TVVS - Test, Verification and Validation of Software

Test management

Ana Paiva

apaiva@fe.up.pt   **www.fe.up.pt/~apaiva**

Some definitions

Test management

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

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. [ISTQB]

Some definitions

Test manager / 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. [ISTQB]

Tester

- A skilled professional who is involved in the testing of a component or system. [ISTQB]

Tasks of the Test Leader

- Coordinate the test strategy and plan with project managers and others
- Write or review a test strategy for the project, and test policy for the organization
- Contribute the testing perspective to other project activities, such as integration planning
- Plan the tests – considering the context and understanding the test objectives and risks
- Initiate the specification, preparation, implementation and execution of tests, monitor the test results and check the exist criteria
- Adapt planning based on test results and progress and take any action necessary to compensate for problems

Tasks of the Test Leader

- Set up adequate configuration management of testware for traceability
- Introduce suitable metrics for measuring test progress and evaluating the quality of the testing and the product
- Decide what should be automated, to what degree, and how
- Select tools to support testing and organize any training in tool use for testers
- Decide about the implementation of the test environment
- Write test summary reports based in the information gathered during testing

Tasks of the Tester

- Review and contribute to test plans
- Analyse, review and assess user requirements, specifications and models for testability
- Create test specification
- Set up the test environment
- Prepare and acquire test data

Tasks of the Tester

- Implement tests on all test levels, execute and log the tests, evaluate the results and document the deviations from expected results
- Use test administration or management tools and test monitoring tools as required
- Automate tests
- Measure performance of components and systems
- Review tests developed by others

Test planning activities

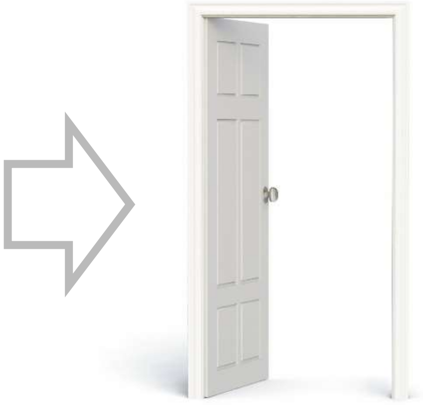
- Determining the scope and risks and identifying the objectives of testing
- Defining the overall approach of testing, including the definition of the test levels and entry and exit criteria
- Integrating and coordinating the testing activities into the software life cycle activities
- Making decisions about what to test, what roles will perform the test activities, how the test activities should be done, and how the test results will be evaluated
- Scheduling test analysis and design activities

Test planning activities

- Scheduling test implementation, execution and evaluation
- Assigning resources for the different activities defined
- Defining the amount, level of detail, structure and templates for the test documentation
- Selecting metrics for monitoring and controlling test preparation and execution, defect resolution and risk issues
- Selecting the level of detail for test procedures in order to provide enough information to support reproducible test preparation and execution

Entry criteria

- Typically entry criteria may cover the following:
  - Test environment availability and readiness
  - Test tool readiness in the test environment
  - Testable code availability
  - Test data availability



Exit criteria

- Thoroughness measures, such as coverage of code, functionality or risk
- Estimates of defect density or reliability measures
- Cost
- Residual risks, such as defects not fixed or lack of test coverage in areas
- Schedules such as those based on time to market

Test estimation

Test estimation

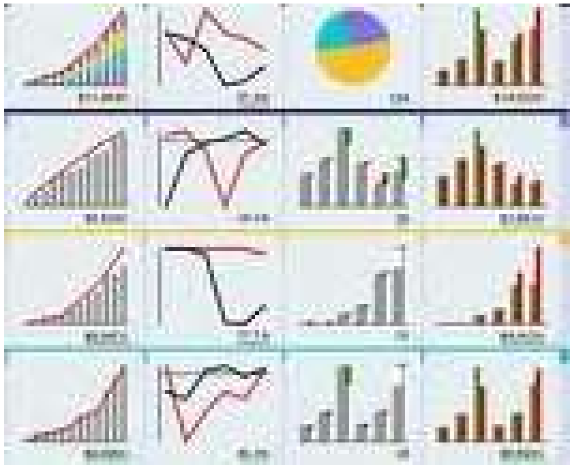
- The calculated approximation of a result related to various aspects of testing (e.g., effort spent, completion date, costs involved, number of test cases, etc.) which is usable even if input data may be incomplete, uncertain, or noisy. [ISTQB]

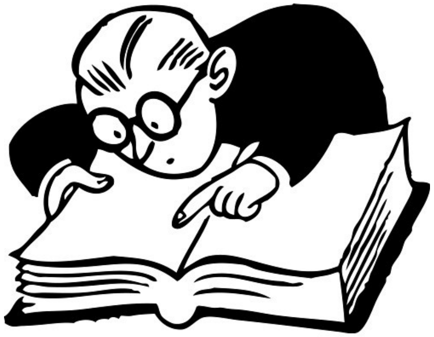
- How do you estimate the test effort

?

Test estimation

- The metrics-based approach: estimating the testing effort based on metrics of former or similar projects or based on typical values
  - E.g., complexity; functional point analysis, test point analysis, …
- The expert-based approach: estimating the tasks based on estimates made by the owner of the tasks or by experts





Test progress monitoring

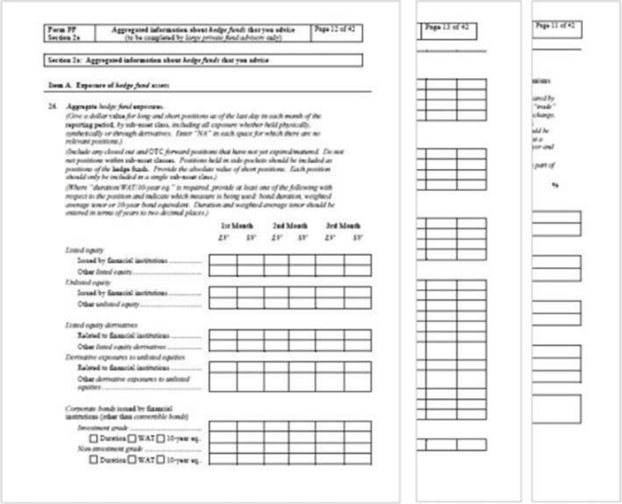
- Percentage of work done in test case preparation
- Percentage of work done in test environment preparation
- Test case execution
- Defect information
- Test coverage of requirements, risks or code
- Subject confidence of testers in the product
- Dates of test milestones
- Testing costs, including the cost compared to the benefit of finding the next defect or to run the next test

Test control

- Making decisions based on information from test monitoring
- Re-prioritizing tests when an identified risk occurs
- Changing the test schedule due to availability or unavailability of a test environment
- Setting an entry criterion requiring fixes to have been re-tested by a developed before accepting them into a build

Test reporting

- What happened during a period of testing, such as dates when exit criteria were met
- Analysed information and metrics to support recommendations and decisions about future actions, such as an assessment of defects remaining, the economic benefit of continued testing, outstanding risks, and the level of confidence in the tested software



Form PF Section 2a	**Aggregated information about *hedge funds* that you advise** (to be completed by *large private fund advisers* only)	Page 12 of 42

Section 2a: Aggregated information about *hedge funds* that you advise

Item A. Exposure of *hedge fund* assets

24. **Aggregate** *hedge fund* **exposures.**
(Give a dollar value for long and short positions as of the last day in each month of the **reporting period**, *by sub-asset class, including all exposure whether held physically, synthetically or through derivatives. Enter "NA" in each space for which there are no relevant positions.)*
(Include any closed out and OTC forward positions that have not yet expired/matured. Do not net positions within sub-asset classes. Positions held in side pockets should be included as positions of the **hedge funds**. *Provide the absolute value of short positions. Each position should only be included in a single sub-asset class.)*
(Where "duration/WAT/10-year eq." is required, provide at least one of the following with respect to the position and indicate which measure is being used: bond duration, weighted average tenor or 10-year bond equivalent. Duration and weighted average tenor should be entered in terms of years to two decimal places.)

	1st Month		2nd Month		3rd Month	
	LV	SV	LV	SV	LV	SV
Listed equity						
Issued by financial institutions						
Other *listed equity*						
Unlisted equity						
Issued by financial institutions						
Other *unlisted equity*						
Listed equity derivatives						
Related to financial institutions						
Other *listed equity derivatives*						
Derivative exposures to unlisted equities						
Related to financial institutions						
Other *derivative exposures to unlisted equities*						
Corporate bonds issued by financial institutions (other than *convertible bonds*)						
Investment grade						
□ Duration □ WAT □ 10-year eq.						
Non-investment grade						
□ Duration □ WAT □ 10-year eq.						

Page 13 of 42

Page 11 of 42

Test strategy, test approach

Typical approaches include:

- **Analytical approaches**, such as risk-based testing where testing is directed to areas of greatest risk
- **Model-based approaches**, such as stochastic testing using statistical information about failure rates or usage
- **Methodical approaches**, such as failure-based, experience-based, checklist-based, and quality characteristic-based
- **Process- or standard-compliant approaches**, such as those specified by industry-specific standards or the various agile methodologies

Test strategy, test approach

Typical approaches include:

- **Dynamic and heuristic approaches**, such as exploratory testing where testing is more reactive to events than pre-planned, and where execution and evaluation are concurrent tasks
- **Consultative approaches**, such as those in which test coverage is driven primary by the advice and guidance of technology and/or business domain experts outside the test team
- **Regression-averse approaches**, such as those that include reuse of existing test material, extensive automation of functional regression tests, and standard test suites

Risk based approach to testing

- Determine the test techniques to be employed
- Determine the extend of testing to be carried out
- Prioritize testing in an attempt to find the critical defects as early as possible
- Determine whether any non-testing activities could be employed to reduce risk (e.g., providing training to inexperienced designers)

Project risks

- Organizational factors
  - Skill, training and staff shortages
  - Personnel issues
  - Political issues, such as
    - Problems with testers communicating their needs and test results
    - Failure by the team to follow up on information found in testing and reviews
  - Improper attitude toward or expectations of testing

Project risks

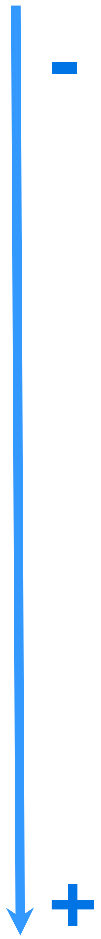
- Technical issues
  - Problems in defining the right requirements
  - The extend to which requirements cannot be met given existing constraints
  - Test environment not ready on time
  - Late data conversion, migration planning and development and testing data conversion/migration tools
  - Low quality of the design, code, configuration data, test data and tests
- Supplier issues
  - Failure of a third party
  - Contractual issues

Product risks

- Failure-prone software delivered
- The potential that the software/hardware could harm to an individual or company
- Poor software characteristics (e.g., functionality, reliability,…)
- Poor data integrity and quality (e.g., data migration…)
- Software that does not perform its intended functions

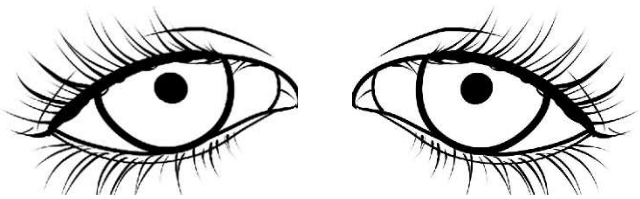
Test independence

- No independent testers – developers test their own code
- Independent testers within the development teams
- Independent test team or group within the organization, reporting to project management or executive management
- Independent testers from business organization or user community
- Independent test specialists for specific test types such as usability
- Independent testers outsourced or external to the organization



Benefits of independence

- Independent testers see other and different defects and are unbiased
- Independent testers can verify assumptions people made during specification and implementation of the system



Drawbacks of test independence

- Isolation from the development team (if treated as totally independent)
- Developers may lose a sense of responsibility for quality
- Independent testers may see as a bottleneck or blamed for delays in release

Configuration management

Configuration management

- 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. [ISTQB]

Configuration management

- For testing, configuration management may involve ensuring the following:
  - All items of testware are identified, version controlled, tracked for changes, related to each other and related to development items so that traceability can be maintained throughout the test process
  - All identified documents and software items are referenced unambiguously in test documentation

Incident management

Incident management

- The process of recognizing, investigating, taking action and disposing of incidents. It involves logging incidents, classifying them and identifying the impact. [ISTQB]

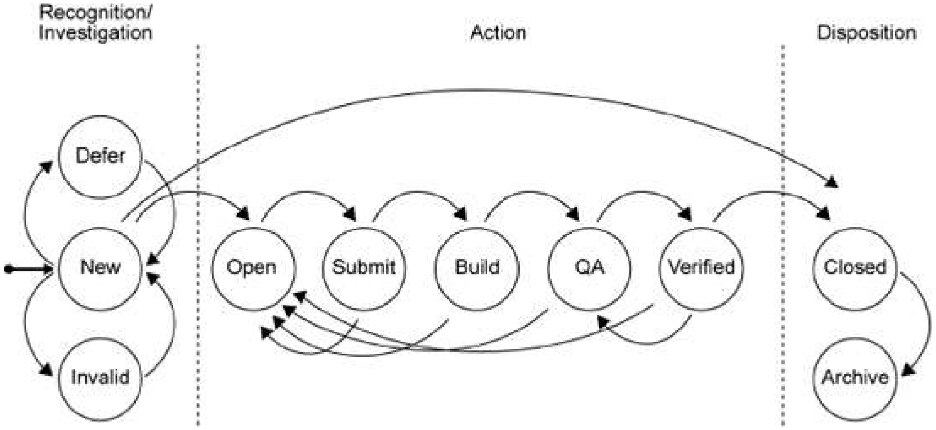
Incident management tool

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

Incident management

Incident reports have the following objectives:

- Provide developers and other parties with feedback about the problem to enable identification, isolation and correction as necessary
- Provide test leaders a means of tracking the quality of the system under test and the progress of the testing
- Provide ideas for test process improvement



Incident

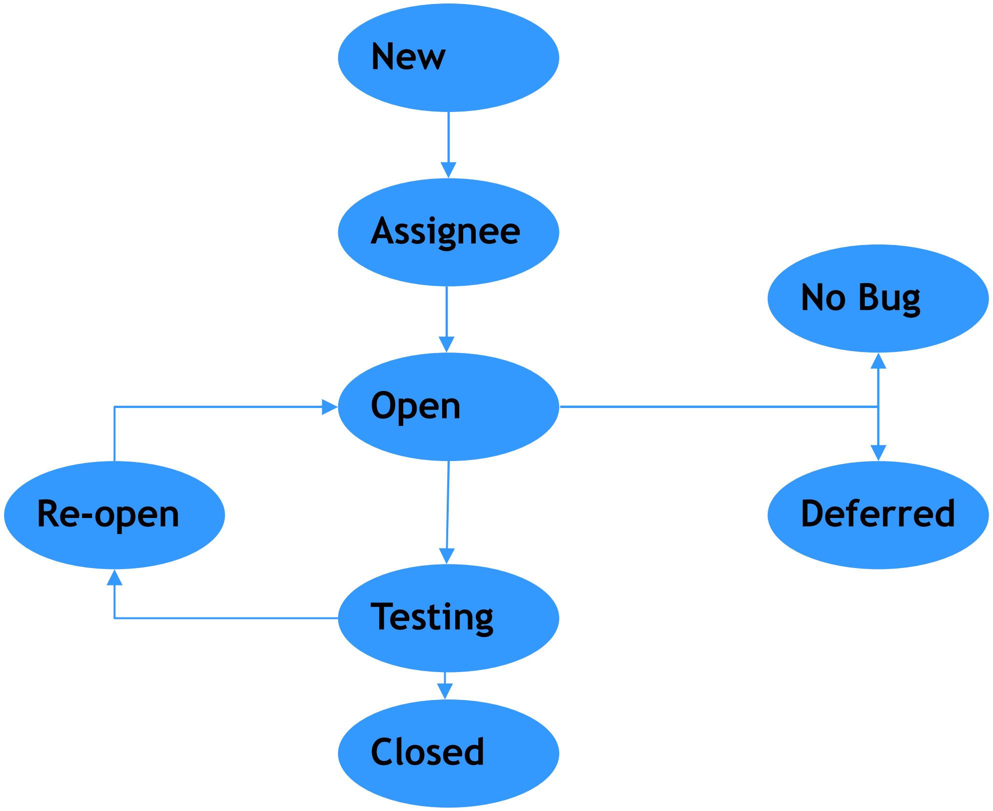
Incident

- Any event occurring that requires investigation. [ISTQB]

Incident

- Incident is raised whenever actual results vary from expected results
- Also called as bugs, defects, problems or issues
- Incident management is very important
- Project status is calculated from incident status
- Tools are available to help incident management

Defect Life Cycle - example



The details that are normally included on an incident report

- Data of issue, issuing organization, assigner, assigne
- Scope, severity and priority of the incident
- References, including the identity of the test case specification that revealed the problem
- Expected and actual results
- Date the incident was discovered
- Identification of the test item (configuration item and environment)
- Software or system life-cycle process in which the incident was observed

The details that are normally included on an incident report

- Description of the incident to enable reproduction and resolution, including logs, database dumps or screenshots
- Urgency/priority to fix
- Status of the incident (e.g., open, duplicate, waiting to be fixed, fixed awaiting confirmation test or closed)
- Conclusions, recommendations and approvals
- Change history, such as the sequence of actions taken by project team members with respoect to the incident to isolate, repair and confirm it as fixed

Test Incident Report - IEEE829

- Test Incident Report Identifier
- Summary
  - Provide enough details to enable others to understand how the incident was discovered
- Incident Description
  - Provide as much details on the incident as possible. For example: Inputs; Expected Results; Actual Results; Anomalies; Date and Time; Procedure Step; Attempts to Repeat; Testers; Observers
- Impact
  - Describe the actual/potential damage caused by the incident.
    - Severity – The potential impact to the system
    - Priority – The order in which the incidents are to be addressed