1 Overview	2 Life cycle components	3 Infrastructure components	4 Management components	5 Standards and Organizing
6 Static tesing	7 Dynamic testing	8 Test management	9 Tools	

Test management

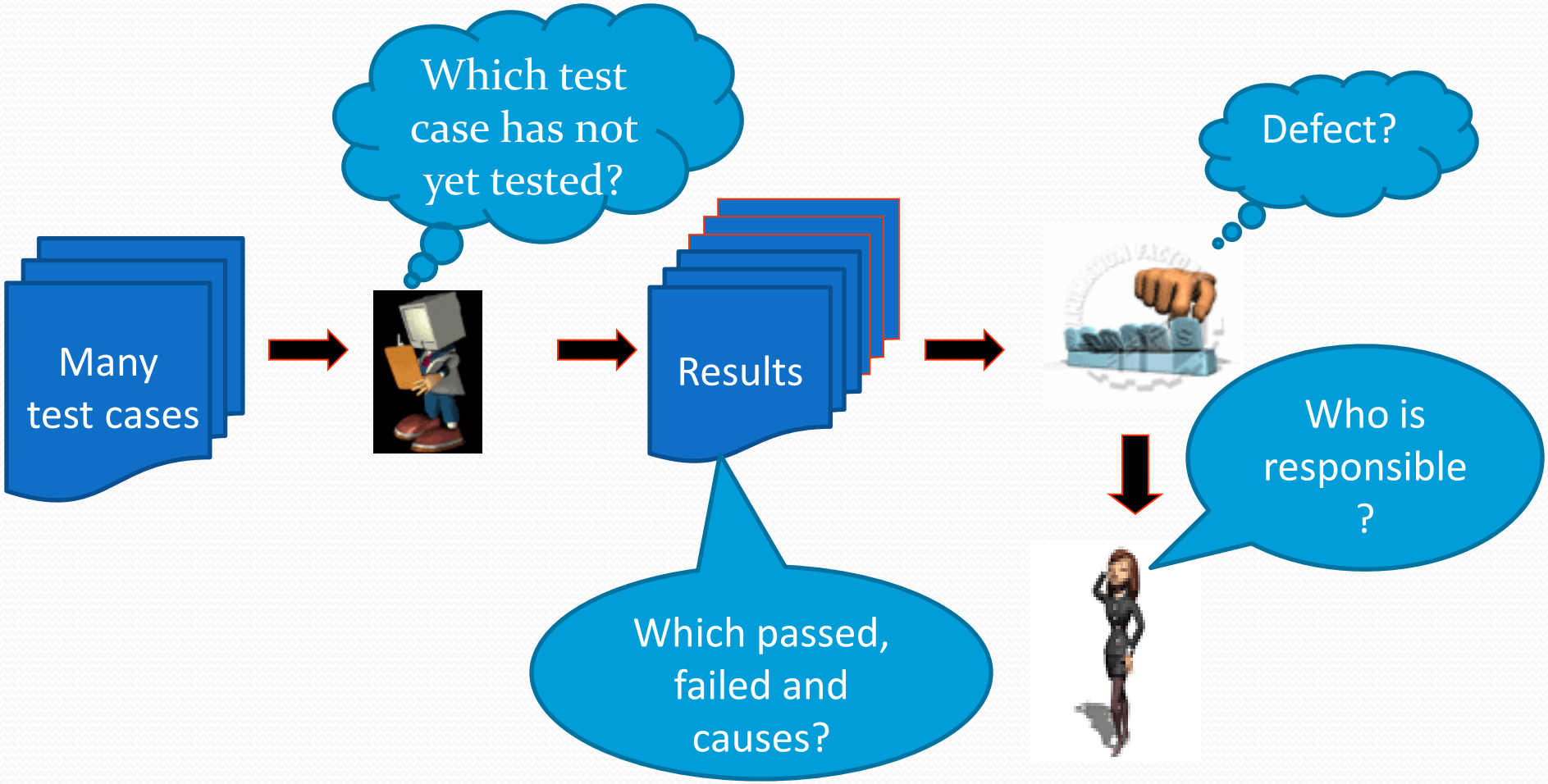
Learning objectives

- Recognize purpose and substance of test plans
- Summarize the purpose and content of the **test summary report document** and **incident report** according to [IEEE 829]
- Differentiate between two conceptually different estimation approaches: the **metrics-based** approach and the **expert-based** approach
- Understand and interpret test metrics for test reporting and test control

References

- Dorothy Grahamet, Erik van Veenendaal, Isabel Evans, Rex Black. *Foundations of software testing: ISTQB Certification*
- FSOFT course

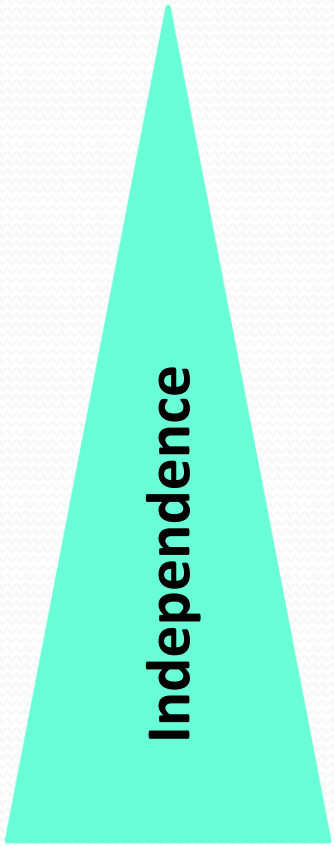
Why needs test management?



Contents

Organisational structures for testing

Only developer responsibility
Development team responsibility
Tester(s) on the development team
Dedicated team of testers (not developers)
Internal test consultants (advice, review, support, not perform the testing)
Outside organisation (3rd party testers)



Testing by developers

- Pro’s:
  - know the code best
  - will find problems that the testers will miss
  - they can find and fix faults cheaply
- Con’s
  - difficult to destroy own work
  - tendency to 'see' expected results, not actual results
  - subjective assessment

Testing by development team

- Pro's:
  - some independence
  - technical depth
- Con's
  - pressure of own development work
  - technical view, not business view
  - lack of testing skill

Tester(s) on development team

- Pro's:
  - independent view of the software
  - dedicated to testing, no development responsibility
  - part of the team, working to same goal: quality
- Con's
  - lack of respect
  - lonely, thankless task
  - corruptible (peer pressure)
  - a single view / opinion

Independent test team

- Pro's:
  - dedicated team just to do testing
  - specialist testing expertise
  - testing is more objective & more consistent
- Con's
  - testers and the test team to become isolated
  - may be antagonistic / confrontational
  - over-reliance on testers, insufficient testing by developers

Internal test consultants

- Pro's:
  - highly specialist testing expertise, **providing support and help** to improve testing done by all
  - better planning, estimation & control from **a broad view** of testing in the organisation
- Con's
  - level of expertise enough?
  - needs good "people" skills - communication

Outside organisation (3rd party)

- Pro's:
  - highly specialist testing expertise (if out-sourced to a good organisation)
  - independent of internal politics
- Con's
  - lack of company and product knowledge
  - expensive?

Usual choices

- Component testing
  - done by programmers (or buddy)
- Integration testing in the small
  - done by programmers
- System testing
  - often done by independent test team
- Acceptance testing
  - done by users (with technical help)

Tasks of a test leader

- Devise the **test objectives, organizational test policies, test strategies** and **test plans**
- **Estimate** the testing to be done and **negotiate** with management
- Recognize when **test automation**
- **Lead**, **guide** and **monitor** the analysis, design, implementation and execution of the test cases, test procedures and test suites
- Ensuring that adequate configuration management of testware

Tasks of a test leader (cont'd)

- Ensure the **test environment** is put into place before test execution and managed during test execution
- Schedule the tests for **execution**, then **monitor**, **measure**, **control** and **report** on the test progress, the product quality status and the test results
- Writing a **test summary report**

May be as a project manager, a development manager or a quality assurance manager,...

The ability of planning, monitoring and controlling the testing work

Tasks of a tester

- Review and contribute to **test plans**
- Analyze, review and assess **requirements** and **design specifications**
- Involved in or even be the primary people **identifying test conditions** and **creating test cases, test procedure specifications** and **test data**, and may **automate or help to automate the tests**
- **Setting up** the test environment or **assist** system administration and network management staff

Tasks of a tester (cont'd)

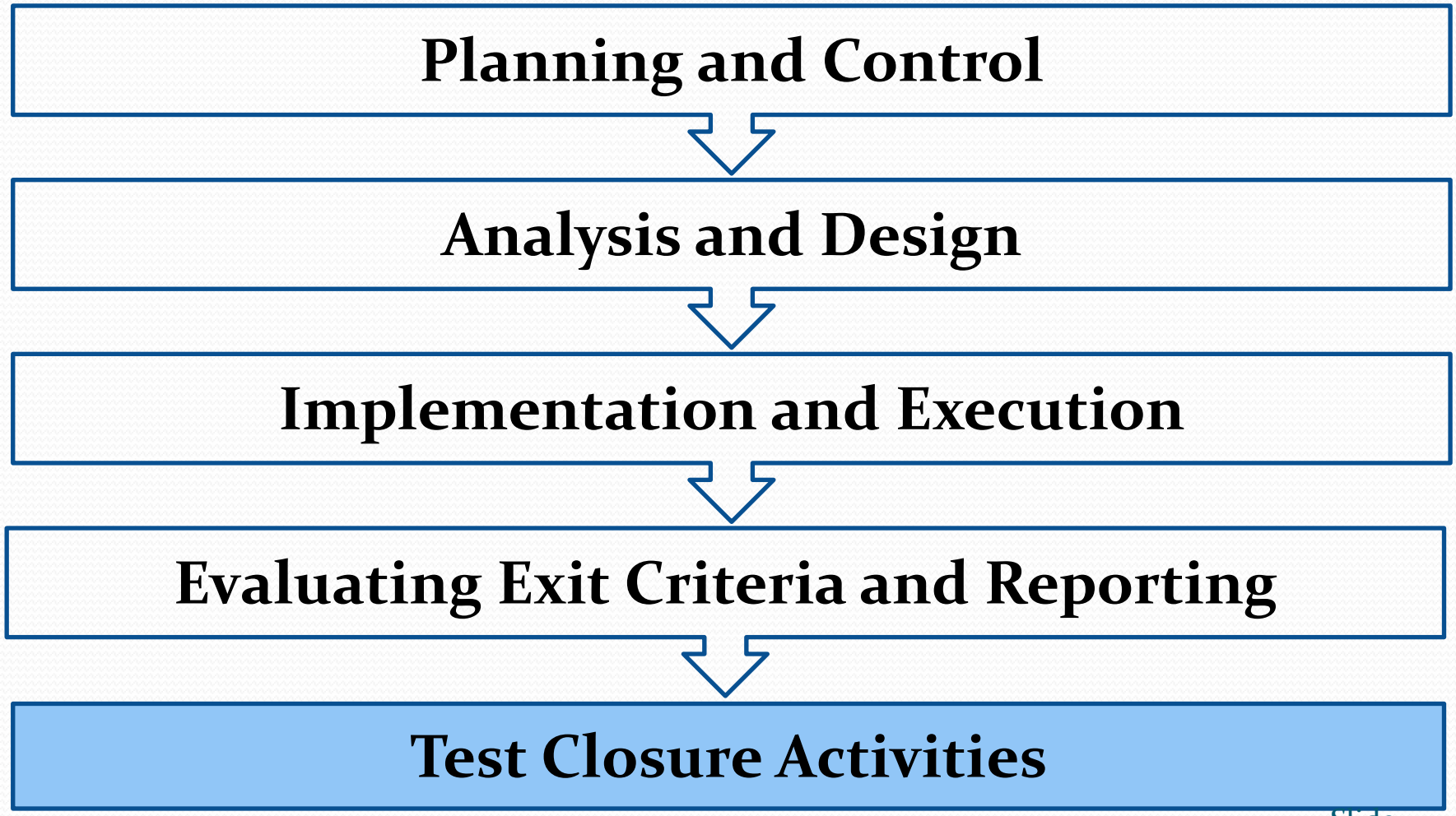
- **Implement** tests on all test levels, **evaluating** the results from expected results
- **Monitor** the testing and the test environment, and often **gather** performance metrics
- **Review** each other's work

Skills needed in testing

- Need basic professional and social qualifications
- Three main areas
  - application or business domain
  - technology
  - testing
- Specialization of skills and separation of roles
- Most projects can benefit from the participation of professional testers, as amateur testing alone will usually not suffice

Contents

Fundamental test process



Fundamental test process Inputs & Outputs

Planning and Control	Analysis and Design	Implementation and Execution	Test Report and Evaluation	Test Closure
1. Business Requirements 2. Functional Specification 3. User Stories (Draft) 4. Project Plan Document 5. Solution Document	1. Business Requirements 2. Functional Specification 3. User Stories 4. Project Plan Document 5. Solution Document 6. Test Plan	1. Test Designs 2. Test Cases 3. Test Procedures 4. Test Data	1. Test Results 2. Test Scripts	1. Test Plan 2. Test Reports
1. Test Plan	1. Test Designs 2. Test Cases 3. Test Procedures 4. Test Data	1. Test Results 2. Test Scripts 3. Defects	1. Test Reports 2. Exit Criteria Evaluation	1. Test Report Summary

Test plan

- A test plan is the project plan for the testing work to be done
  - it is not a test design specification, a collection of test cases or a set of test procedures
- Why do we write test plans?
  - writing a test plan guides our thinking
  - serve as vehicles for communicating with other members of the project team, testers, peers, managers and other stakeholders
  - help to manage change
- Test Plan (FSOFT)
- Test Plan sample (FSOFT)

IEEE 829 Standard Test Plan Template

1. Test Plan Identifier
2. Introduction
3. Test Items
4. Features to be tested
5. Features not to be tested

What?

6. Approach
7. Item pass/fail criteria
8. Suspension criteria and resumption requirements
9. Test deliverables
10. Testing tasks
11. Environmental needs

How?

12. Responsibilities
13. Staffing and training needs

Who?

14. Schedule

When?

15. Risks and contingencies
16. Approvals

Estimation techniques

- The **metrics-based** approach
  - rely upon data collected from previous or similar projects
    - the number of test conditions
    - the number of test cases written
    - the number of test cases executed
    - the time taken to develop test cases
    - the time taken to run test cases
    - the number of defects found
    - the number of environment outages and how long on average each one lasted
- The **expert-based** approach
  - involves consulting the people who will do the work and other people with expertise on the tasks to be done

Estimation techniques
Metric-based estimation

- Can be as simple or sophisticated as you make it
  - classifying the project in terms of size (small, medium or large) and complexity (simple, moderate or complex)
  - look at the average effort per test case in similar past projects
  - building mathematical models

Estimation techniques
Expert-based estimation

- Use the experience of owners of the relevant tasks or experts to derive an estimate
  - business experts
  - test process consultants
  - developers
  - technical architects
  - analysts and designers
  - anyone with knowledge of the application to be tested or the tasks involved in the process

Contents

Test progress monitoring and control

- Include:
  - Test progress monitoring
  - Reporting test status
  - Test control

- Reference:
  - Test Report_v1.0 (FSOFT)
  - Test Summary Report – Template (FSOFT)

Test progress monitoring

- Purposes
  - give the test team and the test manager feedback
  - provide visibility about the test results
  - measure the status of the testing against the exit criteria
  - gather data for use in estimating future test efforts
- How to monitor test progress information?
  - manually using documents, spreadsheets and simple databases
  - using automated tools

Test progress monitoring

- Common metrics for test progress monitoring
  - test case based metrics
  - percentage of work done in test environment preparation
  - test coverage based metrics (requirements, risks, code, configurations or other areas of interest)
  - cost based metrics

Test progress monitoring

Example: Test case summary worksheet

	A	B	C	D	E	F	G	H	I	J	K	L	M
1				System Test Case Summary									
2				Cycle One									
3													
4	Test			System	Bug	Bug	Run	Plan	Act	Plan	Actual	Test	
5	ID	Test Suite/Case	Status	Config	ID	RPN	By	Date	Date	Effort	Effort	Duration	Comment
6													
7	1.000	*Functionality*											
8	1.001	File	Fail	A	701	1	LTW	1/8	1/8	4	6	6	
9	1.002	Edit	Fail	A	709	1	LTW	1/9	1/10	4	8	8	
10					710	5							
11					718	3							
12					722	4							
13	1.003	Font	Pass	B			JHB	1/10	1/10	4	4	4	
14	1.004	Tables	Warn	B	708	15	JHB	1/8	1/9	4	5	5	
15	1.005	Printing	Skip					1/10		4			Out of runway
16		**Suite Summary**						**1/10**	**1/10**	**20**	**23**	**23**	

Reporting test status

- Benefits
  - effectively communicating test results to other project stakeholders
  - support conclusions, recommendations, and decisions
- How to report?
  - IEEE 829 test summary report
  - summaries of the metrics, charts

Reporting test status

- Example

	Unresolved defects		Test cases to be run		
Test targets (product risk areas)	**#**	**%**	**Planned**	**Actual**	**%**
Performance, load, reliability	304	27	3843	1512	39
Robustness, operations, security	234	21	1032	432	42
Functionality, data, dates	224	20	4744	2043	43
Use cases, user interfaces, localization	160	14	498	318	64
Interfaces	93	8	193	153	79
Compatibility	71	6	1787	939	53
Other	21	2	0	0	0
	1107	*100*	*12857*	*5703*	*44*

Test control

- Test control is about **guiding** and **corrective** actions to try to achieve the best possible outcome for the project
- Examples of test control activities
  - reprioritise tests when an identified project risk occurs
  - change the test schedule
  - tighten entry / exit criteria

Contents

Risk

- The possibility of an undesirable outcome
- Determined by
  - likelihood (probability of the risk occurring): between 0% and 100%
  - impact if it did happen
- Two different ways
  - product risks
  - project risks

Product risks

- Factors relating to what is produced by the work, i.e. the thing we are testing
- Possibility that the system or software might fail to satisfy some customer, user, or stakeholder expectation
  - omit some key function
  - unreliable and frequently fail to behave normally
  - cause financial or other damage to a user or the company
  - poor quality characteristic

Project risks

- Factors relating to the way the work is carried out
- Project risks include
  - supplier issues
  - organisational factors
  - technical issues
- What project risks affect testing? - e.g.
  - the late delivery of the test items to the test team
  - availability issues with the test environment
  - excessive delays in repairing defects found in testing

Typical risk-management options

- Mitigation
  - take steps in advance
- Contingency
  - have a plan in place
- Transfer
  - convince some other member of the team or project stakeholder
- Ignore
  - do nothing about the risk

Contents

Incident

- Incident is any event that occurs during testing that requires subsequent investigation or correction
  - actual results do not match expected results (defect)
  - failure of the test environment
  - corrupted test data
  - expected results incorrect
  - tester mistakes
- Incident can be raised for documentation as well as code

Incident report

- The process of incident management
- Goals
  - to **provide programmers, managers and others with detailed information** about the behavior observed and the defect
  - to **provide test leaders with a means of tracking the quality** of the system under test and **the progress of the testing**
  - to **provide ideas for development** and **test process improvements**

What goes in an incident report?

- The outline of a test incident report (IEEE 829)

1. Test incident report identifier
2. Summary
3. Incident description (inputs, expected results, actual results, anomalies, date and time, procedure step, environment, attempts to repeat, testers' and observers' comments)
4. Impact

Severity versus priority

- Severity – the potential impact to the system
  - Mission Critical - Application will not function or system fails
  - Major - Severe problems but possible to work around
  - Minor – Does not impact the functionality or usability of the process but is not according to requirements/design specifications
- Priority – the order in which the incidents are to be addressed
  - Immediate – Must be fixed as soon as possible
  - Delayed – System is usable but incident must be fixed prior to next level of test or shipment
  - Deferred – Defect can be left in if necessary due to time or costs

Incident report life cycle