Introduction to Software Testing

Speaker: Jerry Gao Ph.D.

Computer Engineering Department
San Jose State University

email: jerry.gao@sjsu.edu
URL: http://www.engr.sjsu.edu/gaojerry

Presentation Outline

- *What is Software Testing?*
  - *Definitions*
  - *Testing Objectives*
  - *Who Does Software Testing?*
- *Software Testing Activities*
- *Software Testing Scope*
- *Software Testing Principles*
- *Software Testing Process*
- *Software Testing Myths*
- *Software Testing Limits*
- *Different Types of Software Testing*

What is Software Testing

Several definitions:

"Testing is the process of establishing confidence that a program or system does what it is supposed to." by Hetzel 1973

"Testing is the process of executing a program or system with the intent of finding errors." by Myers 1979

"Testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results."

by Hetzel 1983

What is IEEE's definition?

What is Software Testing

- One of very important software development phases

- A software process based on well-defined software quality control and testing standards, testing methods, strategy, test criteria, and tools.

- Engineers perform all types of software testing activities to perform a software
test process.

- The last quality checking point for software on its production line

Testing Objectives

The Major Objectives of Software Testing:

- Uncover as many as errors (or bugs) as possible in a given timeline.

- Demonstrate a given software product matching its requirement specifications.

- Validate the quality of a software testing using the minimum cost and efforts.

- Generate high quality test cases, perform effective tests, and issue correct and helpful problem reports.

Major goals:

uncover the errors (defects) in the software, including errors in:

- **requirements from requirement analysis**
- **design documented in design specifications**
- **coding (implementation)**
- **system resources and system environment**
- **hardware problems and their interfaces to software**

Who does Software Testing

- Test manager
  - manage and control a software test project
  - supervise test engineers
  - define and specify a test plan

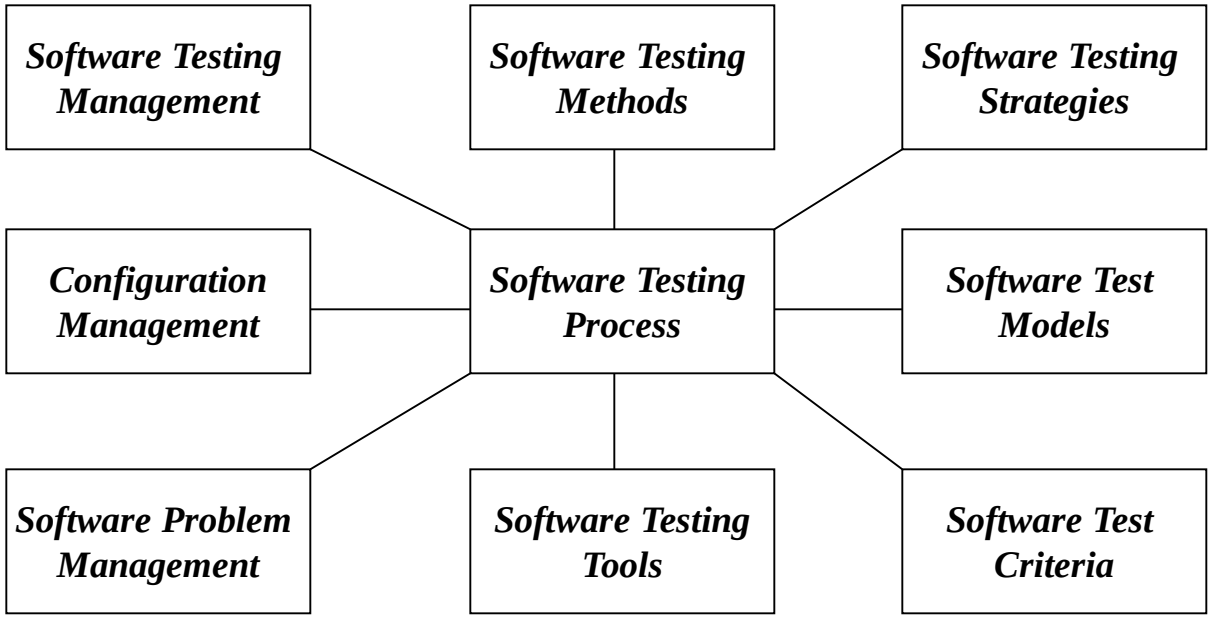
- Software Test Engineers and Testers
  - define test cases, write test specifications, run tests

- Independent Test Group

- Development Engineers
  - Only perform unit tests and integration tests

- Quality Assurance Group and Engineers
  - Perform system testing
  - Define software testing standards and quality control process

Software Testing Scope



Software Testing Activities

- Test Planing
  Define a software test plan by specifying:
    - a test schedule for a test process and its activities, as well as assignments
    - test requirements and items
    - test strategy and supporting tools

- Test Design and Specification
    - Conduct software design based well-defined test generation methods.
    - Specify test cases to achieve a targeted test coverage.

- Test Set up:
    - Testing Tools and Environment Set-up
    - Test Suite Set-up

- Test Operation and Execution
    - Run test cases manually or automatically

Software Testing Activities

- **Test Result Analysis and Reporting**
  Report software testing results and conduct test result analysis

- **Problem Reporting**
  Report program errors using a systematic solution.

- **Test Management and Measurement**
  Manage software testing activities, control testing schedule, measure testing complexity and cost

- **Test Automation**
  - **Define and develop software test tools**
  - **Adopt and use software test tools**
  - **Write software test scripts and facility**

- **Test Configuration Management**
  - **Manage and maintain different versions of software test suites, test environment and tools, and documents for various product versions.**

Verification and Validation

Software testing is one element of a broader topic that is often referred to as
===> Verification and Validation (V&V)

Verification --> refers to the set of activities that ensure that software correctly implements a specific function.

Validation -> refers to a different set of activities that ensure that the software hat has been built is traceable to customer requirements.

Boehm [BOE81]:

Verification: "Are we building the product right?"
Validation: "Are we building the right product?"

The definition of V&V encompasses many of SQA activities, including formal technical reviews, quality and configuration audits performance monitoring, different types of software testing feasibility study and simulation

Software Quality Factors

Functionality (exterior quality)

- Correctness, reliability, usability, and integrity

Engineering (interior quality)

- Efficiency, testability, documentation, structure

Adaptability (future qualities)

- Flexibility, reusability, maintainability

Software Testing Principles

- ***Principle #1: Complete testing is impossible.***

- ***Principle #2: Software testing is not simple.***
  - ***Reasons:***
    - ***Quality testing requires testers to understand a system/product completely***
    - ***Quality testing needs adequate test set, and efficient testing methods***
    - ***A very tight schedule and lack of test tools.***

- ***Principle #3: Testing is risk-based.***

- ***Principle #4: Testing must be planned.***

- ***Principle #5: Testing requires independence.***

- ***Principle #6: Quality software testing depends on:***
  - ***Good understanding of software products and related domain application***
  - ***Cost-effective testing methodology, coverage, test methods, and tools.***
  - ***Good engineers with creativity, and solid software testing experience***

Software Testing Myths

- We can test a program completely. In other words, we test a program exhaustively.

- We can find all program errors as long as test engineers do a good job.

- We can test a program by trying all possible inputs and states of a program.

- A good test suite must include a great number of test cases.

- Good test cases always are complicated ones.

- Software test automation can replace test engineers to perform good software testing.

- Software testing is simple and easy. Anyone can do it. No training is needed.

Software Testing Limits

- Due to the testing time limit, it is impossible to achieve total confidence.

- We can never be sure the specifications are 100% correct.

- We can never be certain that a testing system (or tool) is correct.

- No testing tools can copy with every software program.

- Tester engineers never be sure that they completely understand a software product.

- We never have enough resources to perform software testing.

- We can never be certain that we achieve 100% adequate software testing.

Software Testing Process and Types

Speaker: Jerry Gao Ph.D.

Computer Engineering Department
San Jose State University

email: jerry.gao@sjsu.edu
URL: http://www.engr.sjsu.edu/gaojerry

Presentation Outline

- What is Software Testing Process?
- Software Unit Testing
- Software Integration Testing
- Software Function Validation
- Software System Testing

What is Software Testing Process?

Testing is a process rather than a single activity. This process starts from test planning then designing test cases, preparing for execution and evaluating status till the test closure. So, we can divide the activities within the fundamental test process into the following basic steps:

1) Planning and Control
2) Analysis and Design
3) Implementation and Execution
4) Evaluating exit criteria and Reporting
5) Test Closure activities

Software Test Planning?

Test planning has following major tasks:

i. To determine the scope and risks and identify the objectives of testing.

ii. To determine the test approach.

iii. To implement the test policy and/or the test strategy.

iv. To determine the required test resources like people, test environments, PCs, etc.

v. To schedule test analysis and design tasks, test implementation, execution and evaluation.

vi. To determine the Exit criteria we need to set criteria such as Coverage criteria.

Software Test Control

Test control has the following major tasks:

i. To measure and analyze the results of reviews and testing.

ii. To monitor and document progress, test coverage and exit criteria.

iii. To provide information on testing.

iv. To initiate corrective actions.

v. To make decisions.

Software Test Analysis and Design

Test analysis and Test Design has the following major tasks:

i. To review the test basis.

ii. To identify test conditions.

iii. To design the tests.

iv. To evaluate testability of the requirements and system.

v. To design the test environment set-up and identify and required infrastructure and tools.

Software Test Implementation and Execution

During test implementation and execution, we take the test conditions into test cases and procedures and other test-ware such as scripts for automation, the test environment and any other test infrastructure.

Test implementation has the following major task:

a) To develop and prioritize test cases and test data by using well-defined test methods.

- Define some operation instructions (known as procedures)

- May automate some tests using test harness and tests scripts.

b) To create test suites from the test cases for efficient test execution.

c) To implement and verify the environment.

Software Test Implementation and Execution

a) To execute test suites and individual test cases by following procedures

b) To re-execute the tests that previously failed in order to confirm a fix. This is known as confirmation testing or re-testing.

c) To log the outcome of the test execution and record the identities and versions of the software under tests. The test log is used for the audit trial.

c) To Compare actual results with expected results.

d) To report the problems if there are differences between actual and expected results.

Evaluating Exit criteria and Reporting

The criteria for each level is set up based on the risk assessment of the project. The criteria may vary from project to project.

Exit criteria come into picture, when:
– Maximum test cases are executed with certain pass percentage.
– Bug rate falls below certain level.
– When achieved the deadlines.

Evaluating exit criteria has the following major tasks:

i. To check the test logs against the exit criteria specified in test planning.
ii. To assess if more test are needed or if the exit criteria specified should be changed.
iii. To write a test summary report for stakeholders

Software Test Closure Activities

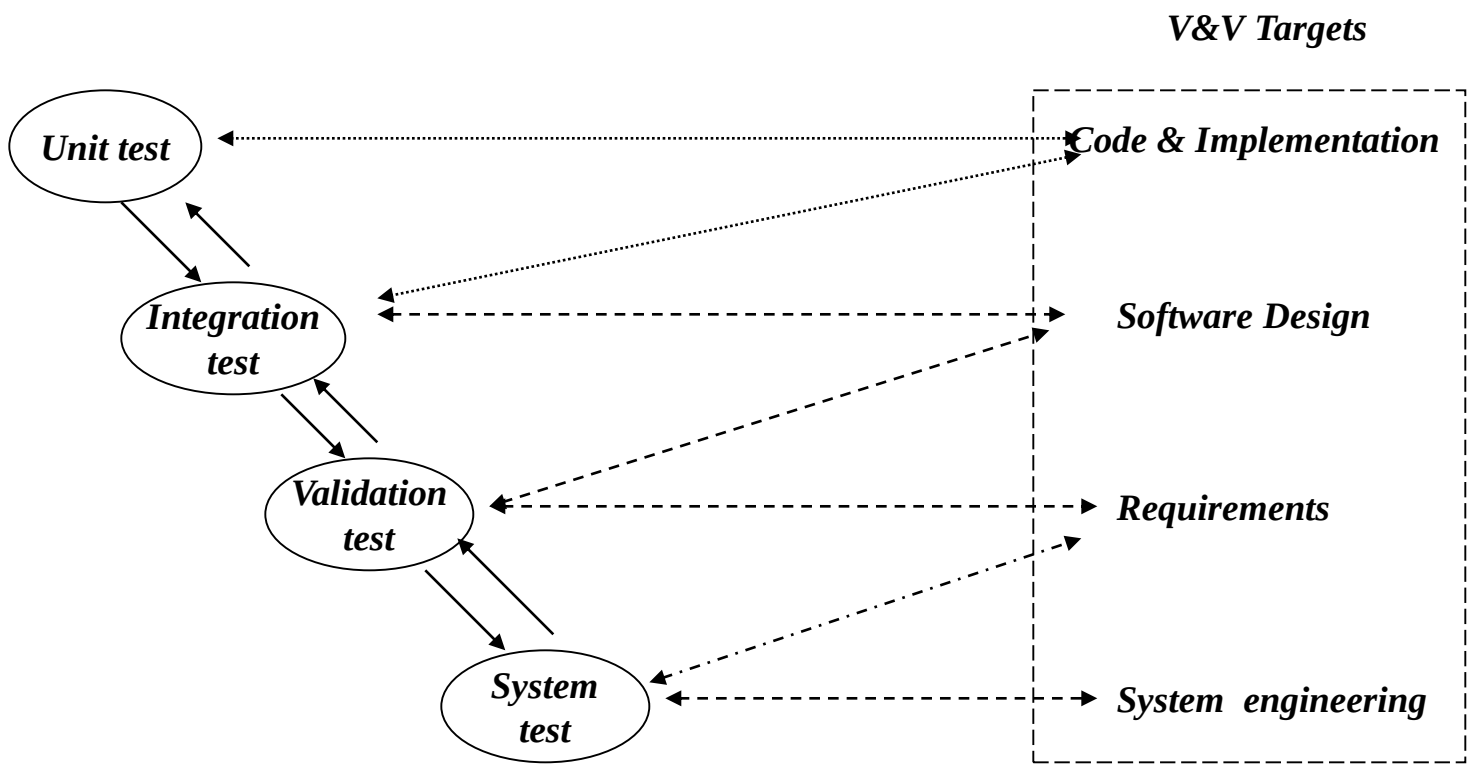
Test closure activities are done when software is delivered. The testing can be closed for the other reasons also like:

- **When all the information has been gathered which are needed for the testing.**
- **When a project is cancelled.**
- **When some target is achieved.**
- **When a maintenance release or update is done.**

Test closure activities include the following major tasks:

i. To check which planned deliverables are actually delivered and to ensure that all incident reports have been resolved.
ii. To finalize and archive test-ware such as scripts, test environments, etc. for later reuse.
iii. To handover the test-ware to the maintenance organization. They will give support to the software.
iv To evaluate how the testing went and learn lessons for future releases and projects.

Software Testing Process

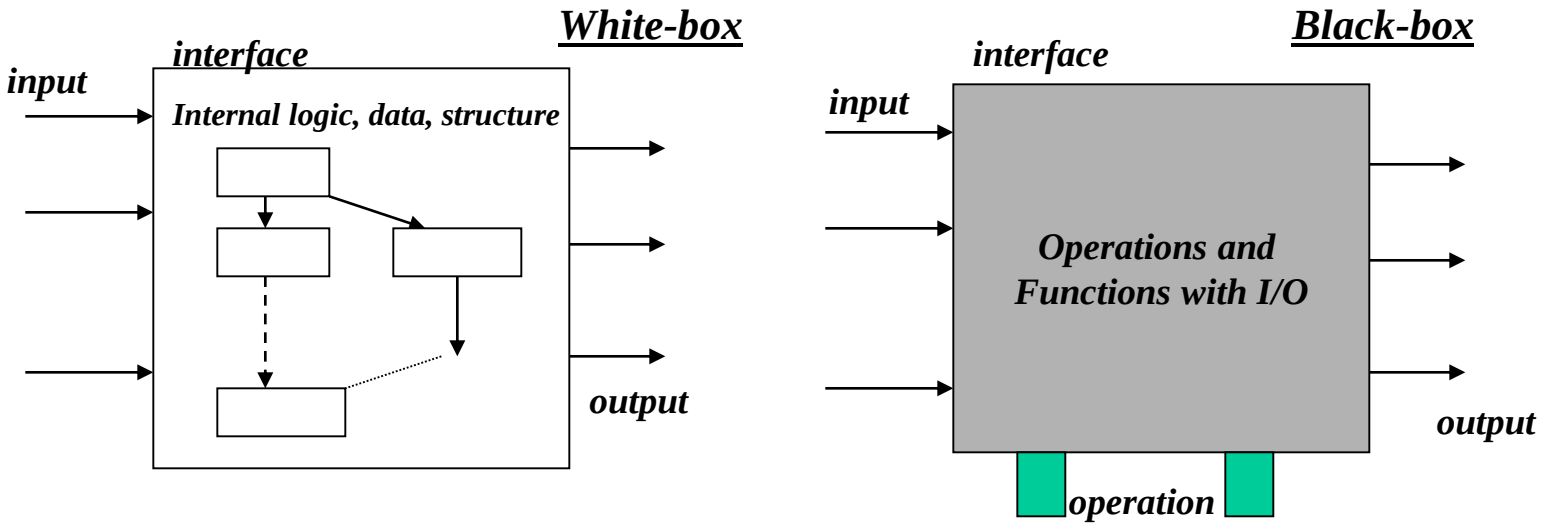


Unit Test (Component Level Test)

Unit testing: Individual components are tested independently to ensure their quality. The focus is to uncover errors in design and implementation, including

- ***data structure in a component***
- ***program logic and program structure in a component***
- ***component interface***
- ***functions and operations of a component***

Unit testers: developers of the components.



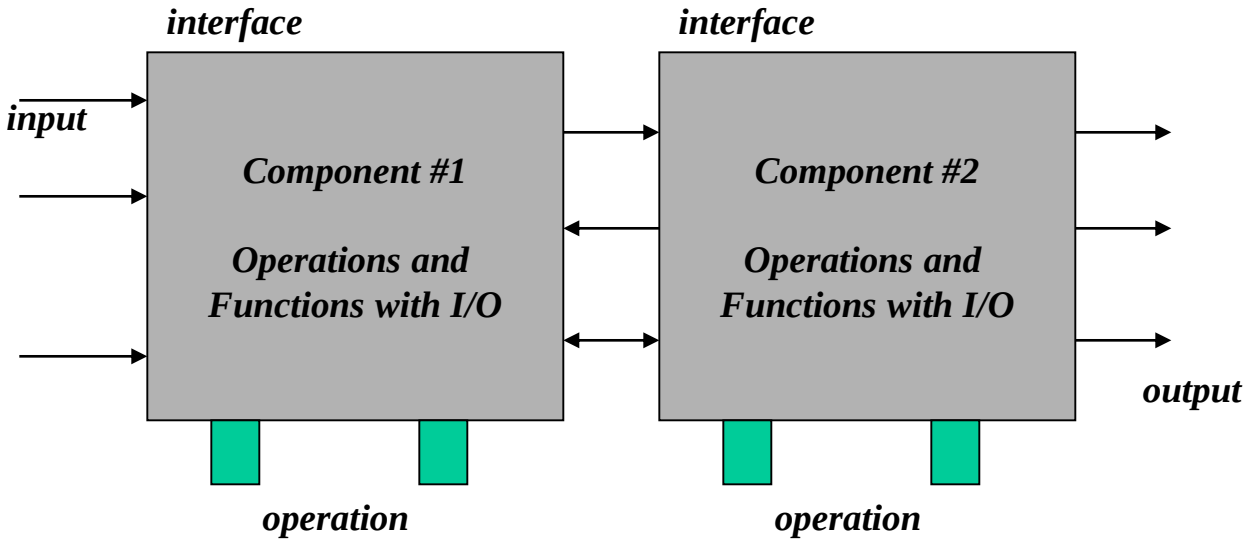
Integration Testing

Integration test: ***A group of dependent components are tested together to ensure their the quality of their integration unit.***

The focus is to uncover errors in:

- ***Design and construction of software architecture***
- ***Integrated functions or operations at sub-system level***
- ***Interfaces and interactions between them***
- ***Resource integration and/or environment integration***

Integration testers: either developers and/or test engineers.



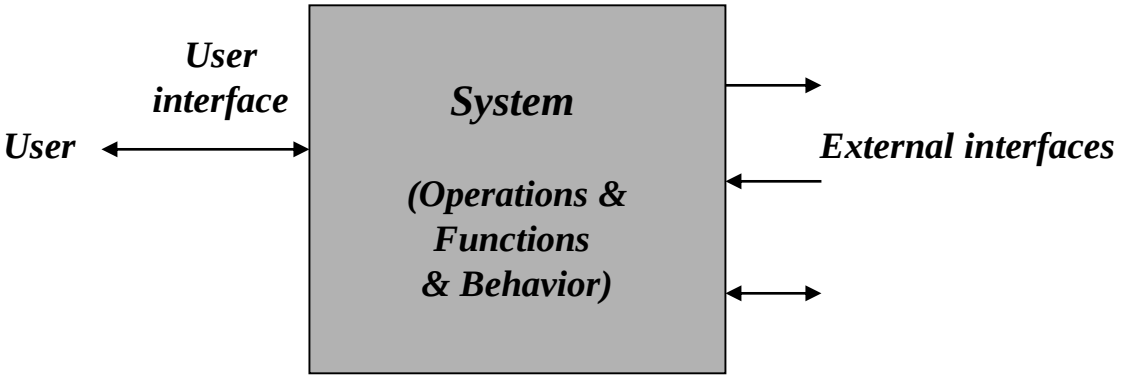
Function Validation Testing

Validation test: ***The integrated software is tested based on requirements to ensure that we have a right product.***

The focus is to uncover errors in:

- ***System input/output***
- ***System functions and information data***
- ***System interfaces with external parts***
- ***User interfaces***
- ***System behavior and performance***

Validation testers: ***test engineers in ITG or SQA people.***



System Testing

System test: *The system software is tested as a whole. It verifies all elements mesh properly to make sure that all system functions and performance are achieved in the target environment.*

The focus areas are:

- *System functions and performance*
- *System reliability and recoverability (recovery test)*
- *System installation (installation test)*
- *System behavior in the special conditions (stress and load test)*
- *System user operations (acceptance test/alpha test)*
- *Hardware and software integration and collaboration*
- *Integration of external software and the system*

System testers: *test engineers in ITG or SQA people.*

When a system is to be marketed as a software product, a testing process called beta testing is often used.

Current Test Issues and Challenges

Software testing is very expensive.

How to achieve test automation?

When should we stop software testing?

Test criteria, test coverage, adequate testing.

Other software testing challenges?

- ***GUI Testing***
- ***Testing Components and Component-Based Software***
- ***Testing Web-based Systems***
- ***Testing Mobile APPs and Mobile Web Applications***
- ***Testing System Security***
- ***Testing SaaS Applications***

Topic #1 – What is Software Testing?

Instructor: Jerry Gao, Ph.D., Professor
San Jose State University

TOPIC #1 – WHAT IS SOFTWARE TESTING?

Validation vs. verification

- Definitions about Software Testing
- Software Testing Objectives and Focuses
- Validation vs. Verification
- Software Testing Scope

Topic #1 – What is Software Testing?

Definitions:

"Testing is the process of establishing confidence that a program or system does what it is supposed to." (by Hetzel 1973)

"Testing is the process of executing a program or system with the intent of finding errors." (by Myers 1979)

"Testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results." (by Hetzel 1983)

TOPIC #1 – WHAT IS SOFTWARE TESTING?

IEEE Definitions:

"Software testing is the process of analyzing a software item to detect the differences between existing and required conditions (that is, bugs) and to evaluate the features of the software item".

"Reliability is the ability of a system or component to perform its required functions under stated conditions for a specified period of time."

Topic #1 – What is Software Testing?

Verification: Are we building the product right?

Verification: the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase

Validation: Are we building the right product?

Validation: process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements

TOPIC #1 – WHAT IS SOFTWARE TESTING?

Major Objectives:

1 **Requirement Confirmation**

Software Product **Requirements**

2 **Bug Detection**

3 **Product Quality Assurance**

Software Product **Bug Reports**

TOPIC #1 – WHAT IS SOFTWARE TESTING?

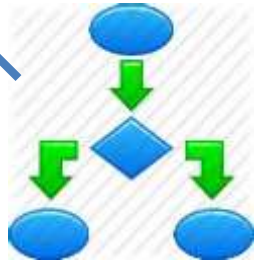
Primary Software Testing Focuses:

Check Program Functions

Check Program Behaviors

Check System

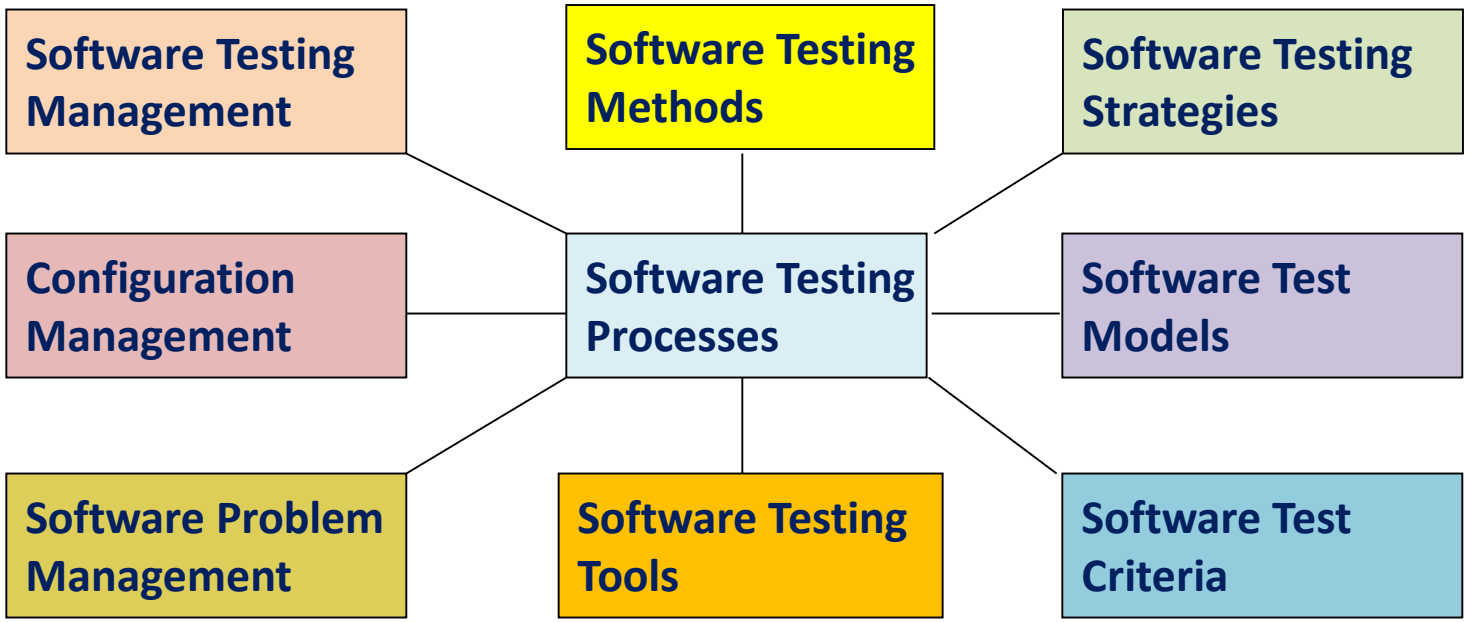
- Reliability
- Availability
- Performance



Check Program Logics/structures

TOPIC #1 – WHAT IS SOFTWARE TESTING?

Software Testing Scope:



Topic #2 –
Why Do We Need Software Testing?

Instructor: Jerry Gao, Ph.D., Professor
San Jose State University

Topic #2 – Why Do We Need Software Testing?

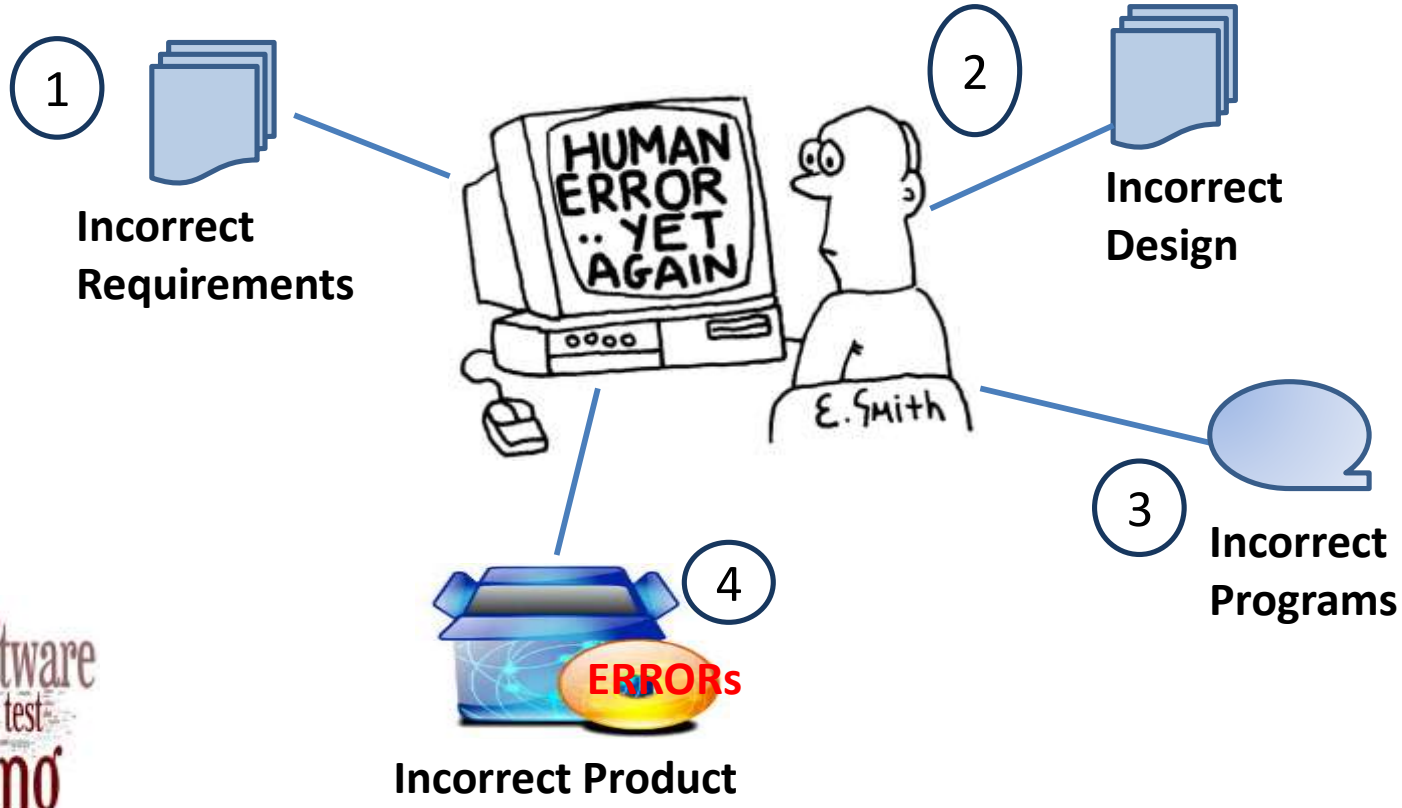
Why Do We Need Software Testing?

Basic Needs for Quality Software Testing

Software Testing Limitations

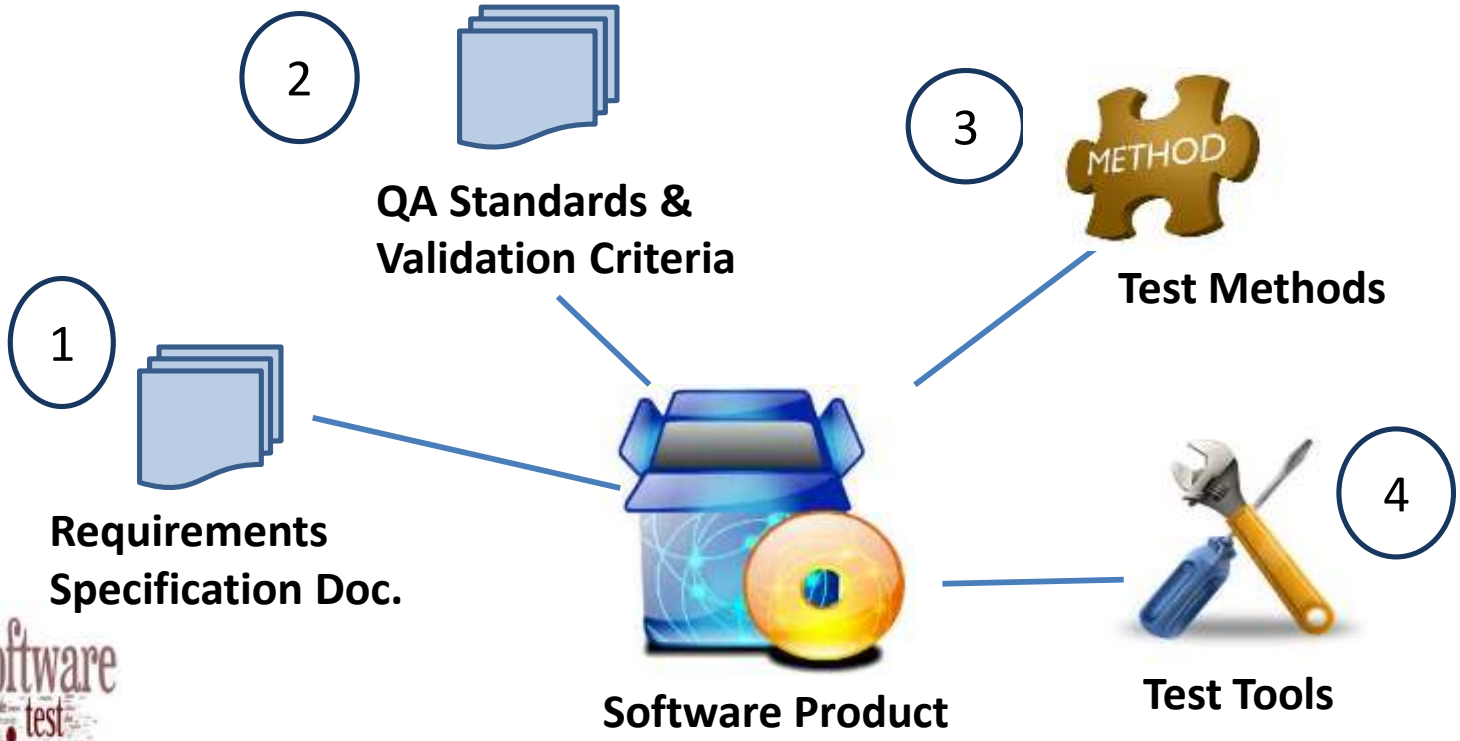
TOPIC #2 – WHY DO WE NEED SOFTWARE TESTING?

Why Do We Need Software Testing?



TOPIC #2 – WHY DO WE NEED SOFTWARE TESTING?

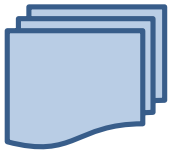
Basic Needs for Quality Software Testing



TOPIC #2 – WHY DO WE NEED SOFTWARE TESTING?

Software Testing Limitations

2

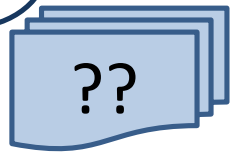


Engineer Limitations

3

Test Tool Limitations

1



Requirement Correctness and Completeness ??

4

Budget & Schedule

Topic #3 – Software Test Process

Instructor: Jerry Gao, Ph.D., Professor
San Jose State University

Topic #3 – Software Testing Process

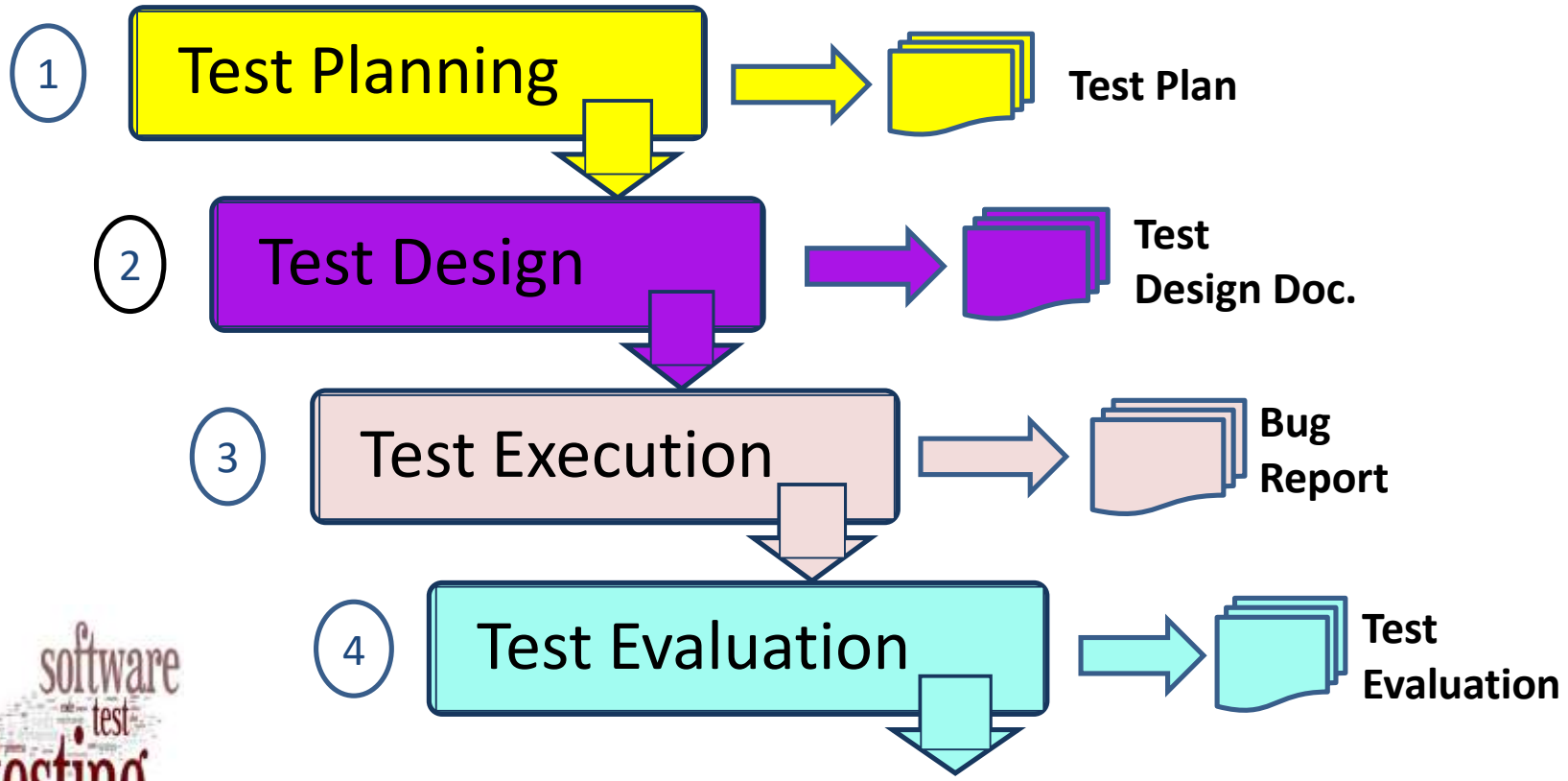
A Common Software Test Process

Who Does Software Testing?

Different Types of Software Testing

TOPIC #3 – SOFTWARE TESTING PROCESS

A Common Software Test Process:



TOPIC #3 – SOFTWARE TESTING PROCESS

Who Does Software Testing?

Test Engineer

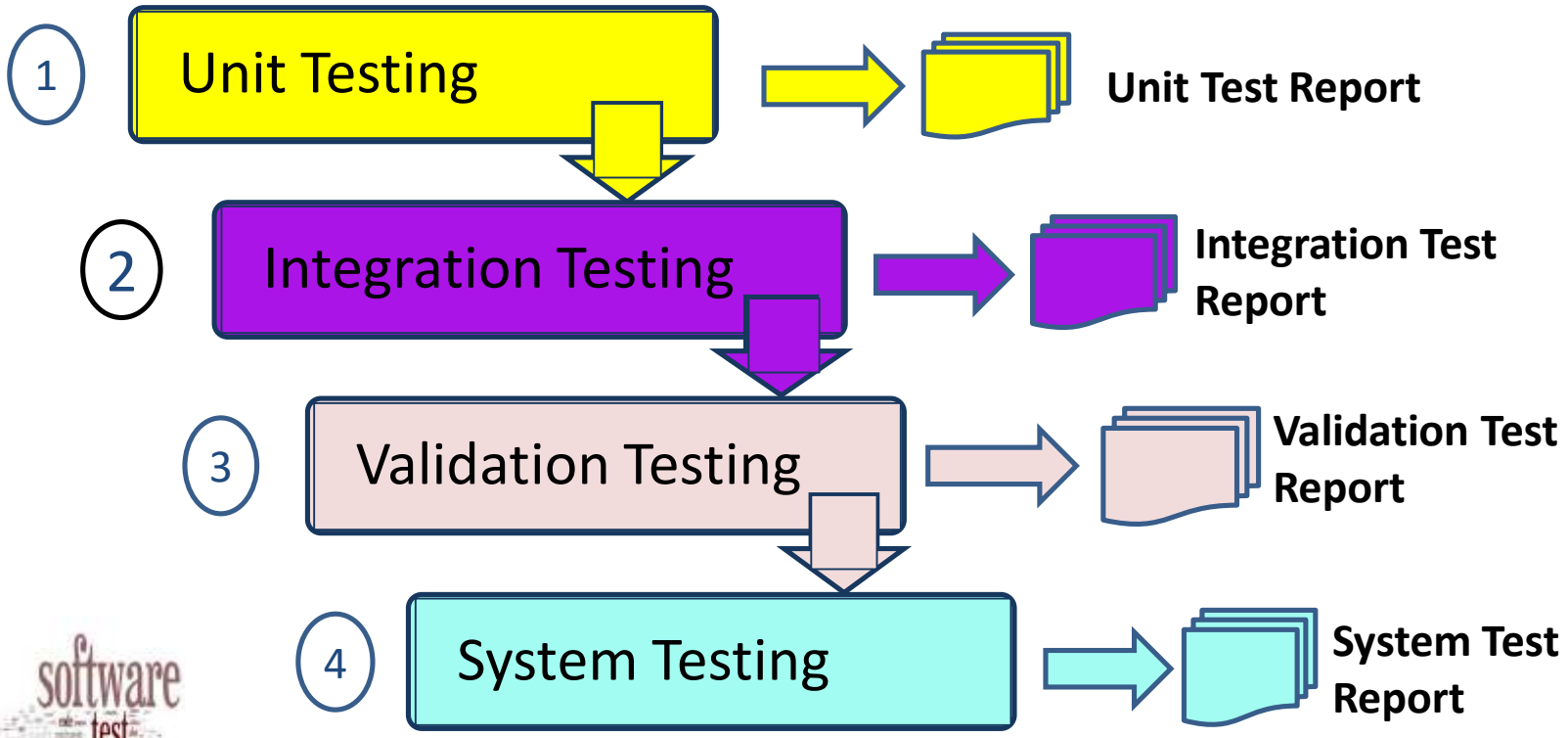
Quality Assurance Engineer

Test Manager

Developer

TOPIC #3 – SOFTWARE TESTING PROCESS

A Software Product Test Life Cycle:



TOPIC #3 – SOFTWARE TESTING PROCESS

Unit Testing

Black-Box Testing

Software Component

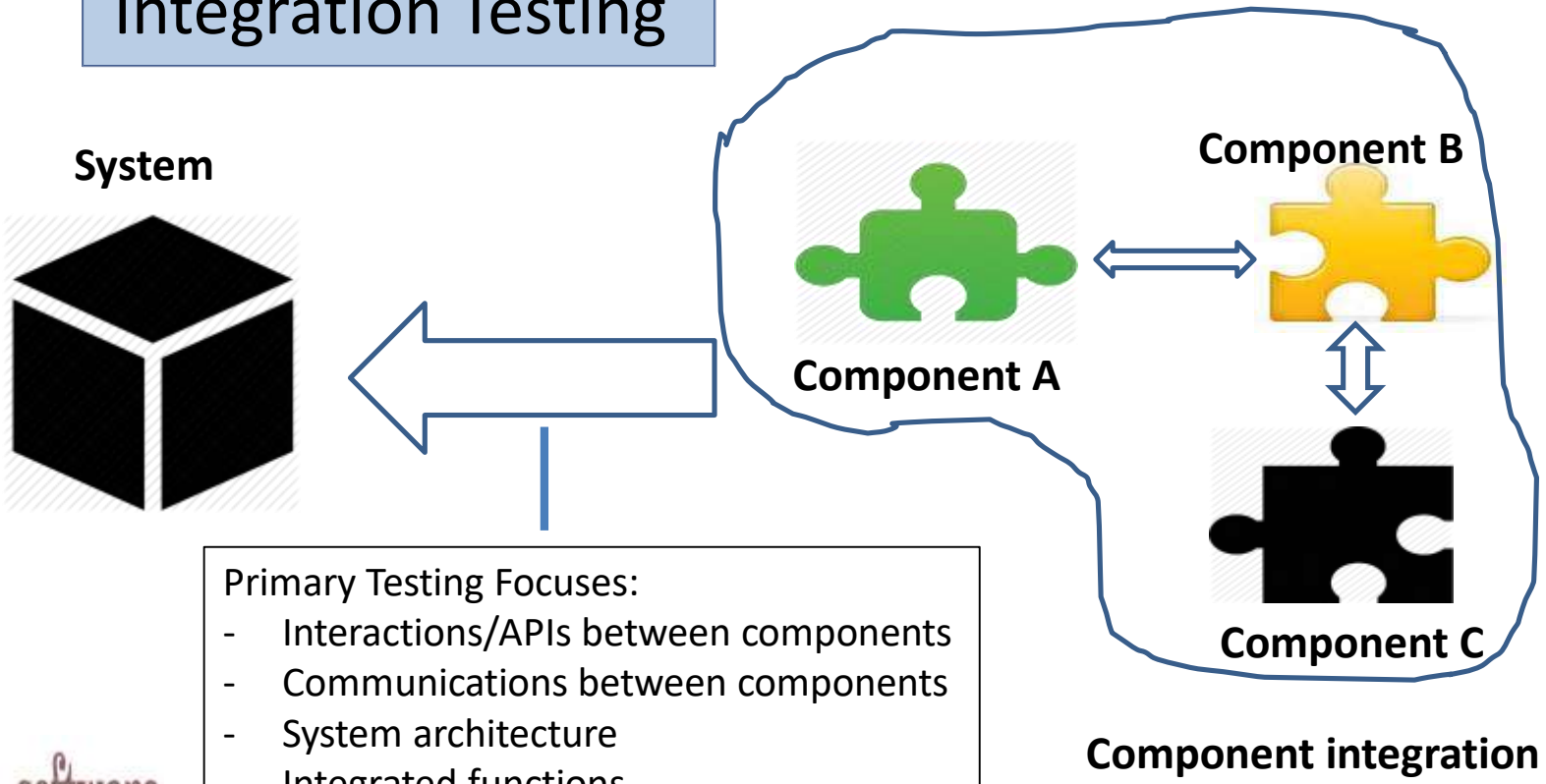
Program Logics & Structures

White-Box Testing

Primary Testing Focuses:
- Component logics and structures
- Component black-box interfaces
- Component external functions
- Component external behaviors

TOPIC #3 – SOFTWARE TESTING PROCESS

Integration Testing

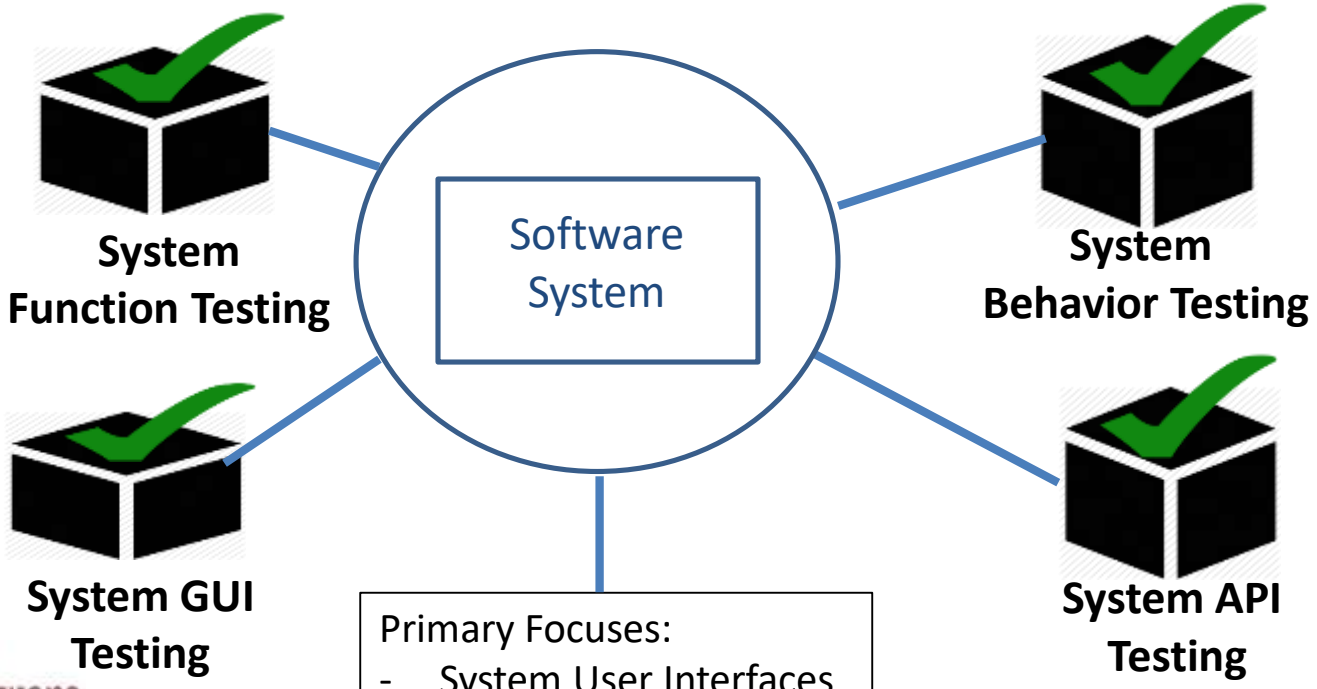


Primary Testing Focuses:

- Interactions/APIs between components
- Communications between components
- System architecture
- Integrated functions
- Integrated system structure

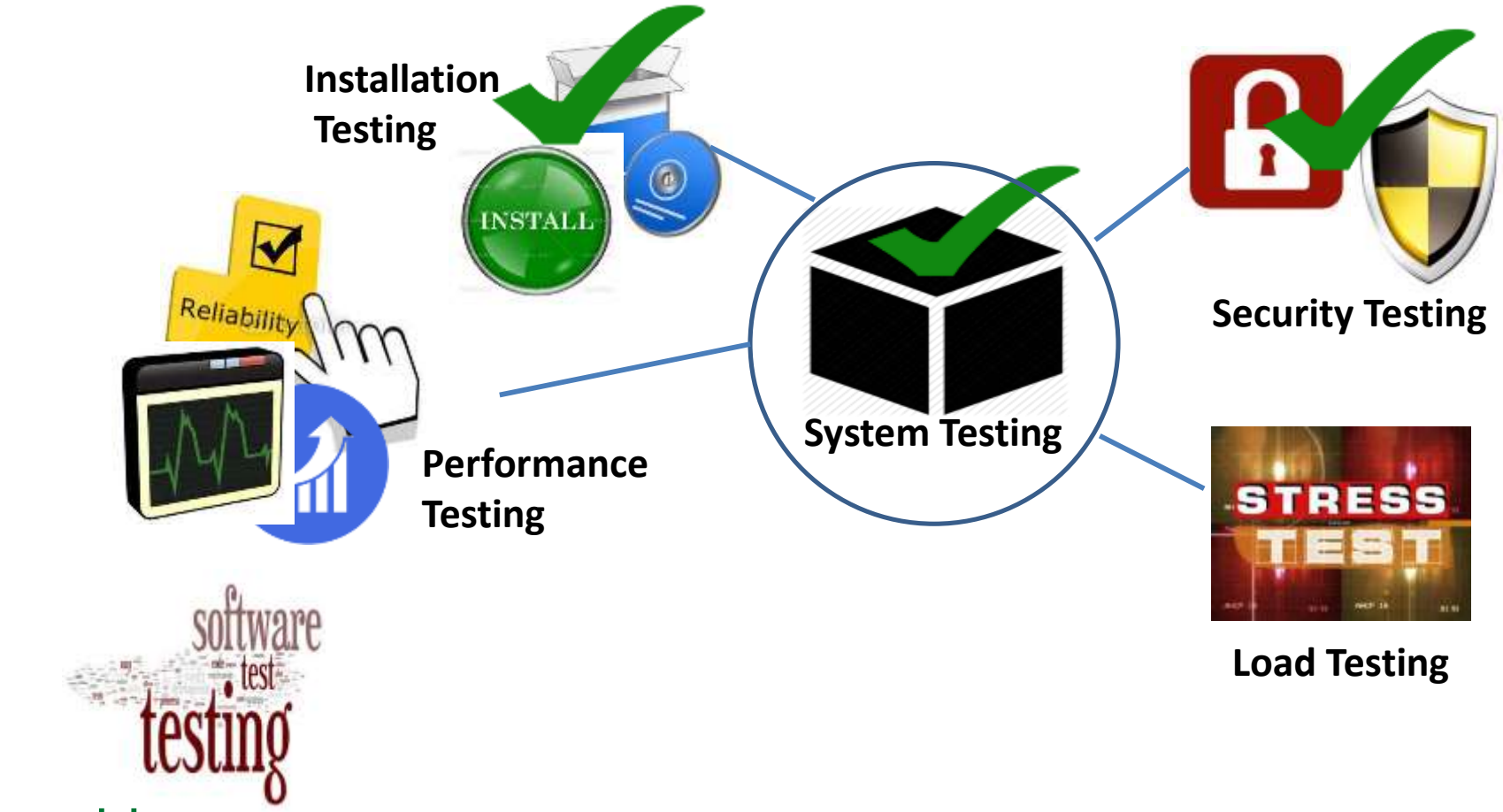
TOPIC #3 – SOFTWARE TESTING PROCESS

System Function Validation



TOPIC #3 – SOFTWARE TESTING PROCESS

System Testing



Topic #4 – Software Test Activities

Instructor: Jerry Gao, Ph.D., Professor
San Jose State University

Topic #4 – Software Testing Activities

- Test Planning Activities
- Test Design Activities
- Test Execution Activities
- Test Evaluation Activities
- Test Management Activities

TOPIC #4 – SOFTWARE TESTING ACTIVITIES

Software Test Planning Activities:

1 **Test Planning** → **Test Plan**

To determine test scope, risks and identify the objectives of testing

To determine the test approach.

To determine QA and testing standards, and coverage policy

To determine tasks and schedules

To determine the required resources for software testing

Software Test Design Activities:

2 **Test Design** → **Test Design**

Target at software testing requirements for under-test product

To use well-defined Test methods

To design and track software tests

Understand the pre-conditions post-conditions for each test

To design and determine software test environment and infrastructures

TOPIC #4 – SOFTWARE TESTING ACTIVITIES

Software Test Execution Activities:

3 **Test Execution** → **Test Results**

To select test strategy and test sets

To execute tests

To determine test procedure

To record test results

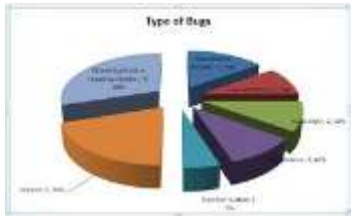
To automate tests using scripting

To record bugs for under-test software

TOPIC #4 – SOFTWARE TESTING ACTIVITIES

Software Test Evaluation Activities:

4 Test Evaluation → **Test Evaluation Report**



Bug analysis

Test analysis

Quality analysis

Cost analysis

Test review

TOPIC #4 – SOFTWARE TESTING ACTIVITIES

Software Test Management Activities:

5 Test Management → Test Management Report

Problem Management

Schedule Management

Quality Management

Cost Management

Project Management