



DPC: 25/30507647 DC

BSI Group Headquarters

389 Chiswick High Road London W4 4AL

Tel: +44 (0)20 8996 9000

Fax: +44 (0)20 8996 7400

www.bsigroup.com

Date: 15 October 2025

Origin: International

Latest date for receipt of comments: 08 December 2025

Project No. 2025/00143

Responsible committee: IST/15 Software and systems engineering

Interested committees:

Title: Draft BS ISO/IEC/IEEE 29119-8 Software and systems engineering — Software testing. Part 8: Model-based testing.

Please notify admin.start@bsigroup.com if you are aware of any keywords that might assist in clarifying or identifying the standard or of the content of this standard:

- i) has any issues related to 3rd party IPR, patent or copyright
- ii) affects other national standard(s)
- iii) requires additional national guidance or information

WARNING: THIS IS A DRAFT AND MUST NOT BE REGARDED OR USED AS A BRITISH STANDARD. THIS DRAFT IS NOT CURRENT BEYOND 08 December 2025

This draft is issued to allow comments from interested parties; all comments will be given consideration prior to publication. No acknowledgement will normally be sent. **See overleaf for information on the submission of comments.**

No copying is allowed, in any form, without prior written permission from BSI except as permitted under the Copyright, Designs and Patent Act 1988 or for circulation within a nominating organization for briefing purposes. Electronic circulation is limited to dissemination by e-mail within such an organization by committee members.

Further copies of this draft may be purchased from BSI Shop <http://shop.bsigroup.com> or from BSI Customer Services, Tel: +44(0) 20 8996 9001 or email cservices@bsigroup.com. British, International and foreign standards are also available from BSI Customer Services.

Information on the co-operating organizations represented on the committees referenced above may be obtained from <https://standardsdevelopment.bsigroup.com/>.

E-mail: admin.start@bsigroup.com

Introduction

This draft standard is based on national and international discussions. Your comments on this draft are invited and will assist in the preparation of the consequent standard.

For international standards, comments will be reviewed by the relevant UK national committee before sending the consensus UK vote and comments to the international committee, which will then decide appropriate action. If the international standard is approved, it is usual for the text to be published as a British Standard.

For national standards, comments will be reviewed by the relevant UK national committee and the resulting standards published as a British Standard.

UK Vote

Please indicate whether you consider the UK should submit a negative (with supporting technical reasons) or positive vote on this draft. Please indicate if you are aware of any reason why this draft standard should not be published as a British Standard.

Submission of Comments

- **Annotated drafts are not acceptable and will be rejected.**
- All comments should be submitted online at <https://standardsdevelopment.bsigroup.com/>. You will need to register in order to comment.

Template for comments and secretariat observations

Date: xx/xx/20xx	Document: ISO/DIS xxxx
------------------	------------------------

1	2	(3)	4	5	(6)	7
MB	Clause No./ Subclause No./Annex (e.g. 3.1)	Paragraph/ Figure/Table/Note	Type of comment	Comment (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
	3.1	Definition 1	ed	Definition is ambiguous and needs clarifying.	Amend to read '...so that the mains connector to which no connection...'	
	6.4	Paragraph 2	te	The use of the UV photometer as an alternative cannot be supported as serious problems have been encountered in its use in the UK.	Delete reference to UV photometer.	



DRAFT International Standard

ISO/IEC/IEEE DIS 29119-8

Software and systems engineering — Software testing —

Part 8: Model-based testing

ICS: 35.080

ISO/IEC JTC 1/SC 7

Secretariat: **BIS**

Voting begins on:
2025-10-15

Voting terminates on:
2026-01-07

This document has not been edited by the ISO Central Secretariat.

THIS DOCUMENT IS A DRAFT CIRCULATED FOR COMMENTS AND APPROVAL. IT IS THEREFORE SUBJECT TO CHANGE AND MAY NOT BE REFERRED TO AS AN INTERNATIONAL STANDARD UNTIL PUBLISHED AS SUCH.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

Reference number
ISO/IEC/IEEE DIS 29119-8:2025(en)

Full document details: 25/30507647 DC

© ISO/IEC 2025
© IEEE 2025



Documents

Document Sharing Hub



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2025

© IEEE 2025

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO or IEEE at the respective address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Institute of Electrical and Electronics Engineers, Inc
3 Park Avenue, New York
NY 10016-5997, USA

Email: stds.ipr@ieee.org
Website: www.ieee.org

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Conformance	6
4.1 Intended usage	6
4.2 Full conformance	6
4.3 Tailored conformance	6
5 Introduction to MBT	6
5.1 MBT concepts	6
5.2 MBT modes in test case generation and execution	7
6 Implementing ISO/IEC/IEEE 29119-2 for MBT	9
7 MBT modifications of the ‘test strategy and planning’ process	10
7.1 Overview	10
7.2 Additional outcomes	11
7.3 Implementation of the design test strategy activity (TP5) for the MBT strategy	11
7.4 MBT strategy documentation	12
7.4.1 Overview	12
7.4.2 MBT strategy	12
8 MBT modifications of the ‘test design and implementation’ process	13
8.1 Overview	13
8.2 Implementation of the create test model (TD1) activity for MBT model creation	13
Annex A (Informative) MBT deployment with automated test execution	15
Annex B (Informative) Additional considerations for MBT	22
Annex C (Informative) Automated test case generation in MBT	32
Annex D (Informative) Examples of MBT	36
Annex E (Informative) Formal MBT	49
Annex F (Informative) Roles and responsibilities for the MBT tasks	53
Annex G (Informative) Mapping of MBT activities to ISO/IEC/IEEE 29119 standards	54
Bibliography	57

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO/IEC documents should be noted. This document was drafted in accordance with the rules given in the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

IEEE Standards documents are developed within IEEE Societies and subcommittees of IEEE Standards Association (IEEE SA) Board of Governors. IEEE develops its standards through an accredited consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE standards are documents developed by volunteers with scientific, academic, and industry-based expertise in technical working groups. Volunteers are not necessarily members of IEEE or IEEE SA and participate without compensation from IEEE. 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.

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and <https://patents.iec.ch>. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Technical Committee ISO/IEC JTC 1, Information technology, Subcommittee SC 7, Software and System Engineering, in cooperation with the Systems and Software Engineering Standards Committee of the IEEE Computer Society, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

A list of all parts in the ISO/IEC 29119 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

The purpose of the ISO/IEC/IEEE 29119 series is to define an internationally agreed set of standards for software testing that can be used by any organisation when performing any form of software testing and using any life cycle. This document explains how the ISO/IEC/IEEE 29119 series can be adopted to support model-based testing (MBT).

Models can form an abstract or a complete description of a system from a particular perspective, for example, software behaviour and design. Software testers can use this document to understand how models of expected behaviour can be used in test design and implementation to derive test cases, including test inputs and expected results.

The concepts relating to software testing defined in ISO/IEC/IEEE 29119-1 are also applicable to this document.

The test process model on which MBT is based is defined in ISO/IEC/IEEE 29119-2. It comprises test process descriptions that define the software testing processes at the organizational level, test management level and dynamic test level. Supporting diagrams describing the processes are also provided in ISO/IEC/IEEE 29119-2. The processes defined in ISO/IEC/IEEE 29119-2 are extended to support MBT.

The templates and examples of test documentation defined in ISO/IEC/IEEE 29119-3 also apply to this document.

Software test design techniques that can be used during test design are defined in ISO/IEC/IEEE 29119-4. Each test design technique defines a test model with syntax, semantics of the model and the way to derive the test case from the test model in ISO/IEC/IEEE 29119-4. MBT models and the associated MBT tools can be used to produce a test case that, in part or as a whole, satisfies the test techniques of ISO/IEC/IEEE 29119-4.

ISO/IEC/IEEE 29119-5 addresses using keywords to support testing. A MBT approach can generate keyword test cases as defined in ISO/IEC/IEEE 29119-5.

Software and systems engineering — Software testing —

Part 8: Model-based testing

1 Scope

This standard provides requirements and guidelines on applying model-based testing (MBT) following the test processes defined in ISO/IEC/IEEE 29119-2. This document covers the following areas:

- a) Definitions for MBT
- b) Implementing ISO/IEC/IEEE 29119-2 for MBT
- c) Implementing ISO/IEC/IEEE 29119-3 for MBT

Using MBT, the generation of a test case is systemized and automated. This standard assumes that test execution is also automated. The implementation of the generation algorithm is tool-dependent and, therefore, is out of the scope of this document. MBT tool selection is also out of the scope of this document.

The test models and the information generated by the supporting MBT tools can be used, in part or whole, to satisfy the documentation requirements of ISO/IEC/IEEE 29119-3. Many MBT tools implement one or more test techniques and associated measures defined in ISO/IEC/IEEE 29119-4. These techniques are associated with individual tool implementation and are outside the scope of this document.

This standard applies to MBT in all development lifecycle models.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes the requirements of this document. For dated references, only the edition cited applies. The latest edition of the referenced documents (including any amendments) applies for undated references.

ISO/IEC/IEEE 29119-2, *Software and systems engineering — Software testing — Part 2: Test processes*

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*

Other standards helpful in implementing and interpreting this document are listed in the Bibliography.

3 Terms and definitions

For this document, the following terms and definitions apply.

ISO, IEC and IEEE maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>
- IEEE Standards Dictionary Online: available at <https://ieeexplore.ieee.org/xpls/dictionary.jsp>

NOTE For additional terms and definitions in the field of systems and software engineering, see ISO/IEC/IEEE 24765,^[3] which is published periodically as a “snapshot” of the SEVOCAB (Systems and software engineering vocabulary) database and is publicly accessible at <https://www.computer.org/sevocab>.

3.1

actual results

set of behaviours or conditions of a *test item* (3.25), or set of conditions of associated data or the *test environment* (3.23), observed as a result of *test execution* (3.24)

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

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.1]

3.2

expected results

observable predicted behaviour of the *test item* (3.25) under specified conditions based on its specification or another source

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.4]

3.3

formal language-based testing

formal testing

testing (3.32) based on the integrated use of two or more formal languages

EXAMPLE Integrated use of a formal specification and a programming language for *test case* (3.19) generation.

3.4

formal MBT

MBT (3.7), in which the models are represented in a formal specification language

3.5

formal specification language

precise, mathematically based language used to describe the behaviour and requirements of a system

EXAMPLE B, VDM, Z^[4].

3.6

generation parameters

test selection criteria

parameters used by an *MBT tool* (3.10) to control the generation of *test cases* (3.19).

3.7

MBT

model-based testing

model-driven testing (MDT)

methodology to automatically generate and execute test cases from models of expected behaviour

3.8

MBT model

test model (3.27) used in *MBT* (3.7)

3.9

MBT test environment

test environment (3.23) that includes *MBT tools* (3.10)

3.10

MBT tool

software tool used to support *MBT* (3.7) activities

EXAMPLE 1 A software tool that supports modelling can be used for MBT model creation.

EXAMPLE 2 A software tool that supports *test case* (3.19) generation from the MBT model.

3.11**model element**

identifiable part of a model

EXAMPLE A single state in a state diagram.

3.12**model fidelity**

degree to which the *MBT* (3.7) model matches the precondition, the input and the expected behaviour of the *test item* (3.25)

3.13**modelling language**

notations and structures to describe models that have an explicit definition of the syntax

Note 1 to entry: Some modelling languages also have a definition of semantics.

3.14**offline MBT mode**

MBT (3.6) approach in which *test cases* (3.19) are generated and stored in a repository or the *MBT tool* (3.9) for execution later

3.15**online MBT mode**

MBT (3.7) approach in which *test cases* (3.19) are created and immediately executed before the next *test case* is generated

3.16**performance testing**

type of *testing* (3.32) conducted to evaluate the degree to which a *test item* (3.25) accomplishes its designated functions within given constraints of time and other resources

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.11]

3.17**security testing**

test type (3.30) conducted to evaluate the degree to which a *test item* (3.25) 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

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.19]

3.18**test**

activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.21]

3.19**test case**

set of preconditions, inputs and *expected results* (3.2), developed to drive the execution of a *test item* (3.25) to meet *test objectives* (3.28)

Note 1 to entry: A *test case* (3.19) is the lowest level of test implementation documentation (i.e. test cases are not made up of test cases) for the *test level* (3.26) or *test type* (3.30) for which it is intended.

Note 2 to entry: Test case preconditions include the required state of the *test environment* (3.23), data (e.g. databases) used by the *test item*, and the test item itself.

Note 3 to entry: Inputs are the data information and actions, where applicable, used to drive *test execution* (3.24).

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.23]

3.20**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* (3.32)

Note 1 to entry: ISO/IEC/IEEE 29119 (all parts) does not use the concept of test conditions but instead uses the concept of a *test model* (3.26) for test design.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.27]

3.21**test coverage**

degree, expressed as a percentage, to which specified *test coverage items* (3.22) have been exercised by a *test case* (3.19) or test cases

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.28]

3.22**test coverage item**

measurable attribute of a *test item* (3.25) that is the focus of *testing* (3.32)

EXAMPLE Equivalence partitions, transitions between states, executable statements.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.29]

3.23**test environment**

environment containing facilities, hardware, software, firmware, and procedures needed to conduct a *test* (3.18)

Note 1 to entry: A test environment can contain multiple environments to accommodate specific *test levels* (3.26) or types (e.g. a unit test environment, a performance test environment).

Note 2 to entry: A test environment can comprise several interconnected systems or virtual environments.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.34]

3.24**test execution**

process of running a *test* (3.18) on the *test item* (3.25), producing *actual results* (3.1)

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.38]

3.25**test item**

test object

work product to be tested

EXAMPLE Software component, system, requirements document, design specification, user guide.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.42]

3.26**test level**

one of a sequence of test stages, each of which is typically associated with the achievement of particular objectives and used to treat particular risks

EXAMPLE The following are common test levels, listed sequentially: unit/component testing, integration testing, system testing, system integration testing, and acceptance testing.

Note 1 to entry: It is not always necessary for a *test item* (3.25) to be tested at all test levels, but the sequence of test levels generally stays the same.

Note 2 to entry: Typical objectives can include consideration of basic functionality for unit/component testing, interaction between integrated components for integration testing, acceptability to end users for acceptance testing

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.43]

3.27

test model

representation of the *test item* ([3.25](#)), which allows the testing to be focused on particular characteristics or qualities

EXAMPLE Requirements statements, equivalence partitions, state transition diagram, use case description, decision table, input syntax, source code, control flow graph, parameters and values, classification tree, natural language.

Note 1 to entry: The test model and the required *test coverage* ([3.21](#)) are used to identify *test coverage items* ([3.22](#)).

Note 2 to entry: A separate test model can be required for each type of required *test coverage* included in the test completion criteria.

Note 3 to entry: A test model can include one or more *test conditions* ([3.20](#)).

Note 4 to entry: Test models are commonly used to support test design (e.g. they are used to support test design in ISO/IEC/IEEE 29119-4, and they are used in *MBT* ([3.7](#)). Other types of models exist to support other aspects of testing, such as test environment models, test maturity models and test architecture models.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.46]

3.28

test objective

reason for performing *testing* ([3.32](#))

EXAMPLE Checking for correct implementation, identification of defects, measuring quality.

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.49]

3.29

test oracle

source of information for determining whether a test has passed or failed

Note 1 to entry: The test oracle is often a specification used to generate *expected results* ([3.2](#)) for individual *test cases* ([3.19](#)), but other sources may be used, such as comparing *actual results* ([3.1](#)) with those of another similar program or system or asking a human expert.

[SOURCE: ISO/IEC/IEEE 29119-1:2022, 3.115]

3.30

test type

testing ([3.32](#)) that is focused on specific quality characteristics

EXAMPLE *Security testing* ([3.17](#)), functional testing, usability testing, and *performance testing* ([3.16](#)).

Note 1 to entry: A test type can be performed at a single *test level* ([3.26](#)) or across several test levels (e.g., performance testing performed at unit and system test levels).

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.60]

3.31

testware

artefacts produced during the test process required to plan, design, and execute tests

Note 1 to entry: Testware can include such things as documentation, scripts, inputs, expected results, files, databases, environment, and any additional software or utilities used in the course of testing.

[SOURCE: ISO/IEC/IEEE 29119-1:2022, 3.132]

3.32 testing

set of activities conducted to facilitate the discovery and/or evaluation of properties of one or more *test items* ([3.25](#))

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

[SOURCE: ISO/IEC/IEEE 29119-2:2021, 3.61]

3.33 traceability information

records of the relationship between products of the development and test process

Note 1 to entry: Products of the development and test process include requirements, quality attributes, *MBT models* ([3.8](#)) and *test cases* ([3.19](#)).

4 Conformance

4.1 Intended usage

The requirements in this document are contained in [Clause 6](#), [Clause 7](#) and [Clause 8](#).

This document provides requirements and recommendations for MBT. It is recognized that particular projects or organizations may not need to use all the activities or tasks described in this document. Therefore, implementation of this document typically involves selecting a set of activities suitable for the organization or project.

There are two ways that an organization or tool vendor can claim to conform to the provisions of this part of ISO/IEC/IEEE 29119 – full conformance or tailored conformance.

The organization or tool vendor shall assert whether it claims full or tailored conformance to this part of ISO/IEC/IEEE 29119.

4.2 Full conformance

Full conformance is achieved by providing evidence that all of the requirements of the activities defined in this document have been satisfied.

4.3 Tailored conformance

When this document is used to establish MBT practices that do not qualify for full conformance, the subset of requirements defined in [Clauses 7](#) and [8](#) for which conformance is claimed should be recorded. Tailored conformance is achieved by demonstrating that the recorded subset of requirements has been satisfied.

Where tailoring occurs, the justification shall be provided, either directly or by reference, whenever a requirement defined in [Clauses 7](#) or [8](#) is not followed.

5 Introduction to MBT

5.1 MBT concepts

MBT uses models to systematically and automatically generate test cases. The model is amenable to at least partial automatic processing to generate testware (especially test cases). With systematic and automatic test case generation, MBT can achieve improved test coverage and reduce the effort and duration of software testing. In addition to the process extensions, several factors related to the deployment of MBT are considered. The factors include MBT tool selection, abstraction level definition, traceability requirement, model fidelity and MBT style guide. Refer to [Annex B](#) for detail explanations of the factors.

There are various approaches to using a model to support testing. These approaches can generate test inputs (partial), test oracles (partial), or generate complete test cases. This document focuses on the approach of generating complete test cases.

MBT generates and executes test cases automatically. The implementation of the generation algorithm is tool-dependent and out of the scope of this standard. The requirement or capability of the generation algorithm is common. [Annex C](#) lists and explains examples of test case generation requirements.

Models can be developed for a complete software system or a part of it. Models are defined with formal or semi-formal languages to represent different aspects of a test item. Generally, the more formal the notation used to construct a model, the more precise testware can be generated. Refer to [Annex E](#) for a description of applying MBT using formal specification languages to build models (practice known as formal MBT) in the context of formal language-based testing (formal testing).

It is optimal if the testers create an MBT model independently. By independently developing an MBT model, missing information or defects in the development model are less likely to be duplicated in the test artefacts. When development models^[6] are used as the basis for MBT,^[7] it is typically to save time and money while accepting the associated risks. The possible roles and responsibilities are explained in [Annex F](#).

An MBT model describes aspects of behaviour. Behaviours modelled include desired functionality and desired quality attributes, such as reliability.

Example notations used for behavioural descriptions in MBT include:

- Unified Modelling Language (UML) sequence diagram;
- UML state diagram;
- Business Process Model and Notation (BPMN) diagram.

MBT models are usually developed using graphical, tabular, and/or textual information to support test automation. Since MBT models are abstractions, to be executed, they are transformed into another representation, such as test procedures, scripts, or keyword-based test cases.^[5] An MBT tool transforms abstract models into detailed test cases or procedures. Once placed in the proper tool environment, the test case generated from the model can support automated test execution.

5.2 MBT modes in test case generation and execution

MBT has two basic modes that integrate with automated test execution: online MBT mode and offline MBT mode.

[Figure 1](#) shows the MBT activities in online MBT mode.

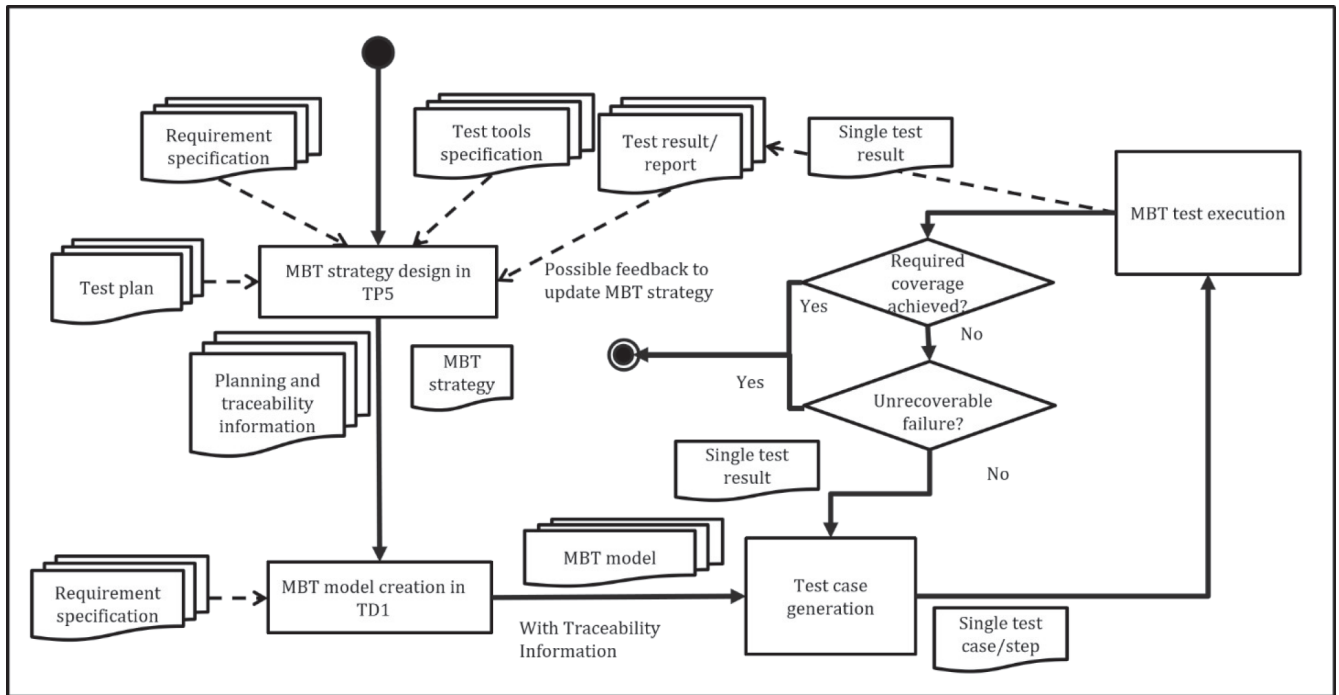


Figure 1 — MBT activities in online MBT mode

In online MBT mode, an MBT tool can connect directly to the test item and test it dynamically. The generation algorithm in the test case generation activity can generate one single test case or test step at a time based on the MBT model and the required test coverage in the MBT strategy. The test case can then be automatically executed in a dynamic test execution environment. The test result is then fed back into the generation algorithm. The generation algorithm can use the test result of the former test cases to optimise the subsequent test case generation.

There can be four types of feedback in online MBT:

- Feedback on required test coverage
- Feedback on test coverage item
- Feedback on the test case
- Feedback on the test procedure

EXAMPLE 1 When an executed test case generates a failure result, the test case generation algorithm can stop generating further test cases associated with that test case or failure. It then can potentially generate different test cases which do not fail. A test item can crash during testing after a sequence of test steps, and then when the test item exercises other test cases with a similar sequence, a crash can happen again. Thus, this can cause inefficiencies or harm the hardware if a system crash occurs frequently. In some embedded systems, the CPU software controls the download of files to the FPGA (Field Programmable Gate Array), and if the control logic is wrong, then an incorrect file written to the FPGA can reduce the lifetime of the FPGA. So, if the failure happens again and again, then the FPGA can possibly be damaged.

EXAMPLE 2 When the required level of test coverage is achieved, the algorithm in the test case generation step chooses to stop generating other test cases.

[Figure 2](#) shows the MBT activities in offline MBT mode.

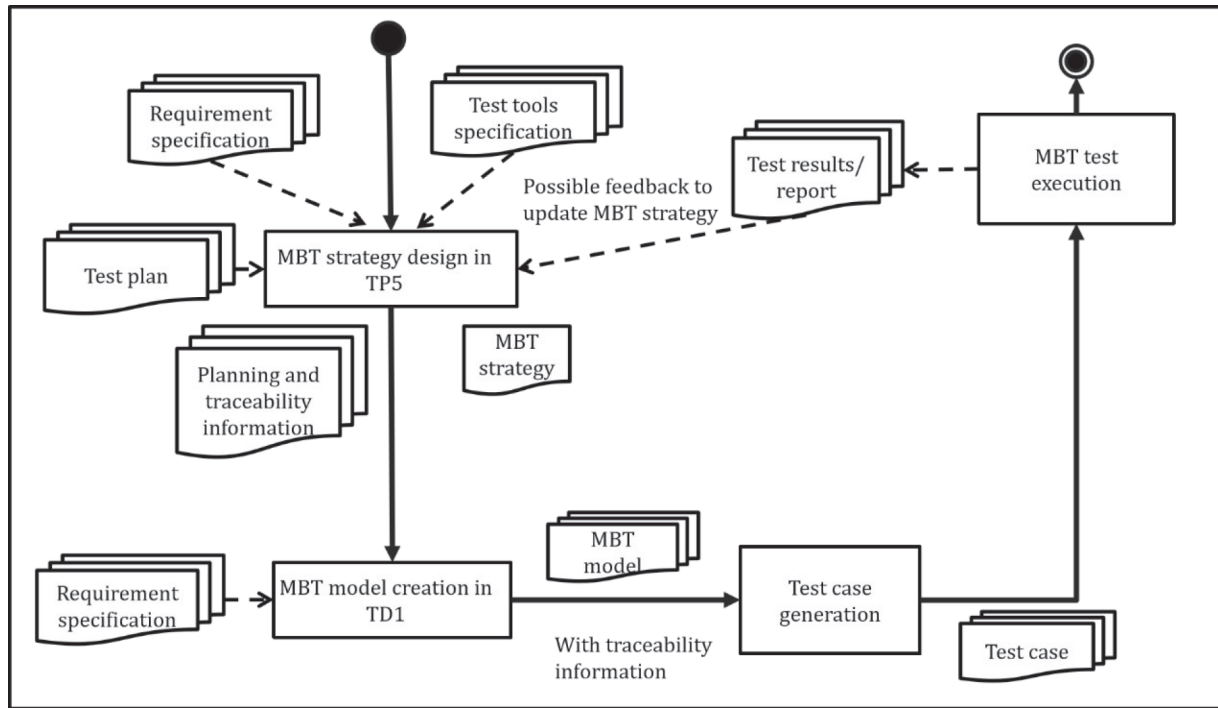


Figure 2 — MBT tasks in offline MBT mode

In offline MBT mode, the generation algorithm can generate a set of test cases in the test case generation activity. These test cases can be optionally reviewed, typically determining if the test cases provide the required coverage of the MBT model. When this is not achieved (and it appears possible to improve MBT model coverage), the generation parameters can be modified, and a new set of tests can be generated. In some situations, the generated tests can also be approved at this stage. Test cases generated in offline MBT mode can be executed automatically by a test automation system when the test cases are machine-executable. The manual execution of a generated test case in offline MBT mode can also be possible when the test cases are human-readable.

The MBT mode can be chosen based on the following:

- MBT tool and test execution tool capability;
- The maturity of the test item.
- Test objectives

NOTE 1 The online MBT mode can be preferred when the maturity of the test item is low and supported by the tool. Also, when the maturity of the test item is low, the test cases can fail frequently. In online MBT mode, the tool can skip testing a failed function and generate test cases for other functions. This action is done so that test execution time is not wasted by continually running a low-maturity function. Both online and offline MBT modes may be applied when the system's maturity is high.

NOTE 2 When regression testing is planned, offline MBT mode is likely to be preferred.

6 Implementing ISO/IEC/IEEE 29119-2 for MBT

MBT shall be performed in accordance with the processes defined in ISO/IEC/IEEE 29119-2. The two processes, “Test strategy and planning process” and “Test design & implementation process”, are modified for application in MBT, as highlighted in [Figure 3](#).

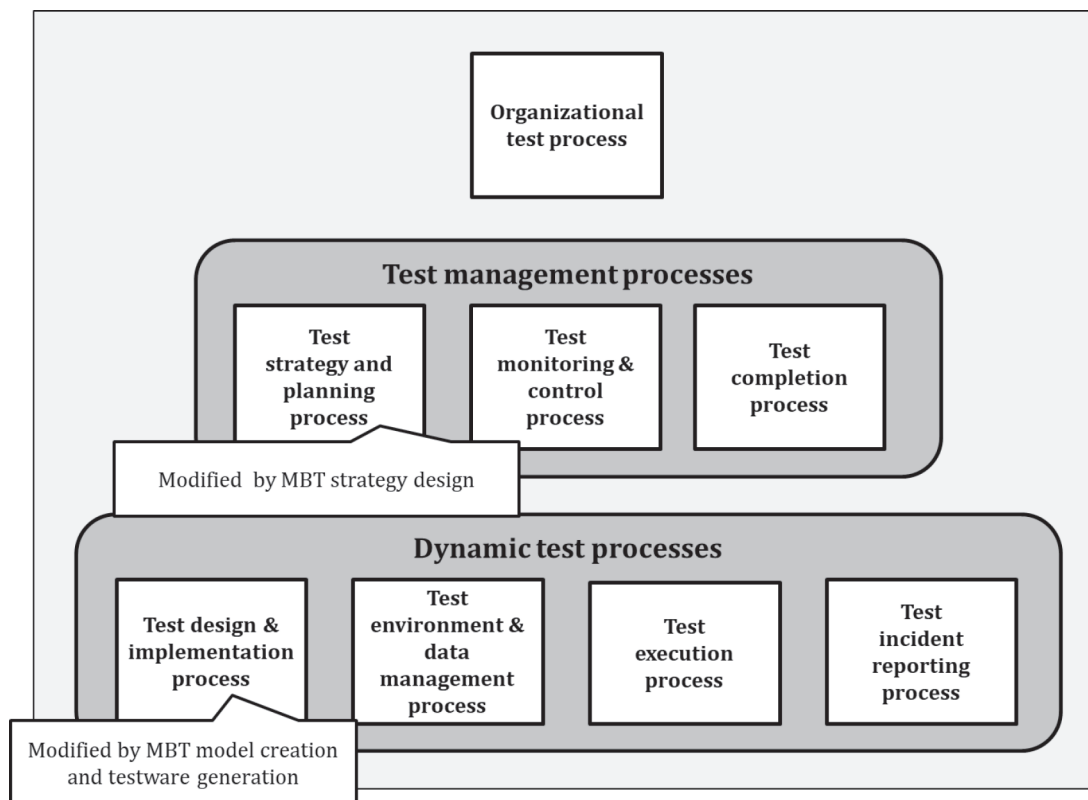


Figure 3 — MBT modifications in the ISO/IEC/IEEE 29119 processes model

For detailed modifications of ‘test strategy and planning process’, refer to [7.3](#). ISO/IEC/IEEE 29119-2, ISO/IEC/IEEE 29119-3 and ISO/IEC/IEEE 29119-4 explicitly describe the use of test models. By using these test models, a test case can be derived. In MBT, the test case can be derived automatically using test case generation algorithms, which follow the steps of ISO/IEC/IEEE 29119-2. These generation algorithms can create test cases more efficiently and effectively than manual effort. For detailed modifications of ‘test design and implementation process’, refer to [8.2](#).

Information can be automatically extracted from the MBT model for inclusion in test documentation, facilitating full or partial conformance with ISO/IEC/IEEE 29119-3 requirements.

7 MBT modifications of the ‘test strategy and planning’ process

7.1 Overview

The test strategy and planning process defined in ISO/IEC/IEEE 29119-2 shall be applied with the ‘design test strategy’ (TP5) activity adjusted as described in [7.3](#). In TP5, MBT focuses on the design of an MBT test strategy. The MBT strategy defines the MBT scope, the required test coverage, the MBT modelling language, model abstraction levels, the MBT style guide, the MBT environment, and MBT metrics.

The overview and purpose of the test strategy and planning process remain unchanged.

The remaining activities, ‘understand context’ (TP1), ‘organize test plan development’ (TP2), ‘identify and analyse risks’ (TP3), ‘identify risk treatment approaches’ (TP4), ‘determine staffing and scheduling’ (TP6), ‘record test plan’ (TP7), ‘gain consensus on test plan’ (TP8) and ‘communicate test plan and make available’ (TP9) shall be applied in accordance with ISO/IEC/IEEE 29119-2.

The test plan documentation defined in ISO/IEC/IEEE 29119-3 applies to MBT. This clause described the extensions to the documentation by MBT in detail.

7.2 Additional outcomes

In addition to the outcomes defined in ISO/IEC/IEEE 29119-2 7.2.3, the MBT test strategy shall be identified by extended TP5 activity.

7.3 Implementation of the design test strategy activity (TP5) for the MBT strategy

This activity consists of the following tasks:

- a) A test strategy shall be designed that considers the test basis, risks, organizational, project, and product constraints.

NOTE 1 This takes into consideration the results of the risk assessment to prioritise the test activities and determine the resources needed to perform actions (e.g. time, costs, skills, tool support and environment needs) while meeting organizational, project and product constraints, such as:

- regulatory standards;
- the requirements of the Organizational test policy and Organizational test practices;
- higher-level test plans and strategies for the project;
- contractual requirements;
- availability of appropriately skilled testers;
- availability of tools and environments.

Where it is not possible to design a test strategy that implements all the requirements of the organizational test practices 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 varies depending on the project and organization. It can require the constraints to be relaxed and the identification of risk treatment approaches activity (TP4). This task is to be repeated until an acceptable test strategy is achieved. Where it is decided to deviate from the organizational test practices, this should be recorded in the test strategy.

NOTE 2 The format of a test strategy is defined in ISO/IEC/IEEE 29119-3. It includes decisions on test levels, test types, aspects to be tested, test design techniques, test completion criteria, and suspension and resumption criteria.

NOTE 3 A test strategy would typically address static testing (e.g. reviews, inspections, static analysis) and dynamic testing.

- b) An MBT strategy shall be designed when MBT is planned and applicable that considers the MBT scope, test completion criteria, modelling language, abstraction level and the MBT style guide

NOTE 4 This new task has been added to the original activity defined in ISO/IEC/IEEE 29119-2 to guide MBT activities in dynamic test processes.

- c) Activities required to implement the test strategy shall be identified.
- d) Metrics for Test Monitoring and Control (see activities TMC1 to TMC4) shall be identified.

NOTE 5 MBT metrics are identified in task d) also. Refer to [Annex B](#) for an example of MBT metrics.

- e) Test data requirements shall be identified.

EXAMPLE Factors to consider when identifying test data requirements include regulations on data confidentiality (it can 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. The MBT online or offline mode for test case generation and execution shall be chosen. (refer to Annex [A.3](#) for an example of MBT mode)

NOTE 6 This modified task has been added to the original activity defined in ISO/IEC/IEEE 29119-2 to define the MBT test case generation and execution mode.

- 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 activities required to implement the test strategy shall be produced.

NOTE 7 The initial test estimate produced in this step is finalised in the record test plan activity (TP7).

- i) The test strategy shall be recorded.

NOTE 8 The test strategy will typically be a section of the Test Plan, but in some cases, it can be recorded as a separate document.

- j) The stakeholders shall obtain approval for the test strategy.

NOTE 9 This can require repeating earlier tasks in this activity.

7.4 MBT strategy documentation

7.4.1 Overview

MBT shall be documented in accordance with the requirements defined in ISO/IEC/IEEE 29119-3. In addition to extending ISO/IEC/IEEE 29119-2 test processes, the test strategy documentation defined in ISO/IEC/IEEE 29119-3:2021, 7.2.7 is extended to support MBT.

7.4.2 MBT strategy

7.4.2.1 MBT scope

For a project test plan or a level test plan, this records the scope of MBT. MBT scope can be defined as features, requirements, and quality attributes to be tested. An example of MBT's scope can be found in Annex [D.2.2](#) and Annex [D.3.3](#).

7.4.2.2 Required test coverage

The MBT strategy identifies and records the required test coverage derived from test completion criteria. An example of required test coverage can be found in Annex [D.2.2](#).

Note if the required test coverage is defined in the test strategy, the MBT strategy refers to the test strategy.

7.4.2.3 Modelling language

The MBT strategy records the selected, tailored, extended modelling language(s).

7.4.2.4 Model abstraction level

The MBT strategy records the decided abstraction level of modelling. Refer to Annex [B.3](#) for a practical approach to abstraction level selection.

7.4.2.5 MBT style guide

The MBT strategy records the MBT style guide, which includes rules to follow in modelling. Refer to Annex [B.6](#) for a detailed example of the MBT style guide and a recommended template of the MBT style guide.

7.4.2.6 MBT metrics

The MBT metrics identify data to be collected during the MBT activities (refer to Annex [B.7](#)).