

# **Standard Glossary of Terms Used in Software Testing**

## **Version 3.0**

### **Terms Used in the Advanced Level - Technical Test Analyst Syllabus**

---

International Software Testing Qualifications Board

---



Copyright © International Software Testing Qualifications Board (hereinafter called ISTQB®).

This document may be copied in its entirety, or extracts made, if the source is acknowledged.

Edited by:

ISTQB Glossary Working Group, Judy McKay (Chair), Matthias Hamburg (Vice-Chair), 2015.

This document was formally released by the General Assembly of the ISTQB on March 26<sup>th</sup>, 2015.

# ISTQB Glossary Report - Advanced Technical Test Analyst

## **acceptance testing**

**Ref:** After IEEE 610    **See Also:** user acceptance testing

Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

---

## **accuracy**

**Ref:** ISO 9126    **See Also:** functionality

The capability of the software product to provide the right or agreed results or effects with the needed degree of precision.

---

## **adaptability**

**Ref:** ISO 9126    **See Also:** portability

The capability of the software product to be adapted for different specified environments without applying actions or means other than those provided for this purpose for the software considered.

---

## **analytical testing**

Testing based on a systematic analysis of e.g., product risks or requirements.

---

## **analyzability**

**Ref:** ISO 9126    **See Also:** maintainability

The capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified to be identified.

---

## **anti-pattern**

Repeated action, process, structure or reusable solution that initially appears to be beneficial and is commonly used but is ineffective and/or counterproductive in practice.

---

## **atomic condition**

A condition that cannot be decomposed, i.e., a condition that does not contain two or more single conditions joined by a logical operator (AND, OR, XOR).

---

## **attractiveness**

**Ref:** ISO 9126    **See Also:** usability

The capability of the software product to be attractive to the user.

---

## **availability**

**Ref:** IEEE 610

The degree to which a component or system is operational and accessible when required for use. Often expressed as a percentage.

---

## **black-box testing**

**Synonyms:** specification-based testing

Testing, either functional or non-functional, without reference to the internal structure of the component or system.

---

## **branch**

A basic block that can be selected for execution based on a program construct in which one of two or more alternative program paths is available, e.g., case, jump, go to, if-then-else.

---

## **branch coverage**

The percentage of branches that have been exercised by a test suite. 100% branch coverage implies both 100% decision coverage and 100% statement coverage.

---

## **buffer**

**Ref:** IEEE 610

A device or storage area used to store data temporarily for differences in rates of data flow, time or occurrence of events, or amounts of data that can be handled by the devices or processes involved in the transfer or use of the data.

---

## **buffer overflow**

**See Also:** buffer

A memory access failure due to the attempt by a process to store data beyond the boundaries of a fixed length buffer, resulting in overwriting of adjacent memory areas or the raising of an overflow exception.

---

## **call graph**

An abstract representation of calling relationships between subroutines in a program.

---

## **capture/playback tool**

**Synonyms:** capture/replay tool, record/playback tool

A type of test execution tool where inputs are recorded during manual testing in order to generate automated test scripts that can be executed later (i.e. replayed). These tools are often used to support automated regression testing.

---

## **certification**

The process of confirming that a component, system or person complies with its specified

---

requirements, e.g., by passing an exam.

---

### **changeability**

**Ref:** ISO 9126    **See Also:** maintainability

The capability of the software product to enable specified modifications to be implemented.

---

### **co-existence**

**Ref:** ISO 9126    **See Also:** portability

The capability of the software product to co-exist with other independent software in a common environment sharing common resources.

---

### **code**

**Ref:** IEEE 610

Computer instructions and data definitions expressed in a programming language or in a form output by an assembler, compiler or other translator.

---

### **combinatorial testing**

**See Also:** classification tree method, n-wise testing, pairwise testing, orthogonal array testing

A black box test design technique in which test cases are designed to execute specific combinations of values of several parameters.

---

### **commercial off-the-shelf (COTS)**

**Synonyms:** off-the-shelf software

A software product that is developed for the general market, i.e. for a large number of customers, and that is delivered to many customers in identical format.

---

### **complexity**

**See Also:** cyclomatic complexity

The degree to which a component or system has a design and/or internal structure that is difficult to understand, maintain and verify.

---

### **compliance**

**Ref:** ISO 9126

The capability of the software product to adhere to standards, conventions or regulations in laws and similar prescriptions.

---

### **component**

**Synonyms:** module, unit

A minimal software item that can be tested in isolation.

---

### **component integration testing**

**Synonyms:** link testing

Testing performed to expose defects in the interfaces and interaction between integrated components.

---

## **component testing**

**Ref:** After IEEE 610

**Synonyms:** module testing, program testing, unit testing

The testing of individual software components.

---

## **condition**

**See Also:** condition testing

**Synonyms:** branch condition

A logical expression that can be evaluated as True or False, e.g.,  $A > B$ .

---

## **condition coverage**

**Synonyms:** branch condition coverage

The percentage of condition outcomes that have been exercised by a test suite. 100% condition coverage requires each single condition in every decision statement to be tested as True and False.

---

## **condition testing**

A white-box test design technique in which test cases are designed to execute condition outcomes.

---

## **configuration**

The composition of a component or system as defined by the number, nature, and interconnections of its constituent parts.

---

## **configuration management**

**Ref:** IEEE 610

A discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements.

---

## **configuration management tool**

A tool that provides support for the identification and control of configuration items, their status over changes and versions, and the release of baselines consisting of configuration items.

---

## **control flow**

A sequence of events (paths) in the execution through a component or system.

---

---

## **control flow analysis**

A form of static analysis based on a representation of unique paths (sequences of events) in the execution through a component or system. Control flow analysis evaluates the integrity of control flow structures, looking for possible control flow anomalies such as closed loops or logically unreachable process steps.

---

## **control flow graph**

An abstract representation of all possible sequences of events (paths) in the execution through a component or system.

---

## **control flow testing**

**See Also:** decision testing, condition testing, path testing

An approach to structure-based testing in which test cases are designed to execute specific sequences of events. Various techniques exist for control flow testing, e.g., decision testing, condition testing, and path testing, that each have their specific approach and level of control flow coverage.

---

## **coverage**

**Synonyms:** test coverage

The degree, expressed as a percentage, to which a specified coverage item has been exercised by a test suite.

---

## **cyclomatic complexity**

**Ref:** After McCabe

**Synonyms:** cyclomatic number

The maximum number of linear, independent paths through a program. Cyclomatic complexity may be computed as  $L = N + 2P$ , where  $L$  = the number of edges/links in a graph,  $N$  = the number of nodes in a graph,  $P$  = the number of disconnected parts of the graph (e.g., a called graph or subroutine).

---

## **data flow**

**See Also:** Beizer

An abstract representation of the sequence and possible changes of the state of data objects, where the state of an object is any of creation, usage, or destruction.

---

## **data flow analysis**

A form of static analysis based on the definition and usage of variables.

---

## **data flow testing**

A white-box test design technique in which test cases are designed to execute definition-use pairs of variables.

---

---

## **data-driven testing**

**Ref:** Fewster and Graham    **See Also:** keyword-driven testing

A scripting technique that stores test input and expected results in a table or spreadsheet, so that a single control script can execute all of the tests in the table. Data-driven testing is often used to support the application of test execution tools such as capture/playback tools.

---

## **debugging tool**

**Synonyms:** debugger

A tool used by programmers to reproduce failures, investigate the state of programs and find the corresponding defect. Debuggers enable programmers to execute programs step by step, to halt a program at any program statement and to set and examine program variables.

---

## **decision**

A program point at which the control flow has two or more alternative routes. A node with two or more links to separate branches.

---

## **decision condition coverage**

The percentage of all condition outcomes and decision outcomes that have been exercised by a test suite. 100% decision condition coverage implies both 100% condition coverage and 100% decision coverage.

---

## **decision condition testing**

A white-box test design technique in which test cases are designed to execute condition outcomes and decision outcomes.

---

## **decision coverage**

The percentage of decision outcomes that have been exercised by a test suite. 100% decision coverage implies both 100% branch coverage and 100% statement coverage.

---

## **decision outcome**

The result of a decision (which therefore determines the branches to be taken).

---

## **decision testing**

A white-box test design technique in which test cases are designed to execute decision outcomes.

---

## **defect**

**Synonyms:** bug, fault, problem

A flaw in a component or system that can cause the component or system to fail to perform its required function, e.g., an incorrect statement or data definition. A defect, if encountered during

---

execution, may cause a failure of the component or system.

---

### **defect management**

**Ref:** After IEEE 1044

**Synonyms:** problem management

The process of recognizing, investigating, taking action and disposing of defects. It involves recording defects, classifying them and identifying the impact.

---

### **defect management tool**

**See Also:** incident management tool

**Synonyms:** bug tracking tool, defect tracking tool

A tool that facilitates the recording and status tracking of defects and changes. They often have workflow-oriented facilities to track and control the allocation, correction and re-testing of defects and provide reporting facilities.

---

### **defect report**

**Ref:** After IEEE 829

**Synonyms:** bug report, problem report

A document reporting on any flaw in a component or system that can cause the component or system to fail to perform its required function.

---

### **definition-use pair**

The association of a definition of a variable with the subsequent use of that variable. Variable uses include computational (e.g., multiplication) or to direct the execution of a path (predicate use).

---

### **domain**

The set from which valid input and/or output values can be selected.

---

### **driver**

**Ref:** After TMap

**Synonyms:** test driver

A software component or test tool that replaces a component that takes care of the control and/or the calling of a component or system.

---

### **dynamic analysis**

**Ref:** After IEEE 610

The process of evaluating behavior, e.g., memory performance, CPU usage, of a system or component during execution.

---

### **dynamic analysis tool**

A tool that provides run-time information on the state of the software code. These tools are most

---



commonly used to identify unassigned pointers, check pointer arithmetic and to monitor the allocation, use and de-allocation of memory and to flag memory leaks.

---

### **dynamic testing**

Testing that involves the execution of the software of a component or system.

---

### **efficiency**

**Ref:** ISO 9126

(1) The capability of the software product to provide appropriate performance, relative to the amount of resources used under stated conditions. (2) The capability of a process to produce the intended outcome, relative to the amount of resources used.

---

### **entry point**

An executable statement or process step which defines a point at which a given process is intended to begin.

---

### **error**

**Ref:** After IEEE 610

**Synonyms:** mistake

A human action that produces an incorrect result.

---

### **exercised**

A program element is said to be exercised by a test case when the input value causes the execution of that element, such as a statement, decision, or other structural element.

---

### **exhaustive testing**

**Synonyms:** complete testing

A test approach in which the test suite comprises all combinations of input values and preconditions.

---

### **exit criteria**

**Ref:** After Gilb and Graham

**Synonyms:** completion criteria, test completion criteria

The set of generic and specific conditions, agreed upon with the stakeholders for permitting a process to be officially completed. The purpose of exit criteria is to prevent a task from being considered completed when there are still outstanding parts of the task which have not been finished. Exit criteria are used to report against and to plan when to stop testing.

---

### **expected result**

**Synonyms:** expected outcome, predicted outcome

The behavior predicted by the specification, or another source, of the component or system under specified conditions.

---

---

## **fail**

**Synonyms:** test fail

A test is deemed to fail if its actual result does not match its expected result.

---

## **failover testing**

**See Also:** recoverability testing

Testing by simulating failure modes or actually causing failures in a controlled environment. Following a failure, the failover mechanism is tested to ensure that data is not lost or corrupted and that any agreed service levels are maintained (e.g., function availability or response times).

---

## **failure**

**Ref:** After Fenton

Deviation of the component or system from its expected delivery, service or result.

---

## **failure mode**

**Ref:** IEEE 610

The physical or functional manifestation of a failure. For example, a system in failure mode may be characterized by slow operation, incorrect outputs, or complete termination of execution.

---

## **fault attack**

**See Also:** negative testing

**Synonyms:** attack

Directed and focused attempt to evaluate the quality, especially reliability, of a test object by attempting to force specific failures to occur.

---

## **fault injection**

**See Also:** fault tolerance

The process of intentionally adding defects to a system for the purpose of finding out whether the system can detect, and possibly recover from, a defect. Fault injection intended to mimic failures that might occur in the field.

---

## **fault seeding**

**Ref:** After IEEE 610

**Synonyms:** bebugging, error seeding

The process of intentionally adding defects to those already in the component or system for the purpose of monitoring the rate of detection and removal, and estimating the number of remaining defects. Fault seeding is typically part of development (pre-release) testing and can be performed at any test level (component, integration, or system).

---

## **fault seeding tool**

**Synonyms:** error seeding tool

A tool for seeding (i.e., intentionally inserting) faults in a component or system.

---

---

## **fault tolerance**

**Ref:** ISO 9126    **See Also:** reliability, robustness

The capability of the software product to maintain a specified level of performance in cases of software faults (defects) or of infringement of its specified interface.

---

## **feature**

**Ref:** After IEEE 1008

**Synonyms:** software feature

An attribute of a component or system specified or implied by requirements documentation (for example reliability, usability or design constraints).

---

## **finite state machine**

**Ref:** IEEE 610

A computational model consisting of a finite number of states and transitions between those states, possibly with accompanying actions.

---

## **formal review**

A review characterized by documented procedures and requirements, e.g., inspection.

---

## **functional integration**

**See Also:** integration testing

An integration approach that combines the components or systems for the purpose of getting a basic functionality working early.

---

## **functional testing**

**See Also:** black-box testing

Testing based on an analysis of the specification of the functionality of a component or system.

---

## **functionality**

**Ref:** ISO 9126

The capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions.

---

## **functionality testing**

The process of testing to determine the functionality of a software product.

---

## **hyperlink**

A pointer within a web page that leads to other web pages.

---

## **hyperlink test tool**

A tool used to check that no broken hyperlinks are present on a web site.

---

## **incident**

**Ref:** After IEEE 1008

**Synonyms:** deviation, software test incident, test incident

Any event occurring that requires investigation.

---

## **indicator**

**Ref:** ISO 14598

A measure that can be used to estimate or predict another measure.

---

## **input**

A variable (whether stored within a component or outside) that is read by a component.

---

## **input value**

**See Also:** input

An instance of an input.

---

## **inspection**

**Ref:** After IEEE 610, IEEE 1028    **See Also:** peer review

A type of peer review that relies on visual examination of documents to detect defects, e.g., violations of development standards and non-conformance to higher level documentation. The most formal review technique and therefore always based on a documented procedure.

---

## **installability**

**Ref:** ISO 9126 .    **See Also:** portability

The capability of the software product to be installed in a specified environment.

---

## **installability testing**

**See Also:** portability testing

Testing the installability of a software product.

---

## **installation wizard**

Supplied software on any suitable media, which leads the installer through the installation process. It normally runs the installation process, provides feedback on installation results, and prompts for options.

---

## **integration**

The process of combining components or systems into larger assemblies.

---

---

## integration testing

**See Also:** component integration testing, system integration testing

Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems.

---

## interoperability

**Ref:** After ISO 9126    **See Also:** functionality

The capability of the software product to interact with one or more specified components or systems.

---

## interoperability testing

**See Also:** functionality testing

**Synonyms:** compatibility testing

Testing to determine the interoperability of a software product.

---

## keyword-driven testing

**See Also:** data-driven testing

**Synonyms:** action word-driven testing

A scripting technique that uses data files to contain not only test data and expected results, but also keywords related to the application being tested. The keywords are interpreted by special supporting scripts that are called by the control script for the test.

---

## learnability

**Ref:** ISO 9126    **See Also:** usability

The capability of the software product to enable the user to learn its application.

---

## level test plan

**See Also:** test plan

A test plan that typically addresses one test level.

---

## load profile

**See Also:** operational profile

A specification of the activity which a component or system being tested may experience in production. A load profile consists of a designated number of virtual users who process a defined set of transactions in a specified time period and according to a predefined operational profile.

---

## load testing

**See Also:** performance testing, stress testing

A type of performance testing conducted to evaluate the behavior of a component or system with increasing load, e.g., numbers of parallel users and/or numbers of transactions, to determine what load can be handled by the component or system.

---

---

## **maintainability**

**Ref:** ISO 9126

The ease with which a software product can be modified to correct defects, modified to meet new requirements, modified to make future maintenance easier, or adapted to a changed environment.

---

## **maintainability testing**

**Synonyms:** serviceability testing

Testing to determine the maintainability of a software product.

---

## **maintenance**

**Ref:** IEEE 1219

Modification of a software product after delivery to correct defects, to improve performance or other attributes, or to adapt the product to a modified environment.

---

## **maintenance testing**

Testing the changes to an operational system or the impact of a changed environment to an operational system.

---

## **man-in-the-middle attack**

The interception, mimicking and/or altering and subsequent relaying of communications (e.g., credit card transactions) by a third party such that a user remains unaware of that third party's presence.

---

## **master test plan**

**See Also:** test plan

A test plan that typically addresses multiple test levels.

---

## **maturity**

**Ref:** ISO 9126    **See Also:** Capability Maturity Model Integration, Test Maturity Model integration, reliability

(1) The capability of an organization with respect to the effectiveness and efficiency of its processes and work practices. (2) The capability of the software product to avoid failure as a result of defects in the software.

---

## **mean time between failures (MTBF)**

**See Also:** reliability growth model

The arithmetic mean (average) time between failures of a system. The MTBF is typically part of a reliability growth model that assumes the failed system is immediately repaired, as a part of a defect fixing process.

---

### **mean time to repair (MTTR)**

The arithmetic mean (average) time a system will take to recover from any failure. This typically includes testing to insure that the defect has been resolved.

---

### **measure**

**Ref:** ISO 14598

The number or category assigned to an attribute of an entity by making a measurement.

---

### **measurement**

**Ref:** ISO 14598

The process of assigning a number or category to an entity to describe an attribute of that entity.

---

### **memory leak**

A memory access failure due to a defect in a program's dynamic store allocation logic that causes it to fail to release memory after it has finished using it, eventually causing the program and/or other concurrent processes to fail due to lack of memory.

---

### **metric**

**Ref:** ISO 14598

A measurement scale and the method used for measurement.

---

### **model-based testing (MBT)**

Testing based on or involving models.

---

### **modified condition / decision coverage (MC/DC)**

**Synonyms:** condition determination coverage, modified multiple condition coverage

The percentage of all single condition outcomes that independently affect a decision outcome that have been exercised by a test case suite. 100% modified condition decision coverage implies 100% decision condition coverage.

---

### **monitoring tool**

**Ref:** After IEEE 610.

A software tool or hardware device that runs concurrently with the component or system under test and supervises, records and/or analyses the behavior of the component or system.

---

### **multiple condition coverage**

**Synonyms:** branch condition combination coverage, condition combination coverage

The percentage of combinations of all single condition outcomes within one statement that have been exercised by a test suite. 100% multiple condition coverage implies 100% modified condition decision coverage.

---

---

## multiple condition testing

**Synonyms:** branch condition combination testing, condition combination testing

A white-box test design technique in which test cases are designed to execute combinations of single condition outcomes (within one statement).

---

## negative testing

**Ref:** After Beizer.

**Synonyms:** dirty testing

Tests aimed at showing that a component or system does not work. Negative testing is related to the tester's attitude rather than a specific test approach or test design technique, e.g., testing with invalid input values or exceptions.

---

## neighborhood integration testing

A form of integration testing where all of the nodes that connect to a given node are the basis for the integration testing.

---

## non-functional requirement

A requirement that does not relate to functionality, but to attributes such as reliability, efficiency, usability, maintainability and portability.

---

## non-functional testing

Testing the attributes of a component or system that do not relate to functionality, e.g., reliability, efficiency, usability, maintainability and portability.

---

## operability

**Ref:** ISO 9126    **See Also:** usability

The capability of the software product to enable the user to operate and control it.

---

## operational acceptance testing

**See Also:** operational testing

**Synonyms:** production acceptance testing

Operational testing in the acceptance test phase, typically performed in a (simulated) operational environment by operations and/or systems administration staff focusing on operational aspects, e.g., recoverability, resource-behavior, installability and technical compliance.

---

## operational profile

The representation of a distinct set of tasks performed by the component or system, possibly based on user behavior when interacting with the component or system, and their probabilities of occurrence. A task is logical rather than physical and can be executed over several machines or be executed in non-contiguous time segments.

---



---

## **output**

A variable (whether stored within a component or outside) that is written by a component.

---

## **pairwise integration testing**

A form of integration testing that targets pairs of components that work together, as shown in a call graph.

---

## **pairwise testing**

**See Also:** combinatorial testing, n-wise testing, orthogonal array testing

A black-box test design technique in which test cases are designed to execute all possible discrete combinations of each pair of input parameters.

---

## **pass**

**Synonyms:** test pass

A test is deemed to pass if its actual result matches its expected result.

---

## **path**

**Synonyms:** control flow path

A sequence of events, e.g., executable statements, of a component or system from an entry point to an exit point.

---

## **path testing**

A white-box test design technique in which test cases are designed to execute paths.

---

## **performance**

**Ref:** After IEEE 610    **See Also:** efficiency

**Synonyms:** time behavior

The degree to which a system or component accomplishes its designated functions within given constraints regarding processing time and throughput rate.

---

## **performance testing**

**See Also:** efficiency testing

Testing to determine the performance of a software product.

---

## **performance testing tool**

A tool to support performance testing that usually has two main facilities: load generation and test transaction measurement. Load generation can simulate either multiple users or high volumes of input data. During execution, response time measurements are taken from selected transactions and these are logged. Performance testing tools normally provide reports based on test logs and graphs of load against response times.

---

---

**pointer**

**Ref:** IEEE 610

A data item that specifies the location of another data item. For example, a data item that specifies the address of the next employee record to be processed.

---

**portability**

**Ref:** ISO 9126

The ease with which the software product can be transferred from one hardware or software environment to another.

---

**portability testing**

**Synonyms:** configuration testing

Testing to determine the portability of a software product.

---

**precondition**

Environmental and state conditions that must be fulfilled before the component or system can be executed with a particular test or test procedure.

---

**predicate**

**See Also:** decision

A statement that can evaluate to true or false and may be used to determine the control flow of subsequent decision logic.

---

**process**

**Ref:** ISO 12207

A set of interrelated activities, which transform inputs into outputs.

---

**product risk**

**See Also:** risk

A risk directly related to the test object.

---

**project**

**Ref:** ISO 9000

A project is a unique set of coordinated and controlled activities with start and finish dates undertaken to achieve an objective conforming to specific requirements, including the constraints of time, cost and resources.

---

**project risk**

**See Also:** risk

A risk related to management and control of the (test) project, e.g., lack of staffing, strict deadlines, changing requirements, etc.

---

---

## **pseudo-random**

A series which appears to be random but is in fact generated according to some prearranged sequence.

---

## **quality**

**Ref:** After IEEE 610

The degree to which a component, system or process meets specified requirements and/or user/customer needs and expectations.

---

## **quality assurance**

**Ref:** ISO 9000

Part of quality management focused on providing confidence that quality requirements will be fulfilled.

---

## **quality attribute**

**Ref:** IEEE 610

**Synonyms:** quality characteristic, software product characteristic, software quality characteristic

A feature or characteristic that affects an item's quality.

---

## **recoverability**

**Ref:** ISO 9126    **See Also:** reliability

The capability of the software product to re-establish a specified level of performance and recover the data directly affected in case of failure.

---

## **recoverability testing**

**See Also:** reliability testing

**Synonyms:** recovery testing

Testing to determine the recoverability of a software product.

---

## **reliability**

**Ref:** ISO 9126

The ability of the software product to perform its required functions under stated conditions for a specified period of time, or for a specified number of operations.

---

## **reliability growth model**

A model that shows the growth in reliability over time during continuous testing of a component or system as a result of the removal of defects that result in reliability failures.

---

## **reliability testing**

Testing to determine the reliability of a software product.

---

---

## **replaceability**

**Ref:** ISO 9126    **See Also:** portability

The capability of the software product to be used in place of another specified software product for the same purpose in the same environment.

---

## **requirement**

**Ref:** After IEEE 610

A condition or capability needed by a user to solve a problem or achieve an objective that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.

---

## **resource utilization testing**

**See Also:** efficiency testing

**Synonyms:** storage testing

The process of testing to determine the resource-utilization of a software product.

---

## **result**

**See Also:** actual result, expected result

**Synonyms:** outcome, test outcome, test result

The consequence/outcome of the execution of a test. It includes outputs to screens, changes to data, reports, and communication messages sent out.

---

## **review**

**Ref:** After IEEE 1028

An evaluation of a product or project status to ascertain discrepancies from planned results and to recommend improvements. Examples include management review, informal review, technical review, inspection, and walkthrough.

---

## **risk**

A factor that could result in future negative consequences.

---

## **risk analysis**

The process of assessing identified project or product risks to determine their level of risk, typically by estimating their impact and probability of occurrence (likelihood).

---

## **risk assessment**

**See Also:** product risk, project risk, risk, risk impact, risk level, risk likelihood

The process of identifying and subsequently analyzing the identified project or product risk to determine its level of risk, typically by assigning likelihood and impact ratings.

---

## **risk identification**

The process of identifying risks using techniques such as brainstorming, checklists and failure history.

---

## **risk level**

The importance of a risk as defined by its characteristics impact and likelihood. The level of risk can be used to determine the intensity of testing to be performed. A risk level can be expressed either qualitatively (e.g., high, medium, low) or quantitatively.

---

## **risk mitigation**

**Synonyms:** risk control

The process through which decisions are reached and protective measures are implemented for reducing risks to, or maintaining risks within, specified levels.

---

## **risk-based testing**

An approach to testing to reduce the level of product risks and inform stakeholders of their status, starting in the initial stages of a project. It involves the identification of product risks and the use of risk levels to guide the test process.

---

## **robustness**

**Ref:** IEEE 610    **See Also:** error-tolerance, fault-tolerance

The degree to which a component or system can function correctly in the presence of invalid inputs or stressful environmental conditions.

---

## **safety**

**Ref:** ISO 9126

The capability of the software product to achieve acceptable levels of risk of harm to people, business, software, property or the environment in a specified context of use.

---

## **scalability**

**Ref:** After Gerrard

The capability of the software product to be upgraded to accommodate increased loads.

---

## **scalability testing**

Testing to determine the scalability of the software product.

---

## **security**

**Ref:** ISO 9126    **See Also:** functionality

Attributes of software products that bear on its ability to prevent unauthorized access, whether accidental or deliberate, to programs and data.

---

---

## **security testing**

**See Also:** functionality testing

Testing to determine the security of the software product.

---

## **security tool**

A tool that supports operational security.

---

## **severity**

**Ref:** After IEEE 610

The degree of impact that a defect has on the development or operation of a component or system.

---

## **short-circuiting**

A programming language/interpreter technique for evaluating compound conditions in which a condition on one side of a logical operator may not be evaluated if the condition on the other side is sufficient to determine the final outcome.

---

## **simulation**

**Ref:** ISO 2382/1

The representation of selected behavioral characteristics of one physical or abstract system by another system.

---

## **simulator**

**Ref:** After IEEE 610, DO178b    **See Also:** emulator

A device, computer program or system used during testing, which behaves or operates like a given system when provided with a set of controlled inputs.

---

## **software**

**Ref:** IEEE 610

Computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system.

---

## **software lifecycle**

The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software lifecycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and sometimes, retirement phase. Note these phases may overlap or be performed iteratively.

---

## **specification**

**Ref:** After IEEE 610

---

A document that specifies, ideally in a complete, precise and verifiable manner, the requirements, design, behavior, or other characteristics of a component or system, and, often, the procedures for determining whether these provisions have been satisfied.

---

### **stability**

**Ref:** ISO 9126    **See Also:** maintainability

The capability of the software product to avoid unexpected effects from modifications in the software.

---

### **standard**

**Ref:** After CMMI

Formal, possibly mandatory, set of requirements developed and used to prescribe consistent approaches to the way of working or to provide guidelines (e.g., ISO/IEC standards, IEEE standards, and organizational standards).

---

### **statement**

**Synonyms:** source statement

An entity in a programming language, which is typically the smallest indivisible unit of execution.

---

### **statement coverage**

The percentage of executable statements that have been exercised by a test suite.

---

### **statement testing**

A white-box test design technique in which test cases are designed to execute statements.

---

### **static analysis**

Analysis of software development artifacts, e.g., requirements or code, carried out without execution of these software development artifacts. Static analysis is usually carried out by means of a supporting tool.

---

### **static analyzer**

**Synonyms:** analyzer, static analysis tool

A tool that carries out static analysis.

---

### **stress testing**

**Ref:** After IEEE 610    **See Also:** performance testing, load testing

A type of performance testing conducted to evaluate a system or component at or beyond the limits of its anticipated or specified workloads, or with reduced availability of resources such as access to memory or servers.

---

## **stub**

**Ref:** After IEEE 610

A skeletal or special-purpose implementation of a software component, used to develop or test a component that calls or is otherwise dependent on it. It replaces a called component.

---

## **suitability**

**Ref:** ISO 9126    **See Also:** functionality

The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives.

---

## **system**

**Ref:** IEEE 610

A collection of components organized to accomplish a specific function or set of functions.

---

## **system integration testing**

Testing the integration of systems and packages; testing interfaces to external organizations (e.g., Electronic Data Interchange, Internet).

---

## **system testing**

**Ref:** Hetzel

Testing an integrated system to verify that it meets specified requirements.

---

## **system under test (SUT)**

See test object.

---

## **technical review**

**Ref:** Gilb and Graham, IEEE 1028    **See Also:** peer review

A peer group discussion activity that focuses on achieving consensus on the technical approach to be taken.

---

## **test**

**Ref:** IEEE 829

A set of one or more test cases.

---

## **test approach**

The implementation of the test strategy for a specific project. It typically includes the decisions made that follow based on the (test) project's goal and the risk assessment carried out, starting points regarding the test process, the test design techniques to be applied, exit criteria and test types to be performed.

---



## **test automation**

The use of software to perform or support test activities, e.g., test management, test design, test execution and results checking.

---

## **test case**

**Ref:** After IEEE 610

A set of input values, execution preconditions, expected results and execution postconditions, developed for a particular objective or test condition, such as to exercise a particular program path or to verify compliance with a specific requirement.

---

## **test data**

Data that exists (for example, in a database) before a test is executed, and that affects or is affected by the component or system under test.

---

## **test design**

**See Also:** test design specification

The process of transforming general test objectives into tangible test conditions and test cases.

---

## **test design technique**

**Synonyms:** test case design technique, test specification technique, test technique

Procedure used to derive and/or select test cases.

---

## **test environment**

**Ref:** After IEEE 610

**Synonyms:** test bed, test rig

An environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test.

---

## **test execution**

The process of running a test on the component or system under test, producing actual result(s).

---

## **test execution automation**

The use of software, e.g., capture/playback tools, to control the execution of tests, the comparison of actual results to expected results, the setting up of test preconditions, and other test control and reporting functions.

---

## **test execution tool**

A type of test tool that is able to execute other software using an automated test script, e.g., capture/playback.

---

## **test harness**

A test environment comprised of stubs and drivers needed to execute a test.

---

## **test implementation**

The process of developing and prioritizing test procedures, creating test data and, optionally, preparing test harnesses and writing automated test scripts.

---

## **test infrastructure**

The organizational artifacts needed to perform testing, consisting of test environments, test tools, office environment and procedures.

---

## **test input**

The data received from an external source by the test object during test execution. The external source can be hardware, software or human.

---

## **test level**

**Ref:** After TMap

**Synonyms:** test stage

A group of test activities that are organized and managed together. A test level is linked to the responsibilities in a project. Examples of test levels are component test, integration test, system test and acceptance test.

---

## **test management**

The planning, estimating, monitoring and control of test activities, typically carried out by a test manager.

---

## **test management tool**

A tool that provides support to the test management and control part of a test process. It often has several capabilities, such as testware management, scheduling of tests, the logging of results, progress tracking, incident management and test reporting.

---

## **test manager**

**Synonyms:** test leader

The person responsible for project management of testing activities and resources, and evaluation of a test object. The individual who directs, controls, administers, plans and regulates the evaluation of a test object.

---

## **test object**

**See Also:** test item

**Synonyms:** system under test

The component or system to be tested.

---

---

## **test plan**

**Ref:** After IEEE 829

A document describing the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, the features to be tested, the testing tasks, who will do each task, degree of tester independence, the test environment, the test design techniques and entry and exit criteria to be used, and the rationale for their choice, and any risks requiring contingency planning. It is a record of the test planning process.

---

## **test planning**

The activity of establishing or updating a test plan.

---

## **test script**

Commonly used to refer to a test procedure specification, especially an automated one.

---

## **test session**

**See Also:** exploratory testing

An uninterrupted period of time spent in executing tests. In exploratory testing, each test session is focused on a charter, but testers can also explore new opportunities or issues during a session. The tester creates and executes on the fly and records their progress.

---

## **test strategy**

A high-level description of the test levels to be performed and the testing within those levels for an organization or programme (one or more projects).

---

## **test suite**

**Synonyms:** test case suite, test set

A set of several test cases for a component or system under test, where the post condition of one test is often used as the precondition for the next one.

---

## **test tool**

**Ref:** TMap **See Also:** CAST

A software product that supports one or more test activities, such as planning and control, specification, building initial files and data, test execution and test analysis.

---

## **testability**

**Ref:** ISO 9126 **See Also:** maintainability

The capability of the software product to enable modified software to be tested.

---

## **tester**

A skilled professional who is involved in the testing of a component or system.

---

---

## **testing**

The process consisting of all lifecycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects.

---

## **traceability**

**See Also:** horizontal traceability, vertical traceability

The ability to identify related items in documentation and software, such as requirements with associated tests.

---

## **understandability**

**Ref:** ISO 9126    **See Also:** usability

The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.

---

## **unreachable code**

**Synonyms:** dead code

Code that cannot be reached and therefore is impossible to execute.

---

## **usability**

**Ref:** ISO 9126

The capability of the software to be understood, learned, used and attractive to the user when used under specified conditions.

---

## **usability testing**

**Ref:** After ISO 9126

Testing to determine the extent to which the software product is understood, easy to learn, easy to operate and attractive to the users under specified conditions.

---

## **user acceptance testing**

**See Also:** acceptance testing

Acceptance testing carried out by future users in a (simulated) operational environment focusing on user requirements and needs.

---

## **variable**

An element of storage in a computer that is accessible by a software program by referring to it by a name.

---

## **verification**

**Ref:** ISO 9000

---

Confirmation by examination and through provision of objective evidence that specified requirements have been fulfilled.

---

### **white-box test design technique**

**Synonyms:** structural test design technique, structure-based test design technique, structure-based technique, white-box technique

Procedure to derive and/or select test cases based on an analysis of the internal structure of a component or system.

---

### **white-box testing**

**Synonyms:** clear-box testing, code-based testing, glass-box testing, logic-coverage testing, logic-driven testing, structural testing, structure-based testing

Testing based on an analysis of the internal structure of the component or system.

---

### **wild pointer**

**See Also:** pointer

A pointer that references a location that is out of scope for that pointer or that does not exist.

---