

CZ3002 - Advanced Software Engineering

Test Plan, Strategy, Techniques and Tools

Faculty : Dr Althea Liang
School : School of Computer Science and Engineering
Email : qhliang@ntu.edu.sg
Office : N4-02c-107

Introduction to this Lesson

At the end of the lesson, you should be able to:

- ▶ Describe the fundamentals of Software Engineering Body of Knowledge (SWEBOK) on software testing
- ▶ Different kinds of software testing
 - ❖ Describe the **test targets** and **test levels** in software testing
 - ❖ Apply the model for the software testing life cycle
 - ❖ Integration and system testing
- ▶ Documentations for software testing
- ▶ List the types of test tools used in software testing



Software Testing - Definition and Scope

- ▶ An activity performed for **evaluating product quality** and for **improving it**, by **identifying defects and problems**
- ▶ Consists of **dynamic verification** of the programme behaviour on a **finite set** of test cases, suitably **selected** from the usually infinite executions domain, against the **expected** behaviour



Software Testing - Objectives

- ▶ Uncover as many as errors (or bugs) as possible in a given timeline
- ▶ Demonstrate that a given software product matches its requirement specifications
- ▶ Validate the quality of a software testing using the minimum cost and effort
- ▶ Generate high quality test cases, perform effective tests, and issue correct and helpful problem reports

► Software Testing Fundamentals:

Test Levels

- Unit Test, Integration Test, System Test

Test Techniques

- Behavioural (White-Box), Structural (Black-Box)

Test-Related Measures

- Fault density, coverage

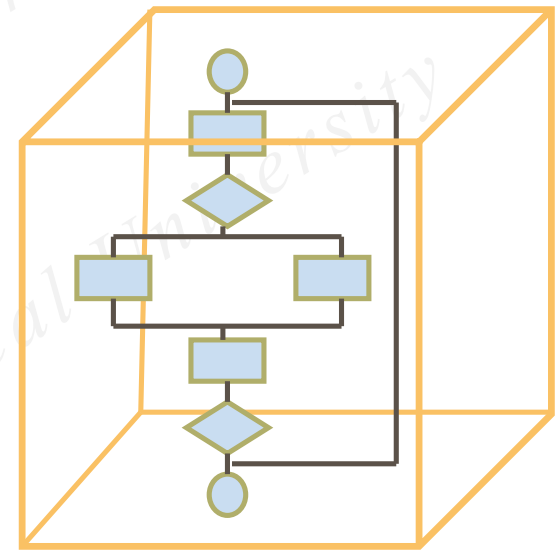
Test Process

- Test process management, documentation

White-Box / Structural Testing

► Tests designed around knowing the internal design structure:

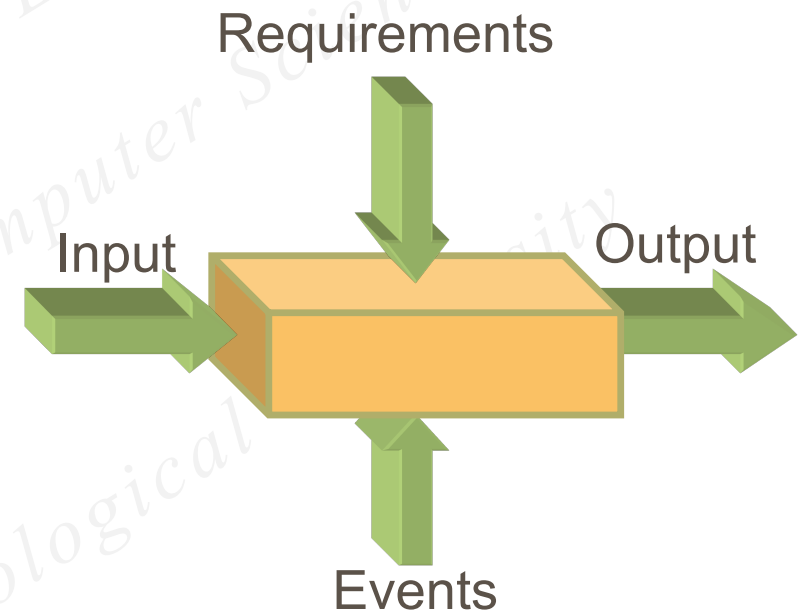
- ❖ Graph-Based Testing
- ❖ Graph Matrices (Cyclomatic Complexity)
- ❖ Structuring Testing
- ❖ Control Flow Testing (Path)
- ❖ Data Flow Testing
- ❖ Slice based Testing
- ❖ Testing Coverage Analysis



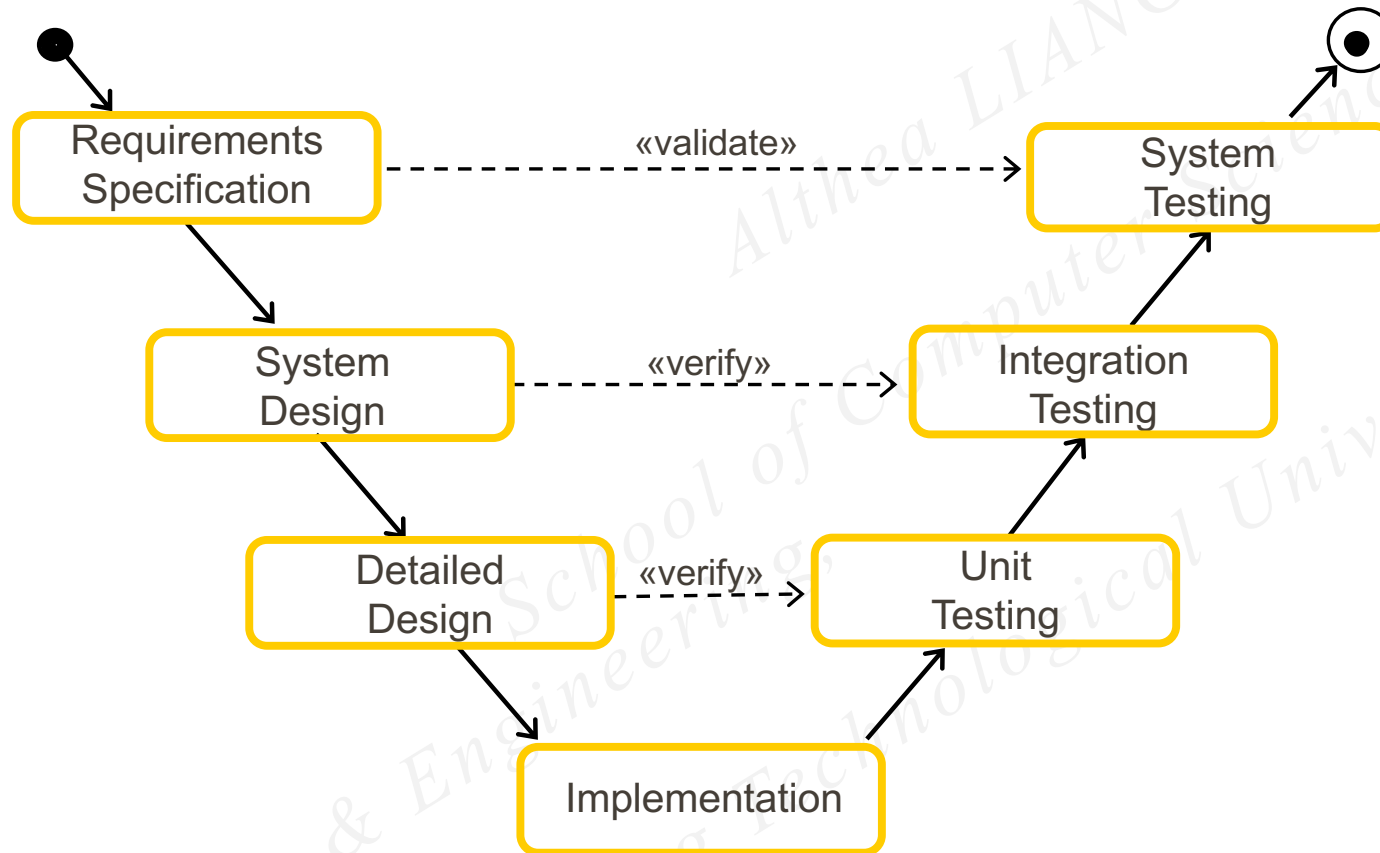
Black Box / Behavioural Testing

► Tests designed around functional requirements

- ❖ Equivalence Partitioning
- ❖ Boundary Value Analysis
- ❖ Decision Table-Based Testing
- ❖ Finite State Machine (FSM)-Based



V-model of Test Levels



Integration Testing

► Decomposition-Based Integration Testing

- ❖ Big-Bang
- ❖ Top-Down
- ❖ Bottom-Up
- ❖ Sandwich
- ❖ Use Case-Driven

Usage-Based Testing

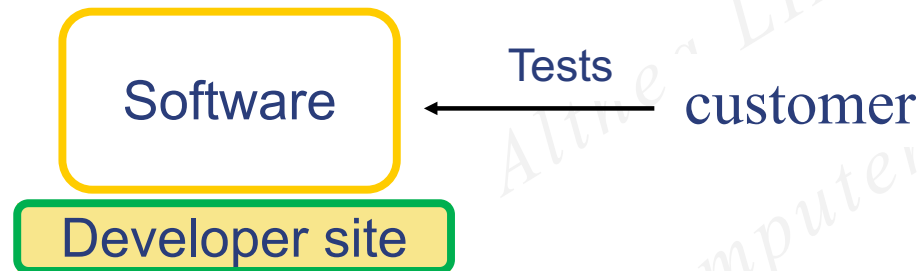
- ▶ Test environment reproduces operational environment of software as closely as possible.
- ▶ Inputs assigned probability distribution (i.e. profile) according to occurrence in actual operation.

System Testing

Types of System Testing	Functions
Functional Testing	Test of functional requirements.
Performance Testing	Test of non-functional requirements.
Pilot Testing	Test of common functionality among selected group of end users in target environment.
Acceptance Testing	Usability, functional and performance tests performed by the customer in development environment against acceptance criteria, operation, contract and regulation.
Installation Testing	Usability, functional and performance tests performed by the Customer in target environment.

Alpha & Beta Test

Alpha Test:



Beta Test:



Performance Testing

Types of Performance Testing	Functions
Stress Testing	Stress limits of system (maximum # of users, peak demands, extended operation)
Volume Testing	Test what happens if large amounts of data are handled
Configuration Testing	Test the various software and hardware configurations
Compatibility Test	Test backward compatibility with existing systems
Security Testing	Try to violate security requirements
Timing Testing	Evaluate response times and time to perform a function
Environmental Test	Test tolerances for heat, humidity, motion, portability
Quality Testing	Test reliability, maintain- ability & availability of the system
Recovery Testing	Tests system's response to presence of errors or loss of data
Human Factors Testing	Tests user interface with user

Stress Testing

- ▶ Find how the system deals with overload
 - ❖ Reason 1: Determine failure behaviour if the load goes above the intended, how “gracefully” does the system fail?
 - ❖ Reason 2: Expose bugs that only occur under heavy loads, especially for OS, middleware, servers, etc.
 - E.g. memory leaks, incorrect resource allocation and scheduling, race conditions

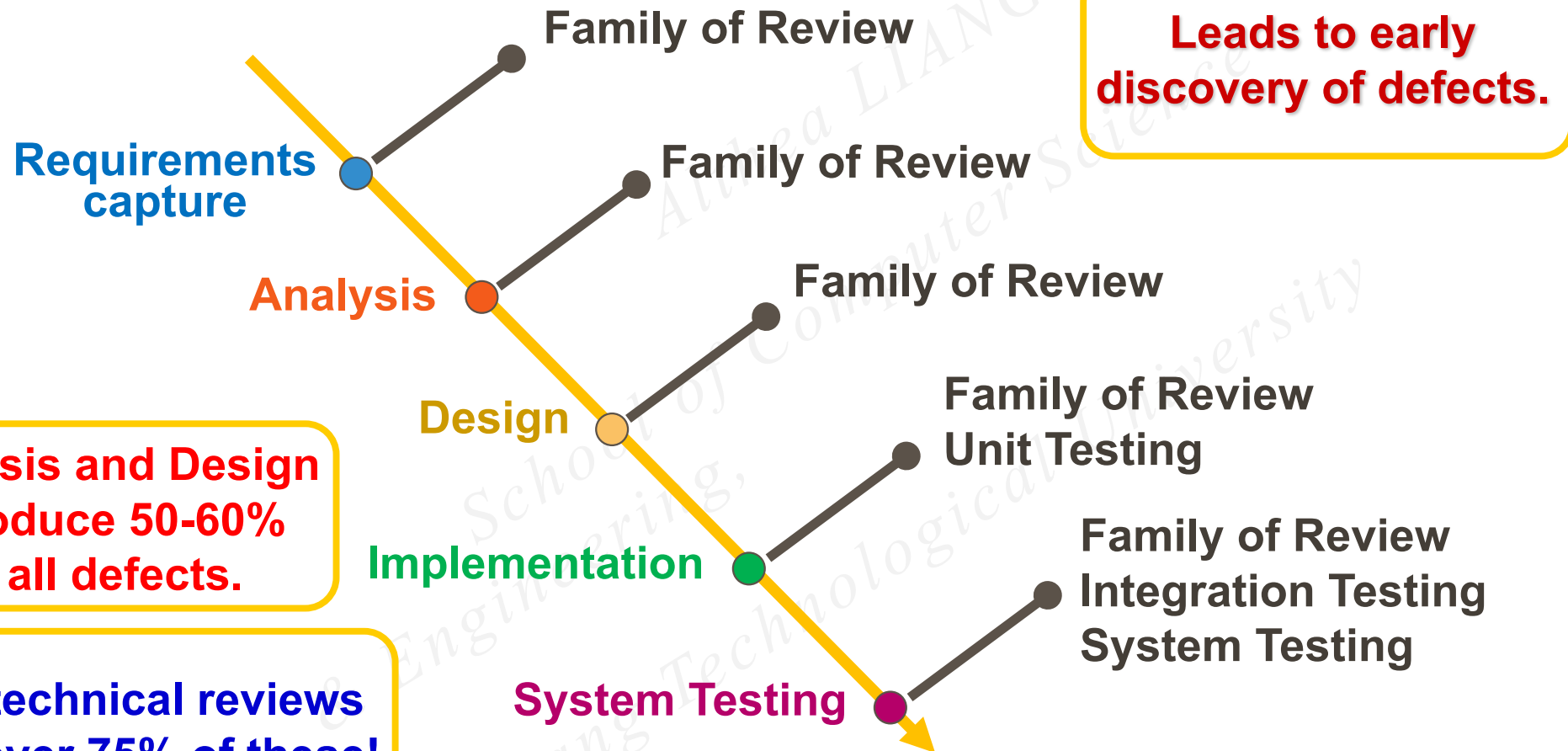
Regression Testing

- ▶ Rerun old tests to see if anything was “broken” by a change
 - ❖ Changes: bug fixes, module integration, maintenance enhancements, etc.
- ▶ Need test automation tools
 - ❖ Load tests, execute them, check correctness
 - ❖ Everything has to be completely automatic
- ▶ Could happen at any time: during initial development or after deployment

Test Strategy – Using Different Techniques for Different Kinds of Faults

	Requirement Faults	Design Faults	Code Faults	Doc Faults
Reviews	Fair	Excellent	Excellent	Good
Prototypes	Good	Fair	Fair	NA
Testing	Poor	Poor	Good	Fair
Correctness Proofs	Poor	Poor	Fair	Fair

Life Cycle of Testing

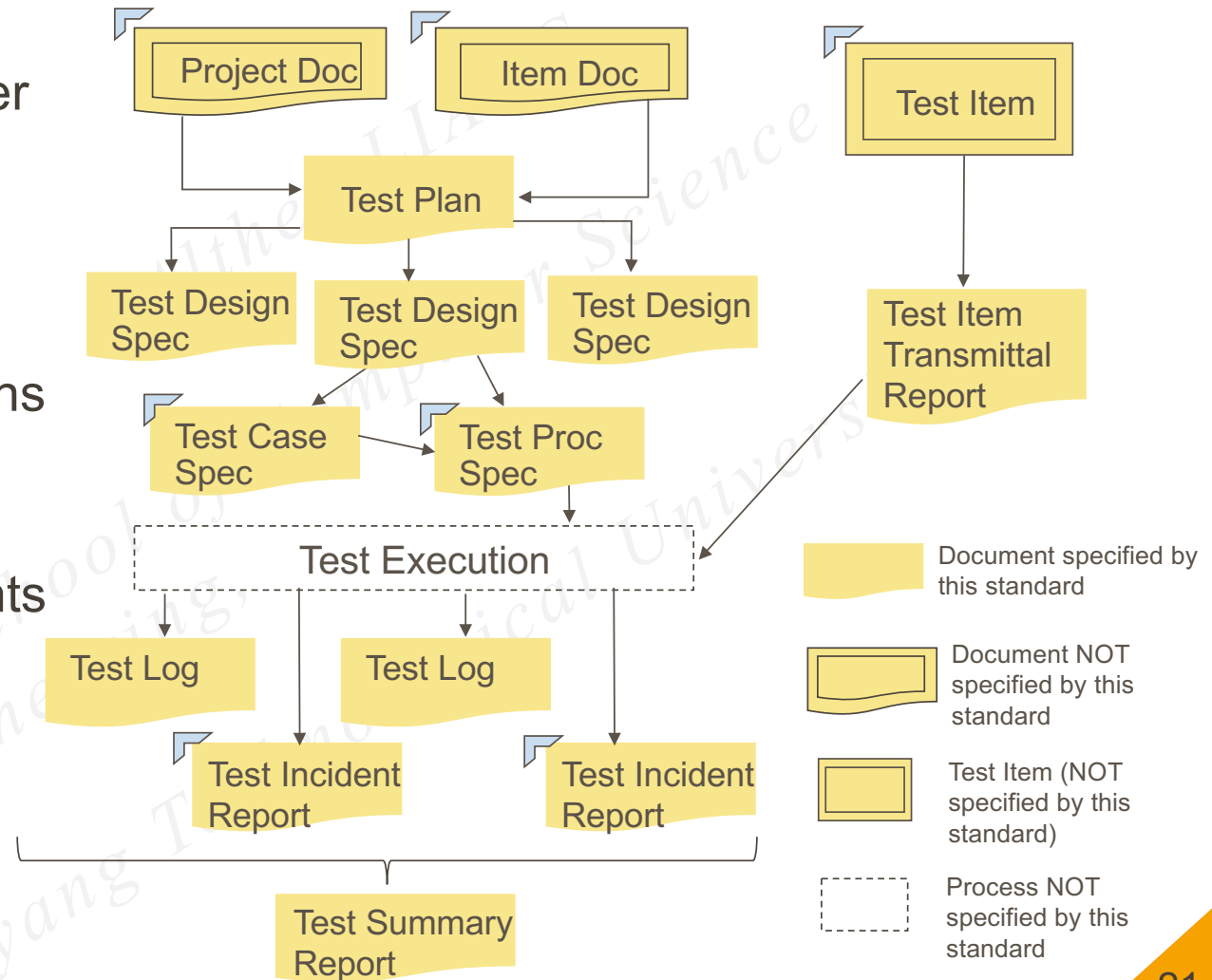


IEEE Standard for Software and System Test Documentation - IEEE 829

1.	Test plan identifier	9.	Test deliverables
2.	Introduction	10.	Testing tasks
3.	Test items	11.	Environmental needs
4.	Features to be tested	12.	Responsibilities
5.	Feature not to be tested	13.	Staffing and training needs
6.	Approach (Strategy / Methods)	14.	Schedule
7.	Item pass/fail criteria	15.	Risks and contingencies
8.	Suspension criteria and resumption requirements	16.	Approvals

Test Case Specification

1. Test case specification identifier
2. Test items
3. Input specifications
4. (Expected) Output specifications
5. Environmental needs
6. Special procedural requirements
7. Inter-case dependencies



Test Tools

- ▶ Test management tools
- ▶ Static analysis tools (e.g., LOC; complexity;...)
- ▶ Test evaluation tools (e.g., code coverage)
- ▶ GUI test drivers & capture/replay tools
- ▶ Load & performance tools
- ▶ Automated test tools

Summary

Now, you should be able to:

- ▶ Describe the fundamentals of Software Engineering Body of Knowledge (SWEBOK) on software testing
- ▶ Different kinds of software testing
 - ❖ Describe the **test targets** and **test levels** in software testing
 - ❖ Apply the model for the software testing life cycle
 - ❖ Integration and system testing
- ▶ Documentations for software testing
- ▶ List the types of test tools used in software testing

Special Thanks to Kydon during the TEL Efforts of the Lecture

End of Software Maintenance

Faculty : Dr Althea Liang
School : School of Computer Science and Engineering
Email : qhliang@ntu.edu.sg
Office : N4-02c-107