlling any phase or type of testing.</i>
6.3.2.3 Activities and tasks 6.3.2.3.2 Project control Task 6.3.2.3.2.2	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4.4 Report (TMC4)	<i>The reporting of progress during testing is supported by clause 7.3.4.4 Report (TMC4), which provides generic tasks for reporting on the progress of any phase or type of testing.</i>
<b>6.3.4 Risk Management Process</b> (note: the mapping for this clause is only applicable for identification and Mitigation of risks that can be mitigated by testing)		
6.3.4.3 Activities and tasks 6.3.4.3.1 Risk management planning Task 6.3.4.3.1.1 Task 6.3.4.3.1.2	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4 Activities and tasks	<i>The definition of risk management processes for testing is supported by clause 6.2 Organizational Test Process, which can be used to define organizational processes for managing and documenting testing-related risks.</i>
6.3.4.3 Activities and tasks 6.3.4.3.1 Risk management planning Task 6.3.4.3.1.3 Task 6.3.4.3.1.4	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.2 Organize Test Plan Development (TP2)	<i>This is supported by the clause 7.2.4.2 Organize Test Plan Development (TP2), which includes the identification and notification of people who will be involved in risk management for testing-related risks.</i>
6.3.4.3 Activities and tasks 6.3.4.3.1 Risk management planning Task 6.3.4.3.1.5	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4 Activities and tasks 6.2.4.1 Develop Organizational Test Specification (OT1), 6.2.4.2 Monitor and Control use of Organizational Test Specification (OT2), and	<i>This is supported by clauses 6.2.4.1, 6.2.4.2 and 6.2.4.3 of the Organizational Test Process, which supports the monitoring, reviewing and updating of risk management processes that are used in testing.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
	6.2.4.3 Update Organizational Test Specification (OT3)	
6.3.4.3 Activities and tasks 6.3.4.3.2 Risk profile management Task 6.3.4.3.2.1 Task 6.3.4.3.2.2	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4 Activities and tasks	<i>Establishing a risk profile for is supported by clause 6 Organizational Test Process, which can be used to define processes, thresholds, and profiles for testing-related risks for a system, a project, a series or projects or an entire organization.</i>
6.3.4.3 Activities and tasks 6.3.4.3.2 Risk profile management Task 6.3.4.3.2.3	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4 Activities and tasks 6.2.4.1 Develop Organizational Test Specification (OT1)	<i>This is supported by clause 6.2.4.1 Develop Organizational Test Specification (OT1), which can include the definition of a risk profile for the organization.</i>
6.3.4.3 Activities and tasks 6.3.4.3.2 Risk profile management Task 6.3.4.3.2.4	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.4 Report (TMC4)	7.3.4.4 Report (TMC4) This activity consists of the following tasks: ... g) b) New risks and changes to existing risks shall be updated in the risk registry and communicated to the relevant stakeholders.
6.3.4.3 Activities and tasks 6.3.4.3.3 Risk analysis Task 6.3.4.3.3.1	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.3 Identify and Analyze Risks (TP3)	<i>This is supported by clause 7.2.4.3 Identify and Analyze Risks (TP3), which provides a generic process for identifying and analyzing testing-related risks.</i>
6.3.4.3 Activities and tasks 6.3.4.3.3 Risk analysis Task 6.3.4.3.3.2	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.3 Identify and Analyze Risks (TP3)	<i>This is supported by clause 7.2.4.3 Identify and Analyze Risks (TP3), which allows risks to be assigned a level of exposure (e.g. impact and likelihood).</i>
6.3.4.3 Activities and tasks 6.3.4.3.3 Risk analysis Task 6.3.4.3.3.3 Task 6.3.4.3.3.4	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.4 Identify Risk Mitigation Approaches (TP4)	<i>This is supported by clause 7.2.4.4 Identify Risk Mitigation Approaches (TP4), which enables the identification of risk mitigation approaches for testing.</i>
6.3.4.3 Activities and tasks 6.3.4.3.4 Risk mitigation Task 6.3.4.3.4.1 Task 6.3.4.3.4.2	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.4 Identify Risk Mitigation Approaches (TP4)	<i>This is supported by clause 7.2.4.4 Identify Risk Mitigation Approaches (TP4), which ensures the various risk mitigation approaches that are proposed are recorded in the draft test plan.</i>
	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.8 Gain Consensus on Test Plan (TP8)	<i>This is also supported by clause 7.2.4.8 Gain Consensus on Test Plan (TP8), which allows stakeholders to comment on the various risk mitigation approaches that are recorded in the draft test plan.</i>
6.3.4.3 Activities and tasks 6.3.4.3.4 Risk mitigation Task 6.3.4.3.4.3 Task 6.3.4.3.4.4 6.3.4.3.5 Risk monitoring Task 6.3.4.3.5.1 Task 6.3.4.3.5.3	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This is supported by clause 7.3.4.2 Monitor (TMC2), which ensures mitigation of testing-related risks are carried out and monitored, and ensures that new testing-related risks that are identified are recorded throughout testing.</i>
6.3.4.3 Activities and tasks 6.3.4.3.5 Risk monitoring Task 6.3.4.3.5.2	7 Test Management Processes 7.3 Test Monitoring and Control Process	<i>This is supported by clause 7.3.4.1 Set-Up (TMC1), which ensures suitable measures for monitoring risk mitigation are identified.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
	7.3.4 Activities and tasks 7.3.4.1 Set-Up (TMC1)	
6.3.4.3 Activities and tasks 6.3.4.3.6 Risk management process evaluation Task 6.3.4.3.6.1	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.3 Identify Lessons Learned (TC3)	<i>This is supported by clause 7.4.4.3 Identify Lessons Learned (TC3), which enables the identification of approaches to improve the risk management process for testing.</i>
6.3.4.3 Activities and tasks 6.3.4.3.6 Risk management process evaluation Task 6.3.4.3.6.2	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4.1 Develop Organizational Test Specification (OT1)	<i>This is supported by clause 6.2.4.1 Develop Organizational Test Specification (OT5), which can be used to review the effectiveness and efficiency of organizational testing processes defined in an organizational test specification.</i>
6.3.4.3 Activities and tasks 6.3.4.3.6 Risk management process evaluation Task 6.3.4.3.6.3	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.3 Identify Lessons Learned (TC3)	<i>This is supported by clause 7.4.4.3 Identify Lessons Learned (TC3), which allows organizational risks to be identified and communicated to the persons responsible for maintaining the organizational test specifications (e.g. to document testing-related risks that are applicable to the organization as a whole to be recorded in the organizational test strategy).</i>
<b>6.3.7 Measurement Process</b>		
6.3.7.3 Activities and tasks 6.3.7.3.1 Measurement planning Task 6.3.7.3.1.1 Task 6.3.7.3.1.2 Task 6.3.7.3.1.3 Task 6.3.7.3.1.4 Task 6.3.7.3.1.5	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.1 Set-Up (TMC1)	<i>This is supported by clause 7.3.4.1 Set-Up (TMC1), which enables the planning of measurement collection for testing.</i>
6.3.7.3 Activities and tasks 6.3.7.3.1 Measurement planning Task 6.3.7.3.1.6 Task 6.3.7.3.1.7	6 Organizational Test Process 6.2 Organizational Test Process 6.2.4 Activities and tasks 6.2.4.1 Develop Organizational Test Specification (OT1)	<i>This is supported by clause 6.2.4.1 Develop Organizational Test Specification (OT1), which supports recording and agreement of measurement processes for testing across all projects.</i>
	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.8 Gain Consensus on Test Plan (TP8)	<i>This is also supported by clause 7.2.4.8 Gain Consensus on Test Plan (TP8), which provides an ability to gain stakeholder approval of all changes to the planned test approach, including measurement activities and resources.</i>
	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.1 Set-Up (TMC1)	<i>This is also supported by clause 7.3.4.1 Set-Up (TMC1), which provides the ability to establish the collection of measures during testing.</i>
6.3.7.3 Activities and tasks 6.3.7.3.2 Measurement performance Task 6.3.7.3.2.1 Task 6.3.7.3.2.2 Task 6.3.7.3.2.3	7 Test Management Processes and 8 Dynamic Test Processes	<i>The generation of metrics is supported by clauses 8 Dynamic Test Processes. These clauses consist of processes that produce metrics such as number of test cases derived, passed and failed (dynamic testing metrics).  The process of analyzing these metrics is supported by clause 7.3.4.3 Control (TMC3) of the Test Management Processes, which enables analysis of collected metrics, in order to inform the control process whether testing is progressing as planned or whether changes to testing are required.  The process of producing information products that report on those metrics is supported by</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
		<i>clause 7.3.4.4 Report (TMC4) of the Test Monitoring and Control Process, and clause 7.4.4.4 Report Test Completion (TC4) of the Test Completion Process.</i>
6.3.7.3 Activities and tasks 6.3.7.3.2 Measurement performance Task 6.3.7.3.2.4	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.4 Report (TMC4)	<i>This is supported by clause 7.3.4.4 Report (TMC4), which enables the communication of measurement results in testing to relevant stakeholders.</i>
6.3.7.3 Activities and tasks 6.3.7.3.3 Measurement evaluation Task 6.3.7.3.3.1 Task 6.3.7.3.3.2	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.3 Identify Lessons Learned (TC3)	<i>This is supported by clause 7.4.4.3 Identify Lessons Learned (TC3), which facilitates the analysis and recording of improvements to measurement activities in testing, and to communicate them to relevant stakeholders.</i>
<b>6.4.2 System Requirements Analysis Process</b>		
6.4.2.3 Activities and tasks 6.4.2.3.1 Requirements specification Task 6.4.2.3.1.1	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which provides a generic process for planning testing, which can include the identification and documentation of testing requirements, which can cover qualification requirements.</i>
<b>6.4.5 System Integration Process</b>		
6.4.5.3 Activities and tasks 6.4.5.3.1 Integration Task 6.4.5.3.1.1	7 Test Management Processes	<i>For planning integration testing, this clause is supported by clause 7 Test Management Processes, which provides generic processes for planning testing, including integration testing.</i>
	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which provides a generic process for testing, which can be used for integration testing.</i>
6.4.5.3 Activities and tasks 6.4.5.3.2 Test readiness Task 6.4.5.3.2.1	8 Dynamic Test Processes 8.2 Test Design & Implementation Process	<i>This is supported by clause 7.2 Test Design &amp; Implementation Process, which provides a generic process for preparing test cases, test data and test procedures for any phase or type of testing, including system qualification testing.</i>
6.4.5.3 Activities and tasks 6.4.5.3.2 Test readiness Task 6.4.5.3.2.2	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This is supported by clause 7.3.4.2 Monitor (TMC2), which provides a generic process for determining aspects such as whether testing is complete and whether the testing process has been appropriate.</i>
<b>6.4.6 System Qualification Testing Process</b>		
6.4.6.3 Activities and tasks 6.4.6.3.1 Qualification testing Task 6.4.6.3.1.1	8 Dynamic Test Processes	<i>This clause can be supported clause 8 Dynamic Test Processes, depending on whether the qualification testing includes dynamic testing. Both are generic processes that can be used for any type of testing, including qualification testing.</i>
6.4.6.3 Activities and tasks 6.4.6.3.1 Qualification testing Task 6.4.6.3.1.2	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This is supported by clause 7.3.4.2 Monitor (TMC2), which provides a generic process for monitoring testing, which can be used to determine test coverage and determine whether expected and actual results of testing match.</i>
<b>6.4.7 Software Installation Process</b>		
6.4.7.3 Activities and tasks 6.4.7.3.1 Software installation Task 6.4.7.3.1.2	7 Test Management Processes	<i>For planning installation testing, this clause is supported by clause 7 Test Management Processes, which provides generic processes for planning testing, including installation testing.</i>



ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which includes generic processes for designing, implementing and executing test cases, and which could be used to determine whether installed software meets requirements specified in a contract.</i>
<b>6.4.8 Software Acceptance Support Process</b>		
6.4.8.3 Activities and tasks 6.4.8.3.1 Software acceptance support Task 6.4.8.3.1.1	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This is supported by clause 7.3.4.2 Monitor (TMC2), which could be used to determine whether a developer is supporting an acquirer's acceptance review and testing.</i>
<b>6.4.9 Software Operation Process</b>		
6.4.9.3 Activities and tasks 6.4.9.3.1 Preparation and Operation Task 6.4.9.3.1.3	7 Test Management Processes	<i>For planning testing of the software in its operational environment, this clause is supported by clause 7 Test Management Processes, which provides generic processes for planning testing, including operation testing.</i>
	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which defines generic test design and problem reporting processes that can be used for testing the software in the operation environment.</i>
6.4.9.3 Activities and tasks 6.4.9.3.2 Operation activation and check-out Task 6.4.9.3.2.1 Task 6.4.9.3.2.2	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which could be used for testing the software in the operation environment.</i>
6.4.9.3 Activities and tasks 6.4.9.3.3 Operational use Task 6.4.9.3.3.1	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which could be used for testing the software in the operation environment.</i>
6.4.9.3 Activities and tasks 6.4.9.3.3 Operational use Task 6.4.9.3.3.1	7 Test Management Processes 7.2.4 Activities and tasks 7.2.4.3 Identify and Analyze Risks (TP3)	<i>The identification of testing-related risks is supported by clause 7.2.4.3 Identify and Analyze Risks (TP3), which can include the identification of risks that can be treated by operational testing.</i>
	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>The monitoring of testing-related risks during operational testing is supported by clause 7.3.4.2 Monitor (TMC2), which provides generic tasks for monitoring progress within any phase or type of testing.</i>
6.4.9.3 Activities and tasks 6.4.9.3.3 Operational use Task 6.4.9.3.3.1	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>The monitoring of any testing cycle is supported by clause 7.3.4.2 Monitor (TMC2), which provides generic tasks for monitoring progress within any phase or type of testing.</i>
<b>6.4.10 Software Maintenance Process</b>		
6.4.10.3 Activities and tasks 6.4.10.3.1 Process implementation Task 6.4.10.3.1.1	7 Test Management Processes	<i>For planning testing-related activities, this clause is supported by clause 7 Test Management Processes, which provides generic processes for planning the activities and tasks involved in maintenance testing.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
6.4.10.3 Activities and tasks 6.4.10.3.3 Modification Implementation Task 6.4.10.3.3.2	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which defines a generic test design, implementation and execution process that can be used to demonstrate that changes and fixes as a result of maintenance have not compromised the ability of the software to meet its requirements.</i>
6.4.10.3 Activities and tasks 6.4.10.3.5 Migration Task 6.4.10.3.5.2 Task 6.4.10.3.5.3	7 Test Management Processes 7.2 Test Planning Process	<i>The testing-related planning elements of this clause can be supported by clause 7.2 Test Planning Process, which can be used to prepare and communicate test plans for migration verification testing.</i>
<b>7.1.1 Software Implementation Process</b>		
7.1.1.4 Activities and tasks 7.1.1.3.1 Software implementation strategy Task 7.1.1.3.1.4	7 Test Management Processes 7.2 Test Planning Process	<i>The testing-related planning elements of this clause can be supported by clause 7.2 Test Planning Process.</i>
<b>7.1.2 Software Requirements Analysis Process</b>		
7.1.2.3 Activities and tasks 7.1.2.3.1 Software requirements analysis Task 7.1.2.3.1.1	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which provides a generic process for planning testing, which can include the identification and documentation of testing requirements, including qualification requirements.</i>
7.1.2.3 Activities and tasks 7.1.2.3.1 Software requirements analysis Task 7.1.2.3.1.3	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This can be supported by clause 7.3.4.2 Monitor (TMC2), which could be used to determine whether an implementer is conducting reviews according to a specific process.</i>
<b>7.1.3 Software Architectural Design Process</b>		
7.1.3.3 Activities and tasks 7.1.3.3.1 Software architectural design Task 7.1.3.3.1.5	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which defines a generic process for test planning, which can be used to identify test requirements and schedules for software integration testing.</i>
7.1.3.3 Activities and tasks 7.1.3.3.1 Software architectural design Task 7.1.3.3.1.7	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>This can be supported by clause 7.3.4.2 Monitor (TMC2), which could be used to determine whether an implementer is conducting reviews according to a specific process.</i>
<b>7.1.4 Software Detailed Design Process</b>		
7.1.4.3 Activities and tasks 7.1.4.3.1 Software detailed design Task 7.1.4.3.1.5	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which defines a generic process for test planning, which can be used to identify test requirements and schedules for testing software units.</i>
7.1.4.3 Activities and tasks 7.1.4.3.1 Software detailed design. Task 7.1.4.3.1.6	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.3 Control (TMC3)	<i>This is supported by clause 7.3.4.3 Control (TMC3), which provides a generic process for making changes to the test approach, which can include updating test plans with required changes to schedules.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
<b>7.1.5 Software Construction Process</b>		
7.1.5.3 Activities and tasks 7.1.5.3.1 Software construction Task 7.1.5.3.1.1	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which includes generic processes for test design and implementation, and which can be used for designing test procedures and data for each software unit and database.</i>
7.1.5.3 Activities and tasks 7.1.5.3.1 Software construction Task 7.1.5.3.1.2	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which provide generic processes that can be used for any phase or type of testing.</i>
7.1.5.3 Activities and tasks 7.1.5.3.1 Software construction Task 7.1.5.3.1.4	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.3 Control (TMC3)	<i>This is supported by clause 7.3.4.3 Control (TMC3), which provides a generic process for making changes to the test approach, which can include updating test plans with required changes to schedules.</i>
7.1.5.3 Activities and tasks 7.1.5.3.1 Software construction Task 7.1.5.3.1.5	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>The monitoring of testing is supported by clause 7.3.4.2 Monitor (TMC2), which provides generic tasks for monitoring progress within any phase or type of testing, which can include monitoring the level of test coverage achieved.</i>
<b>7.1.6 Software Integration Process</b>		
7.1.6.3 Activities and tasks 7.1.6.3.1 Software integration Task 7.1.6.3.1.1	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which is a generic test planning process that can be used for defining a test plan for integration testing, including test requirements for integration testing.</i>
7.1.6.3 Activities and tasks 7.1.6.3.1 Software integration Task 7.1.6.3.1.2	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which includes generic processes for the design, documentation and execution of test cases, including integration test cases.</i>
7.1.6.3 Activities and tasks 7.1.6.3.1 Software integration Task 7.1.6.3.1.4	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which includes generic processes for the design and documentation of test cases, including qualification test cases and test procedures.</i>
7.1.6.3 Activities and tasks 7.1.6.3.1 Software integration Task 7.1.6.3.1.5	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.2 Monitor (TMC2)	<i>The monitoring of testing is supported by clause 7.3.4.2 Monitor (TMC2), which provides generic tasks for monitoring progress within any phase or type of testing, which can include monitoring the level of test coverage achieved.</i>
<b>7.1.7 Software Qualification Testing Process</b>		
7.1.7.3 Activities and tasks 7.1.7.3.1 Software qualification testing Task 7.1.7.3.1.1	8 Dynamic Test Processes	<i>This is also supported by clause 8 Dynamic Test Processes, which defines generic processes for carrying out test design, implementation and execution, which includes software qualification testing.</i>
<b>7.2.2 Software Configuration Management Process</b>		
7.2.2.3 Activities and tasks 7.2.2.3.3 Configuration control Task 7.2.2.3.3.1	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, as it defines generic dynamic testing processes that can be used for verification testing.</i>
<b>7.2.3 Software Quality Assurance Process</b>		
7.2.3.3 Activities and tasks 7.2.3.3.1 Process implementation Task 7.2.3.3.1.3	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which is a generic test planning process that can be used for defining a test plan for quality assurance activities that are related to testing.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
7.2.3.3 Activities and tasks 7.2.3.3.1 Process implementation Task 7.2.3.3.1.4	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>The initiation, management and control of any type of testing can be supported by clause 7.2 Test Monitoring and Control Process.</i>
7.2.3.3 Activities and tasks 7.2.3.3.1 Process implementation Task 7.2.3.3.1.5	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>Ensuring that records of any type of testing are made available to the acquirer can be supported by clause 7.2 Test Monitoring and Control Process.</i>
7.2.3.3 Activities and tasks 7.2.3.3.1 Process implementation Task 7.2.3.3.1.6	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>Ensuring that testing staff have freedom, resources and authority to permit the required testing can be supported by clause 7.2 Test Monitoring and Control Process.</i>
7.2.3.3 Activities and tasks 7.2.3.3.2 Product assurance Task 7.2.3.3.2.1	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>Ensuring that test plans are executed as required can be supported by clause 7.2 Test Monitoring and Control Process.</i>
7.2.3.3 Activities and tasks 7.2.3.3.2 Product assurance Task 7.2.3.3.2.3	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, which define generic dynamic testing processes that can be used for this type of assurance.</i>
<b>7.2.4 Software Verification Process</b>		
7.2.4.3 Activities and tasks 7.2.4.3.1 Process implementation Task 7.2.4.3.1.1 Task 7.2.4.3.1.2 Task 7.2.4.3.1.3	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which supports the identification of testing-related risks, consideration of the available testing budget and resources, and the selection of appropriate teams for carrying out the testing.</i>
7.2.4.3 Activities and tasks 7.2.4.3.1 Process implementation Task 7.2.4.3.1.3	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which can include steps for deciding on how to staff the required testing, particularly clause 7.2.4.6 Determine Staffing and Scheduling (TP6), to support the selection of appropriate staff for verification testing.</i>
7.2.4.3 Activities and tasks 7.2.4.3.1 Process implementation Task 7.2.4.3.1.4	7 Test Management Processes 7.2.4 Activities and tasks 7.2.4.5 Design Test Strategy (TP5)	<i>This is supported by clause 7.2.4.5 Design Test Strategy (TP5), which provides a generic process for designing a strategy for testing.</i>
7.2.4.3 Activities and tasks 7.2.4.3.1 Process implementation Task 7.2.4.3.1.5	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which includes generic activities for designing test plans, which can be used to design a test plan to suit the required verification effort.</i>
7.2.4.3 Activities and tasks 7.2.4.3.1 Process implementation Task 7.2.4.3.1.6	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>The initiation, management and control of verification testing can be supported by clause 7.2 Test Monitoring and Control Process.</i>
	8 Dynamic Test Processes	<i>If the verification test plan includes dynamic testing, then carrying out that testing this is supported by clause 8 Dynamic Test Processes, which provides generic processes for carrying out dynamic testing.</i>
7.2.4.3 Activities and tasks 7.2.4.3.2 Verification Task 7.2.4.3.2.3	8 Dynamic Test Processes	<i>This can be supported by clause 8 Dynamic Test Processes, which provides generic processes for carrying out dynamic testing, including checking for proper event sequence in the code.</i>
7.2.4.3 Activities and tasks 7.2.4.3.2 Verification Task 7.2.4.3.2.4	8 Dynamic Test Processes	<i>This can be supported by clause 8 Dynamic Test Processes, which provides generic processes for carrying out dynamic testing, including checking that each unit has been integrated correctly.</i>



ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
<b>7.2.5 Software Validation Process</b>		
7.2.5.3 Activities and tasks 7.2.5.3.1 Process implementation Task 7.2.5.3.1.1	7 Test Management Processes 7.2 Test Planning Process 7.2.4 Activities and tasks 7.2.4.3 Identify and Analyze Risks (TP3)	<i>This is supported by clause 7.2.4.3 Identify and Analyze Risks (TP3), which provides a generic process for risk identification.</i>
	7 Test Management Processes 7.2.4 Activities and tasks 7.2.4.4 Identify Risk Mitigation Approaches (TP4)	<i>This is also supported by clause 7.2.4.4 Identify Risk Mitigation Approaches (TP4), which provides a generic process for risk analysis, which can be used to determine phases and types of testing that are required to treat the identified risks.</i>
	7 Test Management Processes 7.2.3 Activities and tasks 7.2.4.5 Design Test Strategy (TP5)	<i>This is also supported by clause 7.2.4.5 Design Test Strategy (TP5), which provides a generic process for designing a test strategy that includes a determination of the level of independence and effort required during testing.</i>
7.2.5.3 Activities and tasks 7.2.5.3.1 Process implementation Task 7.2.5.3.1.2	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which includes generic activities for designing test plans, which can be used to design a test plan to suit the required validation effort.</i>
7.2.5.3 Activities and tasks 7.2.5.3.1 Process implementation Task 6.2.5.3.1.3	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which can include steps for deciding on how to staff the required testing, particularly clause 7.2.4.6 Determine Staffing and Scheduling (TP6), to support the selection of appropriate staff for validation testing.</i>
7.2.5.3 Activities and tasks 7.2.5.3.1 Process implementation Task 7.2.5.3.1.4	7 Test Management Processes 7.2 Test Planning Process	<i>This is supported by clause 7.2 Test Planning Process, which includes generic activities for designing test plans, which can be used to design a test plan to suit the required validation effort.</i>
7.2.5.3 Activities and tasks 7.2.5.3.1 Process implementation Task 7.2.5.3.1.5	7 Test Management Processes 7.2 Test Monitoring and Control Process	<i>The initiation, management and control of validation testing can be supported by clause 7.2 Test Monitoring and Control Process.</i>
	8 Dynamic Test Processes	<i>If the validation test plan includes dynamic testing, then carrying out that testing this is supported by clause 8 Dynamic Test Processes, which provides generic processes for carrying out dynamic testing.</i>
	8 Dynamic Test Processes 8.5 Test Incident Reporting Process	<i>This is supported by clause 8.5 Test Incident Reporting Process, which provides a generic process for reporting incidents that are detected during testing, including validation testing.</i>
7.2.5.3 Activities and tasks 7.2.5.3.2 Validation Task 7.2.5.3.2.1	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.4 Report Test Completion (TC4)	<i>This is also supported by clause 7.4.4.4 Report Test Completion (TC4), which provides a generic process for reporting on the results of testing to required stakeholders.</i>
	8 Dynamic Test Processes 8.2 Test Design & Implementation Process and 7 Test Management Process 7.2 Test Planning Process	<i>This is supported by 8.2 Test Design &amp; Implementation Process, which supports the preparation of test cases and test specifications for validation testing.</i>  <i>This is also supported by clause 7.2 Test Planning Process, which defines a generic process for test planning, which can be used to identify test requirements for validation testing, prior to the commencement of test case design.</i>

ISO/IEC 12207:2008 Clause	ISO/IEC/IEEE 29119-2 Clause	Mapping Explanation
7.2.5.3 Activities and tasks 7.2.5.3.2 Validation Task 7.2.5.3.2.3	8 Dynamic Test Processes 8.4 Test Execution Process	<i>This is supported by clause 8.4 Test Execution Process, which provides a generic process for executing test cases, which can include validation test cases.</i>
7.2.5.3 Activities and tasks 7.2.5.3.2 Validation Task 7.2.5.3.2.4	8 Dynamic Test Processes 8.4 Test Execution Process 8.4.4 Activities and tasks	<i>This is supported by clause 8.4 Test Execution Process, which provides a process for executing test cases, which can include validation test cases.</i>
7.2.5.3 Activities and tasks 7.2.5.3.2 Validation Task 7.2.5.3.2.5	8 Dynamic Test Processes 8.4 Test Execution Process	<i>This is supported by clause 8.4 Test Execution Process, which provides a generic process for executing test cases in any target environment.</i>
<b>7.2.6 Software Review Process</b>		
7.2.6.3 Activities and tasks 7.2.6.3.1 Process implementation Task 7.2.6.3.1.4	8 Dynamic Test Processes 8.5 Test Incident Reporting Process	<i>This is supported by clause 8.5 Test Incident Reporting Process, which provides a generic process for reporting incidents that are detected during testing.</i>
7.2.6.3 Activities and tasks 7.2.6.3.2 Project Management Reviews Task 7.2.6.3.2.1	7 Test Management Processes 7.3 Test Monitoring and Control Process	<i>This is supported by clause 7.3 Test Monitoring and Control Process, which supports monitoring testing progress throughout the project.</i>
<b>7.2.8 Software Problem Resolution Process</b>		
7.2.8.3 Activities and tasks 7.2.8.3.1 Process implementation Task 7.2.8.3.1.1	8 Dynamic Test Processes 8.5 Test Incident Reporting Process	<i>This is supported by clause 8.5 Test Incident Reporting Process, which provides a generic process for reporting incidents that are detected throughout testing.</i>
7.2.8.3 Activities and tasks 7.2.8.3.1 Process implementation Task 7.2.8.3.1.1	7 Test Management Processes 7.3 Test Monitoring and Control Process 7.3.4 Activities and tasks 7.3.4.4 Report (TMC4)	<i>This can be supported by clause 7.3.4.4 Report (TMC4), which can be used to report on analysis of defect trends throughout testing.</i>
	7 Test Management Processes 7.4 Test Completion Process 7.4.4 Activities and tasks 7.4.4.4 Report Test Completion (TC4)	<i>This can be supported by clause 7.4.4.4 Report Test Completion (TC4), which can be used to report on analysis of defect trends at the end of a testing phase or project.</i>
7.2.8.3 Activities and tasks 7.2.8.3.1 Process implementation Task 7.2.8.3.1.1	8 Dynamic Test Processes 8.5 Test Incident Reporting Process 8.5.4 Activities and tasks	<i>This is supported by clause 8.5 Test Incident Reporting Process, which provides a generic process for reporting incidents, which can be used for reporting and categorising problems.</i>
7.2.8.3 Activities and tasks 7.2.8.3.2 Problem resolution Task 7.2.8.3.2.1	8 Dynamic Test Processes 8.5 Test Incident Reporting Process	<i>This is supported by clause 8.5 Test Incident Reporting Process, which provides a generic process for reporting incidents that are detected throughout testing.</i>
<b>7.3.1 Domain Engineering Process</b>		
7.3.1.3 Activities and tasks 7.3.1.3.4 Asset provision Task 7.3.1.3.4.4	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, as it define generic processes for conducting dynamic testing that can be used for any form of evaluation.</i>
<b>7.3.3 Reuse Program Management Process</b>		
7.3.3.3 Activities and tasks 7.3.3.3.4 Planning Task 7.3.3.3.4.2	8 Dynamic Test Processes	<i>This is supported by clause 8 Dynamic Test Processes, as it defines generic processes for conducting dynamic testing that can be used for any form of evaluation.</i>

## Annex C (informative)

### ISO/IEC/IEEE 29119-2 and ISO/IEC 15288:2008 process alignment

This annex describes the alignment of the processes of ISO/IEC/IEEE 29119-2 and ISO/IEC 15288:2008. The alignment covers all testing-related processes in ISO/IEC 15288:2008.

**Table C.1 — ISO/IEC 15288:2008 to ISO/IEC/IEEE 29119-2 high-level mapping**

Area	ISO/IEC/IEEE 29119-2		ISO/IEC 15288:2008	
	Clause	Process or Activity	Clause	Process or Activity
<b>Organizational Test Process</b>	6.2	Organizational Test Process	6.2.1	Life Cycle Model Management Process
<b>Test Management Processes</b>	7.2	Test Planning Process	6.2.3	Project Portfolio Management Process
			6.3.1	Project Planning Process
			6.3.4	Risk Management Process
			6.4.6	Verification Process
			6.4.8	Validation Process
	7.3	Test Monitoring and Control Process	6.2.3	Project Portfolio Management Process
			6.3.2	Project Assessment and Control Process
			6.3.4	Risk Management Process
			6.3.7	Measurement Process
			6.4.6	Verification Process
			6.4.8	Validation Process
	7.4	Test Completion Process	6.2.3	Project Portfolio Management Process
			6.3.2	Project Assessment and Control Process
			6.3.4	Risk Management Process
			6.3.7	Measurement Process
<b>Dynamic Test Processes</b>	8.2	Test Design & Implementation Process	6.4.8	Validation Process
	8.3	Test Environment Set-Up & Maintenance Process	6.4.8	Validation Process
	8.4	Test Execution Process	6.4.5	Integration Process
			6.4.7	Transition Process
			6.4.8	Validation Process
	8.5	Test Incident Reporting Process	6.4.8	Validation Process
			6.4.10	Maintenance Process

## Annex D (informative)

### ISO/IEC/IEEE 29119-2 and ISO/IEC 17025:2005 process alignment

This annex describes the alignment of the processes of ISO/IEC/IEEE 29119-2 and ISO 17025:2005. The alignment covers all testing-related processes in ISO 17025:2005.

**Table D.1 — ISO 17025:2005 to ISO/IEC/IEEE 29119-2 high-level mapping**

ISO 17025:2005		ISO/IEC/IEEE 29119-2 DIS	
(Criticality of environmental conditions)	5.03.1	8.2.4.6 8.3.4.1	Derive Test Procedures (TD6) Establish Test Environment (ES1)
(Control of environmental conditions)	5.03.2	8.3.4.2	Maintain Test Environment (ES2)
General	5.04.1	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Selection of methods	5.04.2	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Laboratory-developed methods	5.04.3	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
(Provision of equipment)	5.05.01	Nil	Nil
(Demonstration of calibration)	5.05.02	8.3.4.1	Establish Test Environment (ES1)
(Identification of equipment)	5.05.04	8.2.4.6 8.3.4.1	Derive Test Procedures (TD6) Establish Test Environment (ES1)
(Equipment records)	5.05.05	8.3.4.2	Maintain Test Environment (ES2)
(Handling of mal-performing equipment)	5.05.07	8.3.4.2	Maintain Test Environment (ES2)
(Control of adjustments to equipment)	5.05.12	8.3.4.2	Maintain Test Environment (ES2)
(Identification of items)	5.08.2	8.2.4.3	Derive Test Coverage Items (TD3)
(Records of anomalies)	5.08.3	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
(Procedure - Monitoring the validity of tests)	5.09.1	7.3.4.2 7.3.4.3	Monitor (TMC2) Control (TMC3)

## Annex E (informative)

### ISO/IEC/IEEE 29119-2 and ISO/IEC 25051:2006 process alignment

This annex describes the alignment of the processes of ISO/IEC/IEEE 29119-2 and ISO/IEC 25051:2006. The alignment covers all testing-related processes in ISO/IEC 25051:2006.

**Table E.1 — ISO/IEC 25051:2006 to ISO/IEC/IEEE 29119-2 high-level mapping**

ISO/IEC 25051:2006		ISO/IEC/IEEE 29119-2 DIS	
Approach	6.2.1	7.2.4.1	Understand Context (TP1)
Pass/fail criteria	6.2.2	Nil	Nil
Test environment	6.2.3	7.2.4.5 7.3.4.1 8.2.4.6 8.3.4.1 8.4.4.1	Design Test Strategy (TP5) Set-Up (TMC1) Derive Test Procedures (TD6) Establish Test Environment (ES1) Execute Test Procedure(s) (TE1)
Schedule	6.2.4	7.2.4.6 7.2.4.7	Determine Staffing and Scheduling (TP6) Record Test Plan (TP7)
Test case description	6.3.1	8.2.4.4	Derive Test Cases (TD4)
Test procedures	6.3.2	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Execution report	6.4.1	8.4.4.1 8.4.4.3	Execute Test Procedure(s) (TE1) Record Test Execution (TE3)
Anomaly report	6.4.2	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
Assessment of the test results	6.4.3	Nil	Nil

## Annex F (informative)

### ISO/IEC/IEEE 29119-2 and BS 7925-2:1998 process alignment

This annex describes the alignment of the processes of ISO/IEC/IEEE 29119-2 and BS 7925-2:1998. The alignment covers all testing-related processes in BS 7925-2:1998.

**Table F.1 — BS 7925-2:1998 to ISO/IEC/IEEE 29119-2 high-level mapping**

BS 7925-2:1998		ISO/IEC/IEEE 29119-2 DIS	
General	4.1	7.2.4.5 7.3.4.1 8.2.4.1	Design Test Strategy (TP5) Set-Up (TMC1) Identify Feature Sets (TD1)
(Project component test strategy: specification)	4.2.1	7.2.4.5	Design Test Strategy (TP5)
Project component test planning	4.3	7.2.4.5 7.3.4.1	Design Test Strategy (TP5) Set-Up (TMC1)
Component test planning	4.4	7.2.4.5 7.3.4.1	Design Test Strategy (TP5) Set-Up (TMC1)
(Component test specification: Component test case design)	4.5.1	8.2.4.4	Derive Test Cases (TD4)
(Component test specification: Component test case requirements specification)	4.5.2	8.2.4.4	Derive Test Cases (TD4)
(Component test specification: Component test case execution repeatability)	4.5.3	Nil	Nil
Component test execution	4.6	Nil	Nil
(Component test recording: component identification)	4.7.1	8.4.4.1 8.4.4.3	Execute Test Procedure(s) (TE1) Record Test Execution (TE3)
(Component test recording: execution results)	4.7.2	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
(Component test recording: affected activities repeated)	4.7.3	Nil	Nil
(Component test recording: records)	4.7.4	Nil	Nil
Verification of component test completion	4.8	8.4.4.1	Execute Test Procedure(s) (TE1)

## Annex G (informative)

### ISO/IEC/IEEE 29119-2 and IEEE Std 1008-2008 process alignment

This annex describes the alignment of the processes of ISO/IEC/IEEE 29119-2 and IEEE 1008-2008. The alignment covers all testing-related processes in IEEE 1008-2008.

**Table G.1 — IEEE 1008-2008 to ISO/IEC/IEEE 29119-2 high-level mapping**

IEEE 1008-2008		ISO/IEC/IEEE 29119-2 DIS	
Specify a General Approach to Unit Testing	3.1.2.1	7.2.4.3 7.2.4.4 8.2.4.1	Identify and Analyze Risks (TP3) Identify Risk Treatment Approaches (TP4) Identify Feature Sets (TD1)
Specify Completeness Requirements	3.1.2.2	8.2.4.2	Derive Test Conditions (TD2)
Specify Termination Requirements	3.1.2.3	8.2.4.2	Derive Test Conditions (TD2)
Determine Resource Requirements	3.1.2.4	7.2.4.5	Design Test Strategy (TP5)
Specify a General Schedule	3.1.2.5	7.2.4.6 7.2.4.7	Determine Staffing and Scheduling (TP6) Record Test Plan (TP7)
Study the Functional Requirements	3.2.2.1	8.2.4.1	Identify Feature Sets (TD1)
Identify Additional Requirements and Associated Procedures	3.2.2.2	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Identify States of the Unit	3.2.2.3	8.2.4.4	Derive Test Cases (TD4)
Identify Input and Output Data Characteristics	3.2.2.4	8.2.4.2	Derive Test Conditions (TD2)
Select Elements to be Included in the Testing	3.2.2.5	8.2.4.1 8.2.4.2	Identify Feature Sets (TD1) Derive Test Conditions (TD2)
Refine the Approach	3.3.2.1	7.2.4.5 7.3.4.1 8.2.4.4 8.2.4.5 8.2.4.6	Design Test Strategy (TP5) Set-Up (TMC1) Derive Test Cases (TD4) Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Specify Special Resource Requirements.	3.3.2.2	7.2.4.2 7.2.4.5 7.2.4.6 8.2.4.1 8.2.4.2 8.3.4.1	Organize Test Plan Development (TP2) Design Test Strategy (TP5) Determine Staffing and Scheduling (TP6) Identify Feature Sets (TD1) Derive Test Conditions (TD2) Establish Test Environment (ES1)
Specify a Detailed Schedule	3.3.2.3	7.2.4.6 7.2.4.7	Determine Staffing and Scheduling (TP6) Record Test Plan (TP7)
Design the Architecture of the Test Set	3.4.2.1	8.2.4.3 8.2.4.4 8.2.4.5	Derive Test Coverage Items (TD3) Derive Test Cases (TD4) Assemble Test Sets (TD5)
Obtain Explicit Test Procedures as Required	3.4.2.2	8.2.4.5 8.2.4.6	Assemble Test Sets (TD5) Derive Test Procedures (TD6)
Obtain the Test Case Specifications	3.4.2.3	8.2.4.4	Derive Test Cases (TD4)



IEEE 1008-2008		ISO/IEC/IEEE 29119-2 DIS	
Augment, as Required, the Set of Test-Case Specifications Based on Design Information	3.4.2.4	8.2.4.4	Derive Test Cases (TD4)
Obtain and Verify Test Data	3.5.2.1	Nil	Nil
Obtain Special Resources	3.5.2.2	8.3.4.1	Establish Test Environment (ES1)
Obtain Test Items	3.5.2.3	Nil	Nil
Run Tests	3.6.2.1	8.3.4.1 8.4.4.1	Establish Test Environment (ES1) Execute Test Procedure(s) (TE1)
Determine Results	3.6.2.2	8.4.4.1 8.4.4.2 8.4.4.3 8.5.4.1	Execute Test Procedure(s) (TE1) Compare Test Results (TE2) Record Test Execution (TE3) Analyze Test Result(s) (IR1)
Case 1: A Fault in a Test Specification or Test Data	3.6.2.2.1	Nil	Nil
Case 2: A Fault in Test Procedure Execution	3.6.2.2.2	Nil	Nil
Case 3: A Fault in the Test Environment (for example, system software)	3.6.2.2.3	Nil	Nil
Case 4: A Fault in the Unit Implementation	3.6.2.2.4	Nil	Nil
Case 5: A Fault in the Unit Design	3.6.2.2.5	8.4.4.1 8.4.4.3	Execute Test Procedure(s) (TE1) Record Test Execution (TE3)
Check for Normal Termination of the Testing Process	3.7.2.1	Nil	Nil
Check for Abnormal Termination of the Testing Process	3.7.2.2	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
Supplement the Test Set	3.7.2.3	8.2.4.4	Derive Test Cases (TD4)
Describe Testing Status	3.8.2.1	Nil	Nil
Describe Unit's Status	3.8.2.2	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
Complete the Test Summary Report	3.8.2.3	8.4.4.2 8.5.4.1	Compare Test Results (TE2) Analyze Test Result(s) (IR1)
Ensure Preservation of Testing Products	3.8.2.4	7.4.4.1 7.4.4.3 7.4.4.4	Archive Test Assets (TC1) Identify Lessons Learned (TC3) Report Test Completion (TC4)



## Bibliography

The following documents were used during the development of this international standard.

- [1] BS 7925-2:1998, *Software testing — Software component testing*
- [2] BS 7925-1:1998, *Software testing — Vocabulary*
- [3] IEC 60300-3-9:1995, *Risk Analysis of technological systems*
- [4] IEEE Std 601.12-1990, *IEEE Standard Glossary of Software Engineering Terminology*
- [5] IEEE Std 829-2008, *IEEE Standard for Software and System Test Documentation*
- [6] IEEE Std 1008-1987, *IEEE Standard for Software Unit Testing*
- [7] IEEE Std 1012-2012, *IEEE Standard for System and Software Verification and Validation*
- [8] IEEE Std 1028-2008, *IEEE Standard for Software Reviews and Audits*
- [9] ISO 15489-1:2001, *Information and documentation — Records management — Part 1: General*
- [10] ISO 31000:2009, *Risk management — Principles and guidelines*
- [11] ISO 9001:2008, *Quality management systems — Requirements*
- [12] ISO/IEC 12207:2008, *Systems and software Engineering — Software life cycle processes*
- [13] ISO/IEC 16085:2006, *Systems and software Engineering — Life cycle Processes — Risk Management*
- [14] ISO/IEC 25000:2005, *Software Engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE*
- [15] ISO/IEC 25010:2011, *Systems and Software Engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*
- [16] ISO/IEC 25051:2006, *Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing*
- [17] ISO/IEC 90003:2004, *Software engineering — Guidelines for the application of ISO 9001:2000 to computer software*
- [18] ISO/IEC/IEEE 24765:2010, *Systems and software engineering — Vocabulary*
- [19] International Software Testing Qualifications Board (ISTQB), *Standard glossary of terms used in Software Testing* [online]. Updated 1 April 2010 [viewed 11 April 2011]. Available from: <http://www.istqb.org/>



## IEEE Notice to Users

**Notice and Disclaimer of Liability Concerning the Use of IEEE Documents:** IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon any IEEE Standard document.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied "AS IS."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE standard is subjected to review at least every ten years. When a document is more than ten years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

**Translations:** The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE should be considered the approved IEEE standard.

**Official Statements:** A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

**Comments on Standards:** Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important to ensure that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to comments or questions except in those cases where the matter has previously been addressed. Any person who would like to participate in evaluating comments or revisions to an IEEE standard is welcome to join the relevant IEEE working group at <http://standards.ieee.org/develop/wg/>.

Comments on standards should be submitted to the following address: Secretary, IEEE-SA Standards Board; 445 Hoes Lane; Piscataway, NJ 08854; USA

**Photocopies:** Authorization to photocopy portions of any individual standard for internal or personal use is granted by The Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

**Patents:** Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE-SA Website at <http://standards.ieee.org/about/sasb/patcom/patents.html>. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

**Participants:** The list of IEEE participants can be accessed at the following URL: [http://standards.ieee.org/downloads/29119/29119-2-2013/29119-2-2013\\_wg-participants.pdf](http://standards.ieee.org/downloads/29119/29119-2-2013/29119-2-2013_wg-participants.pdf).

**IMPORTANT NOTICE:** IEEE Standards documents are not intended to ensure safety, health, or environmental protection, or ensure against interference with or from other devices or networks. Implementers of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

*This IEEE document is made available for use subject to important notices and legal disclaimers. These notices and disclaimers appear in all publications containing this document and may be viewed at <http://standards.ieee.org/IPR/disclaimers.html>.*

**Abstract:** The purpose of the ISO/IEC/IEEE 29119 series of software testing standards is to define an internationally-agreed set of standards for software testing that can be used by any organization when performing any form of software testing. ISO/IEC/IEEE 29119-2 comprises test process descriptions that define the software testing processes at the organizational level, test management level and dynamic test levels. It supports dynamic testing, functional and non-functional testing, manual and automated testing, and scripted and unscripted testing. The processes defined in ISO/IEC/IEEE 29119-2 can be used in conjunction with any software development lifecycle model.

Since testing is a key approach to risk-mitigation in software development, ISO/IEC/IEEE 29119-2 follows a risk-based approach to testing. Risk-based testing is a common industry approach to strategizing and managing testing. Risk-based testing allows testing to be prioritized and focused on the most important features and functions.

**Keywords:** 29119, 29119-1, software testing, Test Planning Process, Test Plan, verification and validation

---

**ICS 35.080**

**ISBN 978-0-7381-8600-9**

Price based on 59 pages

© ISO/IEC 2013 – All rights reserved

© IEEE 2013 – All rights reserved