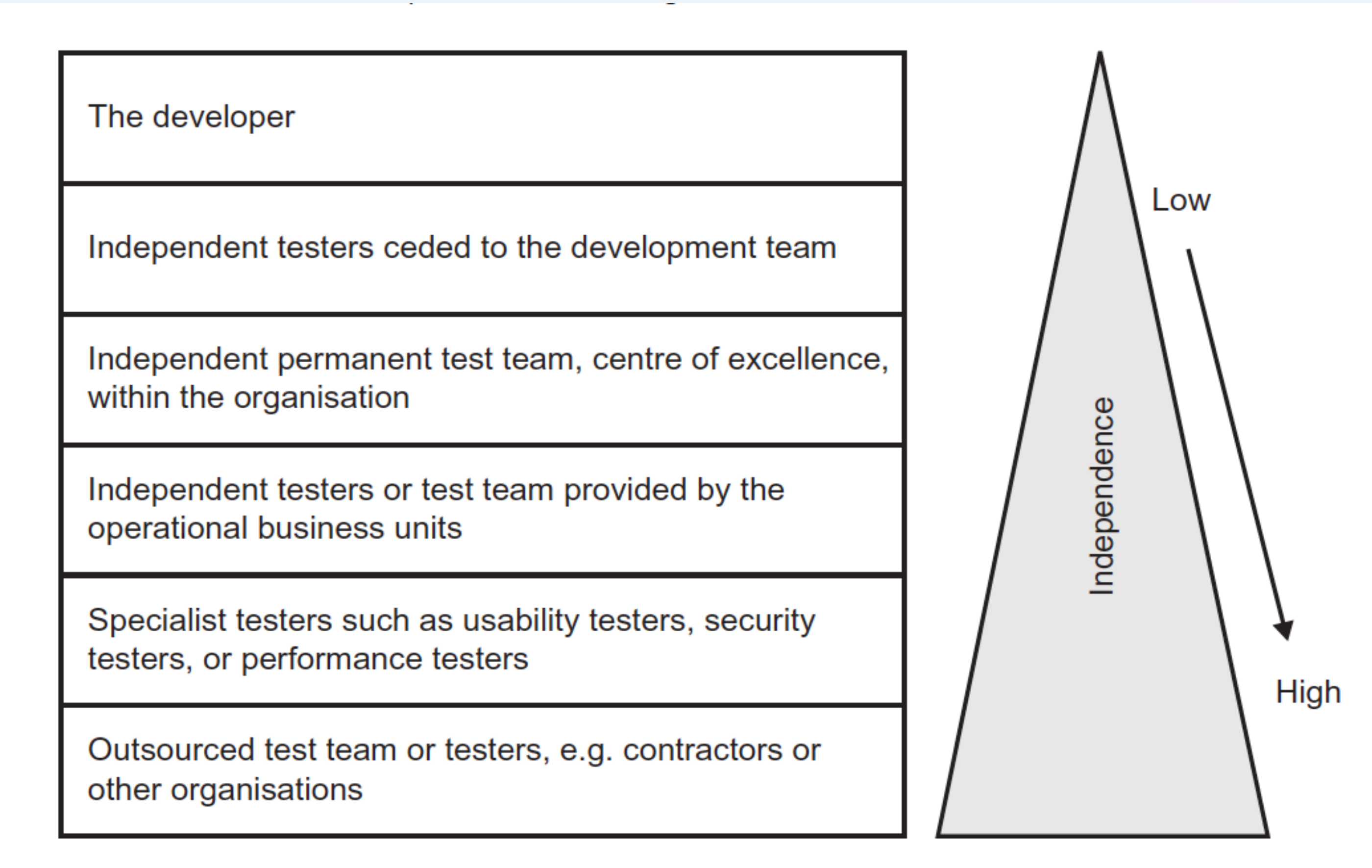
Test organization and independence

- Independent testing is testing carried out by someone other than the creator (developer) of the code being tested.
- By remaining independent it is possible to improve the effectiveness of testing if implemented correctly.
- A developer, as the creator and owner of documents and code related to development, perceives these deliverables as being correct when they are delivered. The general awareness that we all make mistakes is, at this stage, overridden by the belief that what has been produced is what is required. A tester, by contrast, will take the view that anything delivered for testing is likely to contain errors and will search diligently to identify and locate those errors.

Test organization and independence

- There are many options for many levels of independence.
- In general, the more remote a tester is from the production of the document, the greater is the level of independence.



The developer

Independent testers ceded to the development team

Independent permanent test team, centre of excellence, within the organisation

Independent testers or test team provided by the operational business units

Specialist testers such as usability testers, security testers, or performance testers

Outsourced test team or testers, e.g. contractors or other organisations

Test organization and independence

- Independence come at a price. The greater the level of independence, the greater the likelihood of errors in testing arising from unfamiliarity.
- In smaller organizations where everybody contributes to every activity it is harder to differentiate the role of the tester from any other role, and therefore testers may not be very independent at all.

  The key in these circumstances is for the testers to have independence of mind, not necessarily to be in an independent (separate) team.
- In organizations where there are clearly defined roles it is a lot easier for a tester to remain independent.

Test organization and independence

- It is also possible to mix and match the levels of independence. For example:

  a test team made up of permanent resources, business unit resources and contractors.

  For large, complex or safety-critical projects, it is usually best to have multiple levels of testing, with some or all of the levels done by independent testers.

- The ‘agile’ approach to development challenges the traditional approach to independence. In this approach everybody takes on multiple roles and so maintaining total independence is not always possible.

Tasks of test leader and tester

- Test tasks are traditionally carried out by people who make testing a career; however, test tasks may also be carried out by non-testers such as a project manager, quality manager, developer, business and domain expert, infrastructure or IT operations.
- The availability of resources usually determines the resource types that are deployed on each project, e.g. if there are no career testers available an organization may identify non-testing IT or business resources to carry out the role of tester for a specific project or time period.
- The testing roles can be undertaken by anyone with the required skills or who is given the right training. For example, the role of a test leader could be undertaken by a project manager. The decision as to who does what will depend on how a project or organisation is structured, as well as the size and number of resources working on a given project.

Tasks of test leader and tester

It is important to understand here the difference between a **testing role** and a **testing job**.

- **A role** is an activity, or a series of activities given to a person to fulfil, e.g. the role of test leader. A person may therefore have more than one role at any moment depending on their experience and the level of workload on a project.
- **A job** is effectively what an individual is employed to do, so one or many roles could make up a job. For example, a test leader could also be a tester.

Tasks of test leader

- The tasks undertaken by a test leader align very closely with those undertaken by a project manager and align closely with standard approaches to project management.

  Typical test leader tasks may include:

  - Coordinating the development of the test strategy and plan with project managers and others.
  - Writing or reviewing test strategies produced for the project, and test policies produced for the organization.
  - Contributing the testing perspective to other project activities, such as development delivery schedules.

Tasks of test leader

- Planning the development of the required tests – which will include ensuring that the development uses the correct understanding of the risks, selecting the required test approaches (test levels, cycles, approach, objectives and incident management planning), estimating the time and effort and converting to the cost of testing and acquiring the right resources.
- Managing the specification, preparation, implementation and execution of tests, including the monitoring and control of all the specification and execution.
- Taking the required action, including adapting the planning, based on test results and progress (sometimes documented in status reports), and any action necessary to compensate for problems or delays.
- Ensuring that adequate configuration management of testware is in place and that the testware is fully traceable, e.g. there is a hierarchical relationship established between the requirements and the detailed specification documents.

Tasks of test leader

- Putting in place suitable metrics for measuring test progress and evaluating the quality of the testing delivered and the product.
- Agreeing what should be automated, to what degree, and how, ensuring it is implemented as planned.
- Where required, selecting tools to support testing and ensuring any tool training requirements are met.
- Agreeing the structure and implementation of the test environment.
- Scheduling all testing activity.
- At the end of the project, writing a test summary report based on the information gathered during testing.

These tasks are not, however, all of the tasks that could be carried out by test leaders, just the most common ones. The key is to ensure that everyone is aware of who is doing what tasks, that they are completed on time and within budget, and that they are tracked through to completion.

Tasks of tester

The tasks typically undertaken by a tester may include:

- Reviewing and contributing to the development of the test plans
- Analyzing, reviewing and assessing user requirements, specifications and models for testability
- Creating test specifications from the test bases
- Setting up the test environment (often coordinating with system administration and network management).
- Preparing and acquiring/copying/creating test data
- Implementing tests on all test levels, executing and logging the tests, evaluating the results and documenting the deviations from expected results as defects.

Tasks of tester

- Using test administration or management and test monitoring tools as required.
- Automating tests (may be supported by a developer or a test automation expert).
- Where required, running the tests and measuring the performance of components and system (if applicable).
- Reviewing tests developed by other testers.

Tasks of tester

- If specialist testers are not available, then additional resources could be used at different test levels:
  - For component and integration testing, any additional roles would typically be filled by someone from a development background.
  - For system and user acceptance testing, any additional roles would typically be filled by someone from a business or user background.
  - System operators (sometimes known as production support) would be responsible for operational acceptance testing.

Test planning

Mirror, mirror on the wall, how much testing to test it all?
TEST 1
TEST 2
TEST 3
ILLUSTRATED BY SEGUE TECHNOLOGIES

Test planning

Test planning is the most important activity undertaken by a test leader in any test project.

It ensures that there is initially a list of tasks and milestones in a baseline plan to track progress against, as well as defining the shape and size of the test effort.

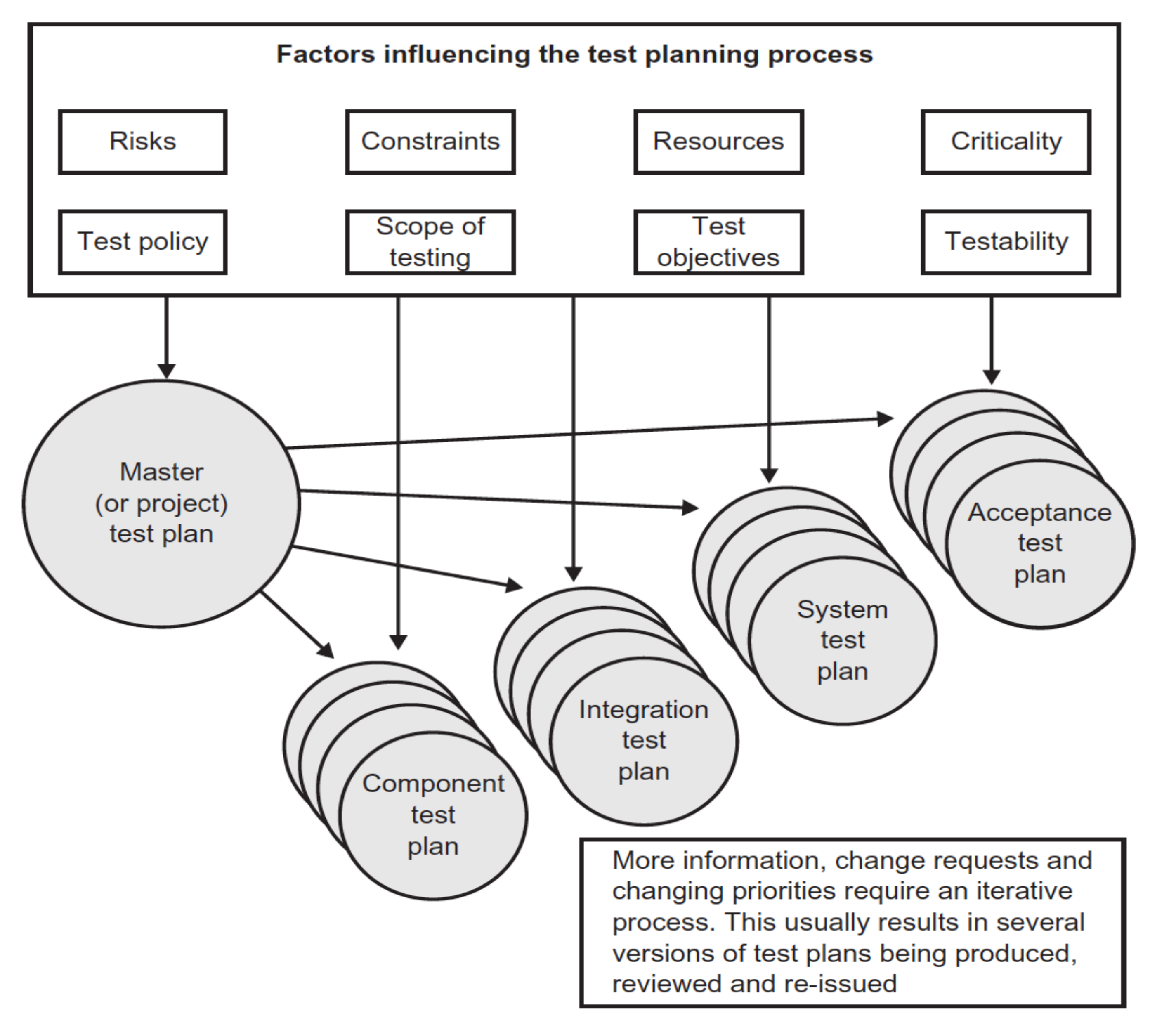
Test planning is used in development and implementation projects as well as maintenance (change and fix) activities.

Test planning

The main document produced in test planning is often called a master test plan or a project test plan. It defines the high level of the test activities being planned. It is normally produced during the early phases of the project (e.g. initiation) and will provide sufficient information to enable a test project to be established.

The details of the test-level activities are documented within test-level plans, e.g. the system test plan. These documents will contain the detailed activities and estimates for the relevant test level.

Test plans in V-model



More information, change requests and changing priorities require an iterative process. This usually results in several versions of test plans being produced, reviewed and re-issued

Test planning

- The contents sections for either a master test plan or test level plans are normally identical or very similar.
- IEEE 829, the Standard for Software Test Documentation, contains details of what the content of the plans should be.
- The planning is a continual activity that takes place in all life-cycle stages. As risks and changes occur, the plan and planning should be amended to recognize these and reflect the current position.

Test plan sections

- Test plan identifier
  - A unique identifying reference (typically document name and version)
- Introduction
  - Brief introduction to the document and project for which it has been produced
- Test items
  - A test item is a software item that is the object of testing
  - A software item is one or more items of source code, object code, control data, etc.

    This section should contains any documentation references, e.g. design documents.

Test plan sections

- Features to be tested
  - Identify all software features and combinations of features and associated test design specification
- Features not to be tested
  - Identify all software features and significant combinations and state the reasons for not including them
- Approach
  - Details the overall approach to testing; this could include a detailed process definition, or could refer to other documentation where the detail is documented, i.e. a test strategy

Test plan sections

- Item pass/fail criteria
  - Used to determine whether a software item has passed or failed its test
- Suspension and resumption requirements
  - Suspension requirements define criteria for stopping part or all of the testing activity
  - Resumption requirements specify the requirements to resume testing
- Test deliverables
  - The documents that testing will deliver, such as:
    - Test plans (for each test level),
    - Test specifications (design, case and procedure)
    - Test summary reports

Test plan sections

- Testing tasks
  - All tasks for planning and executing the testing, including the intertask dependencies
- Environmental needs
  - Definition of all environmental requirements such as hardware, software, PCs, desks, etc
- Responsibilities
  - Identifies the roles and tasks to be used in the test project and who will own them
- Staffing and training needs
  - Identifies any actual staffing requirements and any specific skills and training requirements, e.g. autimation

Test plan sections

- Schedule
  - Document delivery dates and key milestones
- Risks and contingencies
  - High-level project risks and assumptions and contingencies a contingency plan for each risk
- Approvals
  - Identifies all approvers of the document, their titles and the date of signature

Test planning

A useful revision aid to help remember the 16 sections of the IEEE 829 test plan is the acronym '***SPACEDIRT***', each letter mapping to one or several sections of the test plan:

S scope (including test items, features to be tested and features not to be tested)

P people (including responsibilities, staff and training and approvals)

A approach

C criteria (including item pass/fail criteria and suspension and resumption requirements)

E environment needs

D deliverables (test)

I identifier and introduction (test plan)

R risks and contingencies

T testing tasks and schedule

JUST TEST IT.