# Introduction to Software Verification, Validation and Testing

## Overview

Ira A. Fulton Schools of Engineering
Arizona State University

# Overview

# Objectives

**Objective**

Define common testing terminology

# Objectives

### Objective

Define common testing terminology

### Objective

Describe how testing is integrated into software development phases

# Objectives

**Objective**

Define common testing terminology

**Objective**

Describe how testing is integrated into software development phases

**Objective**

Define the objectives of the different levels of testing

# Objectives

### Objective

Define common testing terminology

### Objective

Describe how testing is integrated into software development phases

### Objective

Define the objectives of the different levels of testing

### Objective

Explain best practices for software testing

# Summary

# Introduction to Software Verification, Validation and Testing

## Testing Background

Ira A. Fulton Schools of Engineering
Arizona State University

# Objective

### Objective

Define common testing terminology

# Testing Background

# History

- **The term software engineering was first used at a workshop in West Germany in 1968 considering the growing problems of software development**
  - High Cost
  - Difficult to Manage
  - Poor Reliability
  - Lack of User Acceptance
  - Difficult to Maintain

# Current State of Software Development

- High Cost
- Difficult to manage
- Poor Reliability
- Lack of User Acceptance
- Difficult to Maintain

# Poor Reliability

Software defects rates are around 1 delivered defect per thousand lines of code

With applications spanning millions of lines of code, customers experience many defects

# Definitions

## Reliability:

- The probability that a software program operates for some given time period without software error

# Testers vs Pollsters Analogy

# Definitions

## Validation:

- Are we building the right product?

## Verification:

- Are we building the product right?

## Testing:

- The examination of the behavior of the program by executing it on sample data sets

# Definitions

## Error:

- Mistake made by a human

## Defect/Fault:

- Result of error manifested in the code

## Failure:

- Software doesn’t do what it is supposed to do

# Summary

# Introduction to Software Verification, Validation and Testing

## Testing Throughout Life Cycle

Ira A. Fulton Schools of Engineering
Arizona State University

# Objectives

### Objective

Describe how testing is integrated into software development phases

### Objective

Define the objectives of the different levels of testing

# Testing Throughout Life Cycle

- Waterfall
- Agile
- TDD

- Agile Methodology: The Complete Guide to Understanding Agile Testing

# Agile Testing

## Continuous Integration (at least daily)

- Static Code Analysis
- Compile
- Unit Test
- Deploy into Test Environment
- Integration / Regression Test

# Test Driven Development (Red, Green, Refactor Cycle)

## Red Phase:

- Write a minimal test on the behavior needed

## Green Phase:

- Write only enough code to make the failing test pass

## Refactor Phase:

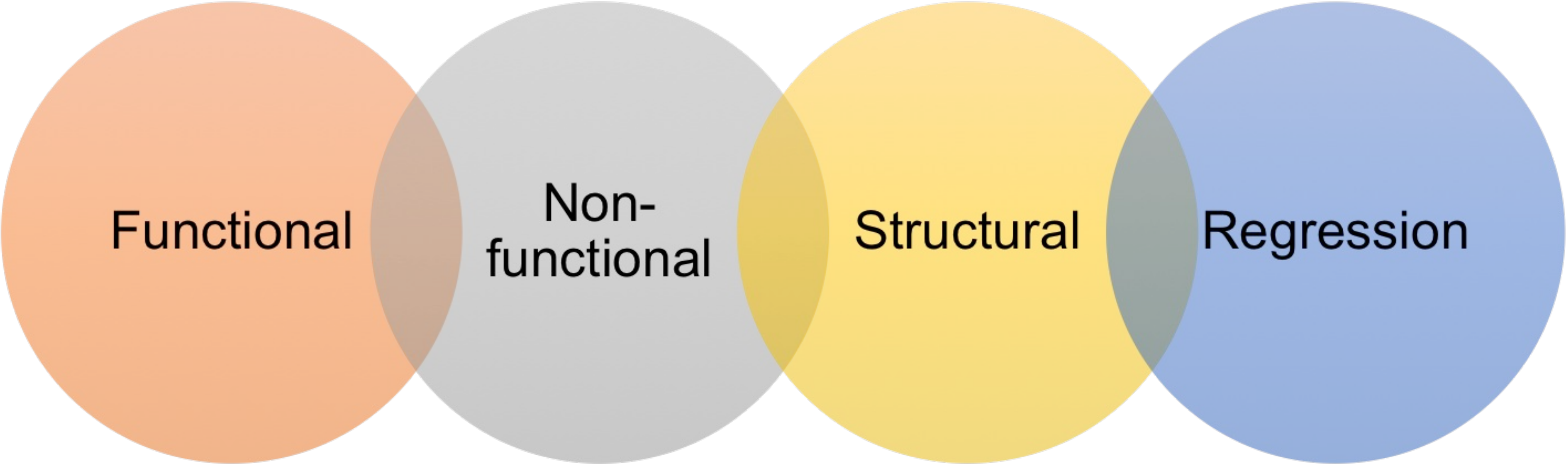
- Improve code while keeping tests green

# Software Development Process vs Test Development Process

# Testing Levels

- Unit / Component
- Integration
- System
- Acceptance
- Beta

# Test Types


Functional

Non-functional

Structural

Regression

# Summary

# Introduction to Software Verification, Validation and Testing

## Testing Principles and Best Practices

Ira A. Fulton Schools of Engineering
Arizona State University

# Objective

### Objective

Explain best
practices for
software testing.

# Testing Principles

## Principle 1:

- Testing only shows the presence of defects – not proof of correctness

## Principle 2:

- Exhaustive testing is impossible

## Principle 3:

- Start testing early

# Testing Principles

## Principle 1:

- Testing only shows the presence of defects – not proof of correctness

## Principle 2:

- Exhaustive testing is impossible

## Principle 3:

- Start testing early

## Principle 4:

- Defects cluster

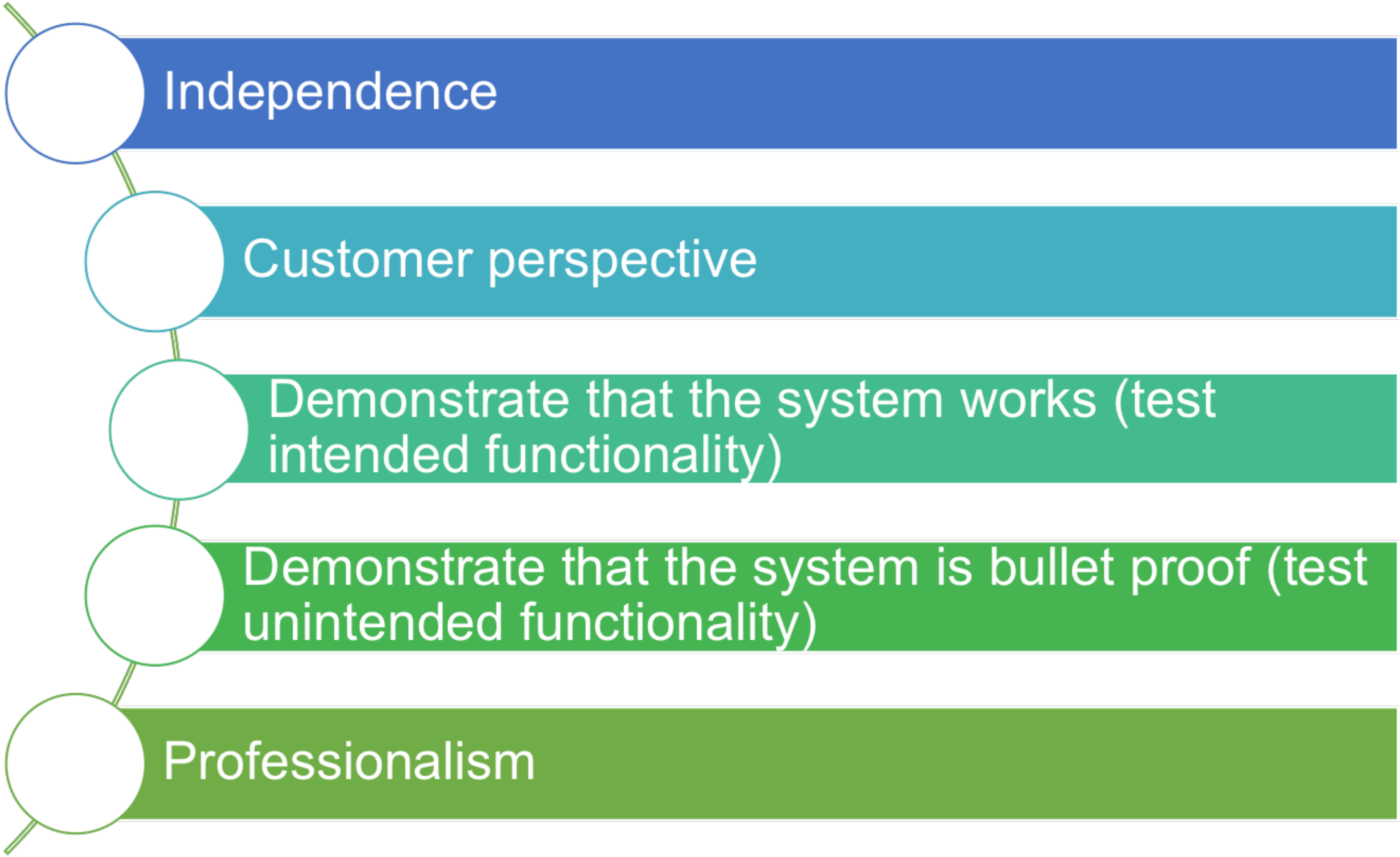
## Principle 5:

- Testing is context dependent

## Principle 6:

- Absence-of-errors fallacy

# Testing Attitude



- Independence
- Customer perspective
- Demonstrate that the system works (test intended functionality)
- Demonstrate that the system is bullet proof (test unintended functionality)
- Professionalism

# Some Classic Testing Mistakes

- **Believing the primary objective of system testing is to find bugs**
  - Test must concentrate on finding important problems
  - Test must provide an estimate of system quality

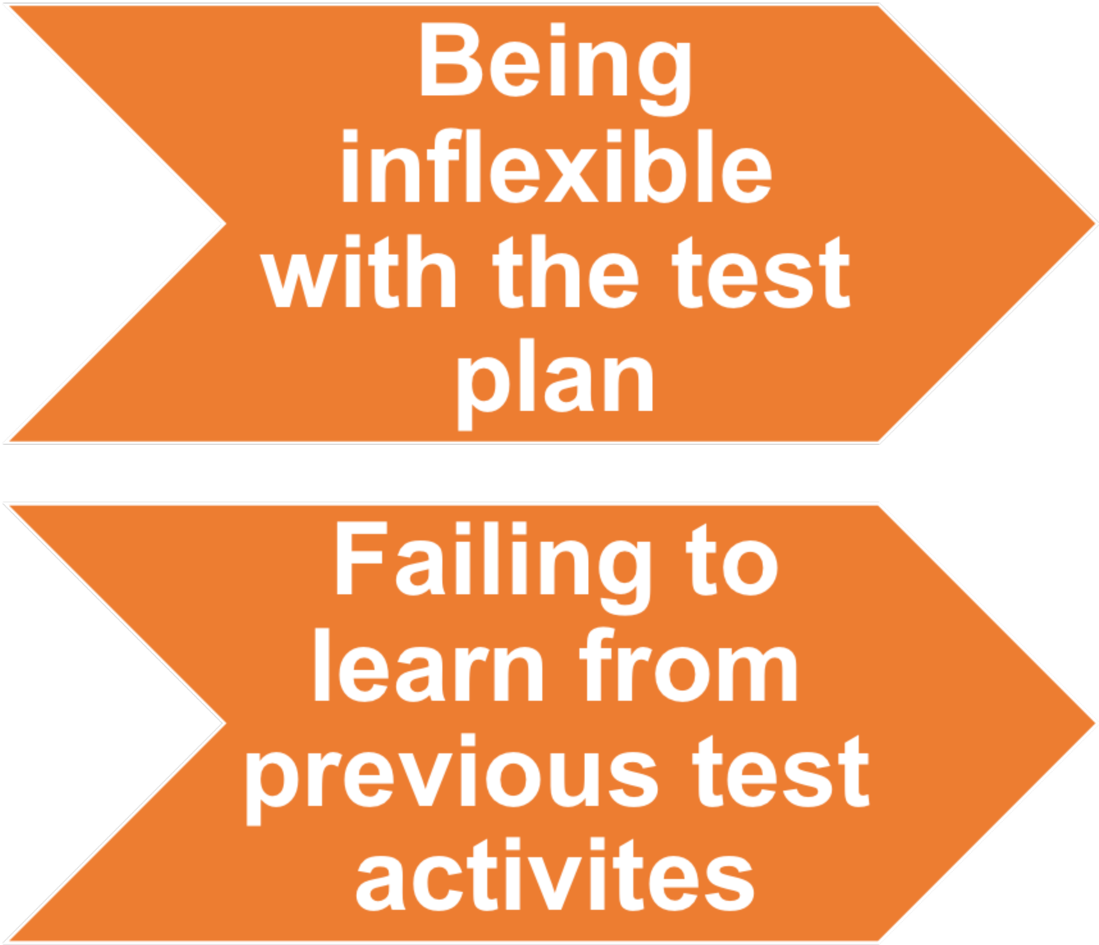
- **Not focusing on usability issues**

- **Starting too late**
  - Test must help development avoid problems

# Some Classic Testing Mistakes

- Delaying stress and performance testing until the end
- Not testing the documentation
- Not staffing the test team with domain experts
- Not communicating well with developers
- Failing to adequately document and review test designs

# Some Classic Testing Mistakes



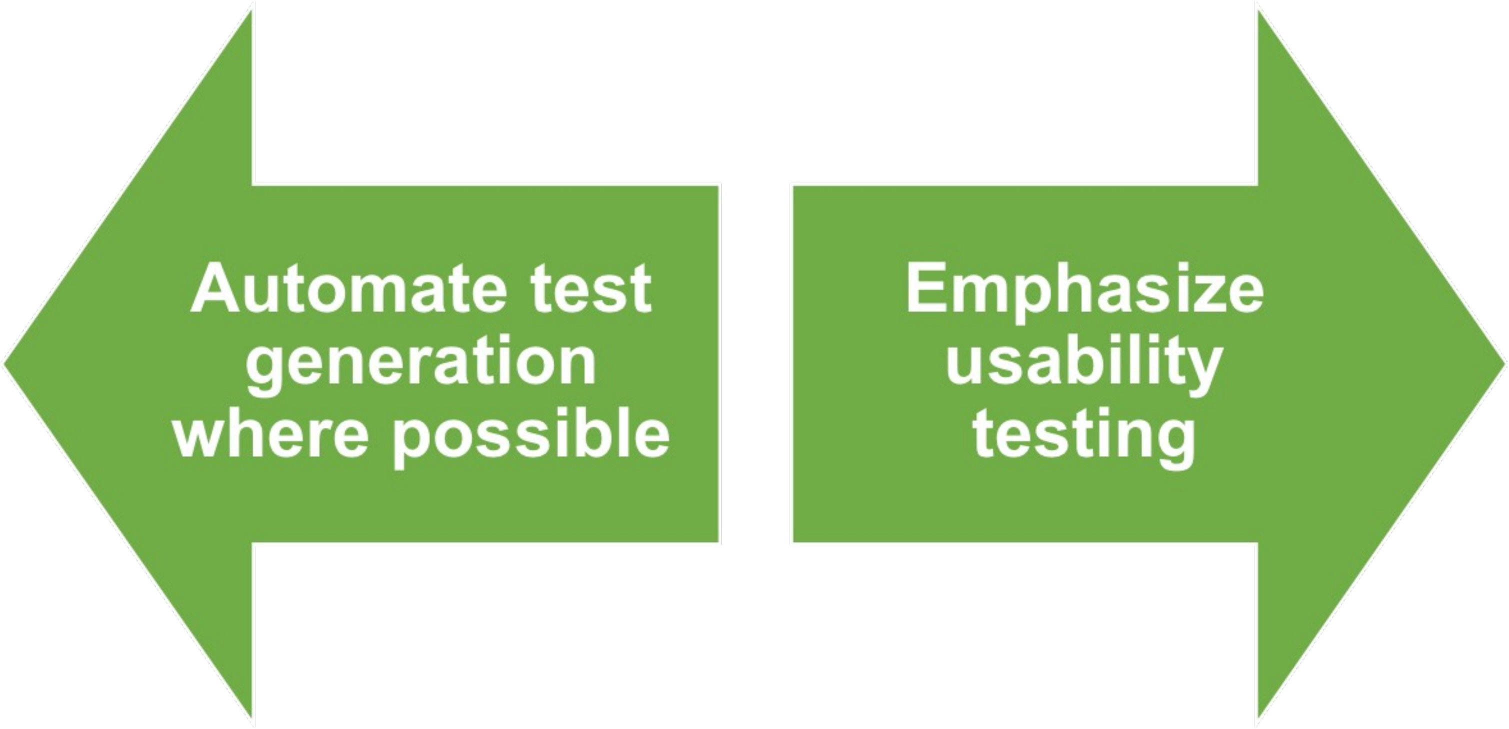
Being inflexible with the test plan

Failing to learn from previous test activites

# Best Testing Practices

- Assess software reliability via statistical testing
- Develop an agile test design
  - Accommodate late changes
  - Emphasis on regression testing
- Utilize model-based testing techniques
  - State diagrams
- Develop cross-functional development and test teams

# Best Testing Practices

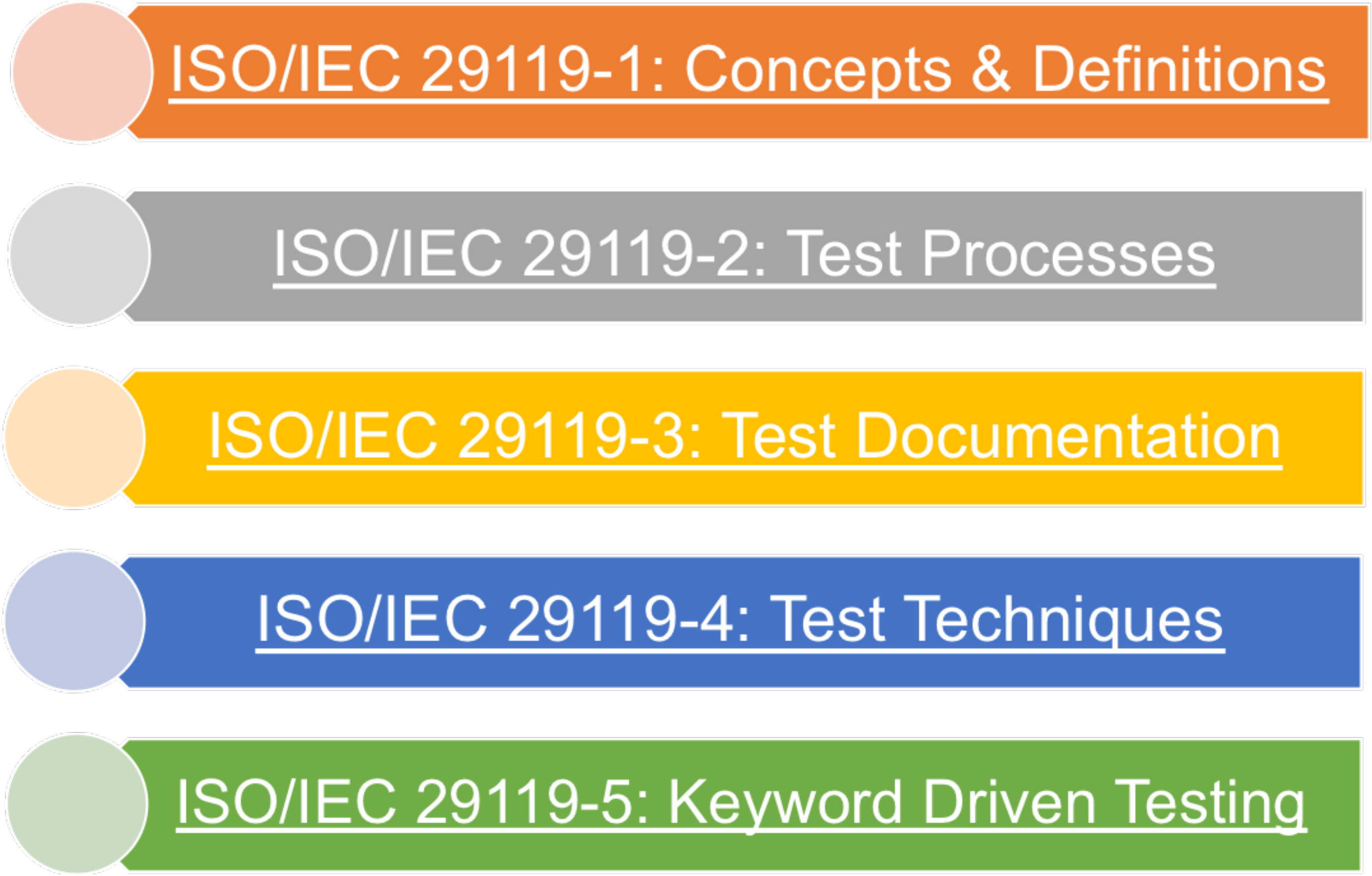


Automate test generation where possible

Emphasize usability testing

# ISO/IEC/IEEE 29119 Software Testing

**Consists of 5 standards applicable within any life cycle or organization**



- ISO/IEC 29119-1: Concepts & Definitions
- ISO/IEC 29119-2: Test Processes
- ISO/IEC 29119-3: Test Documentation
- ISO/IEC 29119-4: Test Techniques
- ISO/IEC 29119-5: Keyword Driven Testing

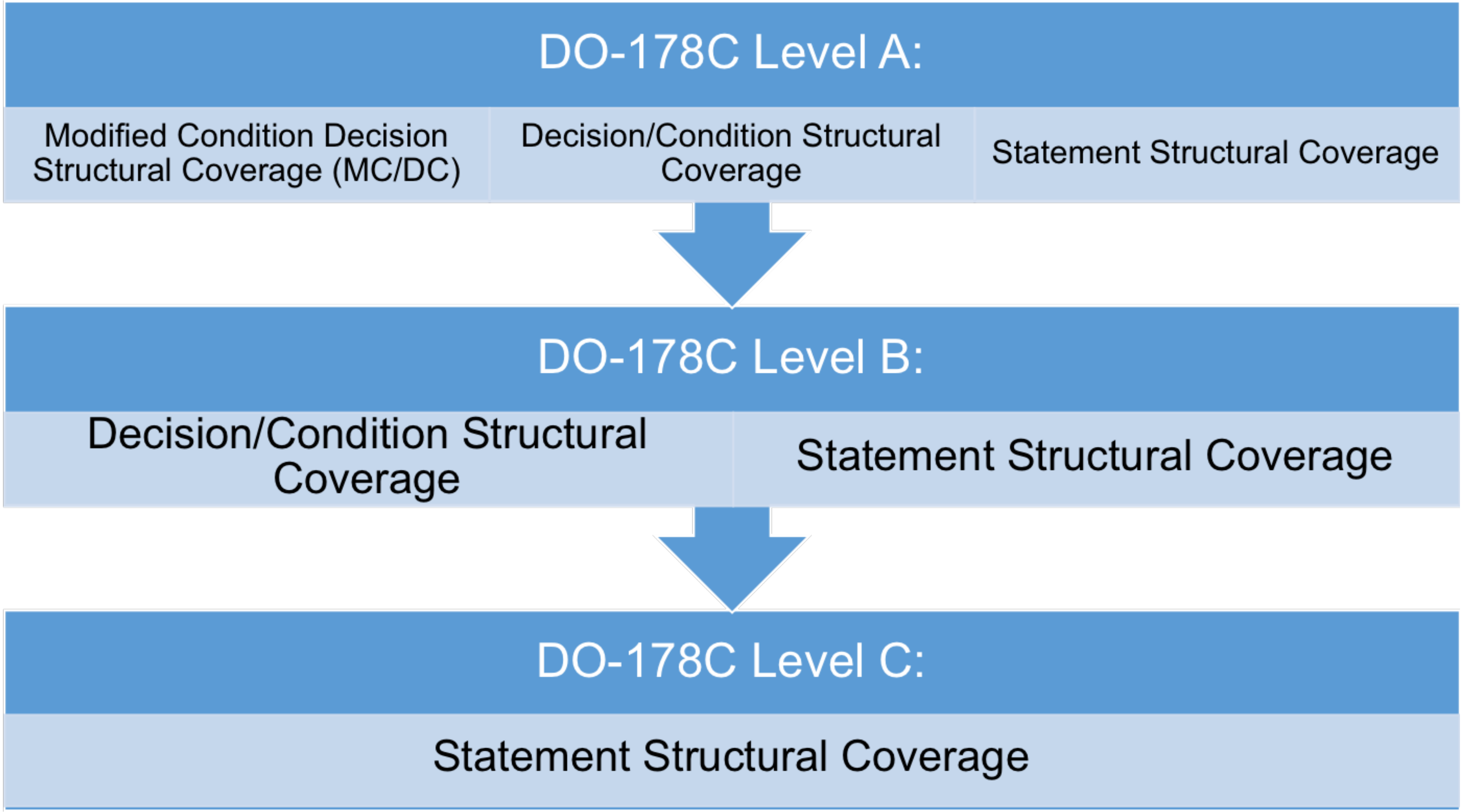
# DO-178C Software Considerations in Airborne Systems and Equipment Certification

**Ties testing requirements to consequences of a software error:**

- Level A: Catastrophic
- Level B: Hazardous/Severe
- Level C: Major
- Level D: Minor
- Level E: No Effect

# DO-178C Software Considerations in Airborne Systems and Equipment Certification

**According to Criticality Levels the following test coverage is required:**



DO-178C Level A:

Modified Condition Decision Structural Coverage (MC/DC)

Decision/Condition Structural Coverage

Statement Structural Coverage

DO-178C Level B:

Decision/Condition Structural Coverage

Statement Structural Coverage

DO-178C Level C:

Statement Structural Coverage

# When to Stop Testing

- Out of time / money
- No more defects found
- Demonstrated all requirements are met
- Demonstrated code coverage
- Meets reliability objects
- Customer is satisfied

# ISTQB Code of Ethics

## Public:

- Certified software testers shall act consistently with the public interest

## Client and Employer:

- Certified software testers shall act in a manner that is in the best interests of their client and employer, consistent with the public interest

# ISTQB Code of Ethics

## Public:

- Certified software testers shall act consistently with the public interest

## Client and Employer:

- Certified software testers shall act in a manner that is in the best interests of their client and employer, consistent with the public interest

## Product:

- Certified software testers shall ensure that deliverables they provide (on products and systems they test) meet highest professional standards possible

## Judgement:

- Certified software testers shall maintain integrity and independence in their professional judgement

# ISTQB Code of Ethics

## Management:

- Certified software test managers and leaders shall subscribe to and promote an ethical approach to management of software testing

## Profession:

- Certified software testers shall advance the integrity and reputation of the profession consistent with the public interest

# ISTQB Code of Ethics

## Management:

- Certified software test managers and leaders shall subscribe to and promote an ethical approach to management of software testing

## Profession:

- Certified software testers shall advance the integrity and reputation of the profession consistent with the public interest

## Colleagues:

- Certified software testers shall be fair to and supportive of their colleagues, and promote cooperation with software developers

## Self:

- Certified software testers shall participate in lifelong learning regarding the practice of their profession and shall promote on ethical approach to the practice of the profession

# Summary