

# Testing and Test Planning

---

## Outline

Goals

How Testing Fits In

Testing Tactics

Testing Strategy

- \* Next semester, test development and evaluation methodologies will be considered in greater detail.

## Goals for Testing (Subset of Goals for QA)

### Utility/Correctness

- Meets users needs

- Meets functional specifications

### Reliability

- Avoids unacceptable behavior under normal conditions

### Robustness

- Behaves properly across a variety of conditions and inputs, normal and abnormal

### Availability

- Recovers gracefully and quickly from anomalous behavior or conditions

### Maintainability

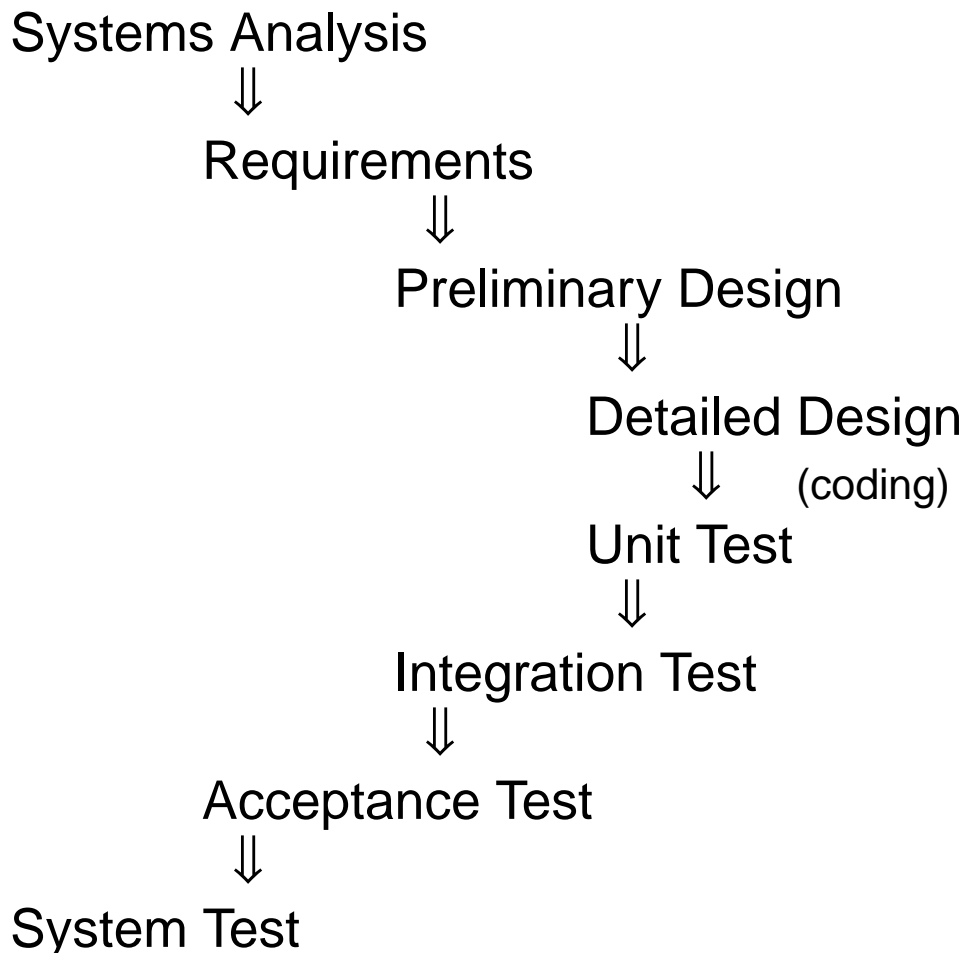
### Efficiency/Performance

### Security

Many of the above seem quite similar, but that's why it's important to recognize all the variations

## Testing & Life Cycle

- Test cycle mirrors life cycle:



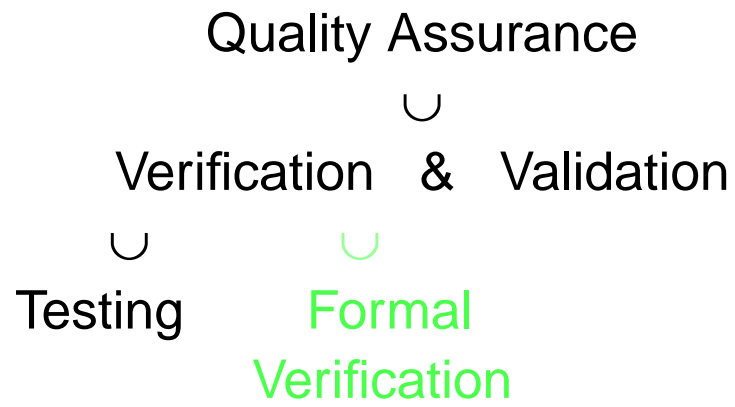
Unit test: individual modules

Integration test: assembled modules

Acceptance test: entire system in final environment

System test: validate requirements against goals

## Where Testing Fits In



## Vocabulary

(bottom up)

### Test Run

Execution of a single test

One run of a stand-alone program with test data as appropriate

### Test Case

Details required for a test run

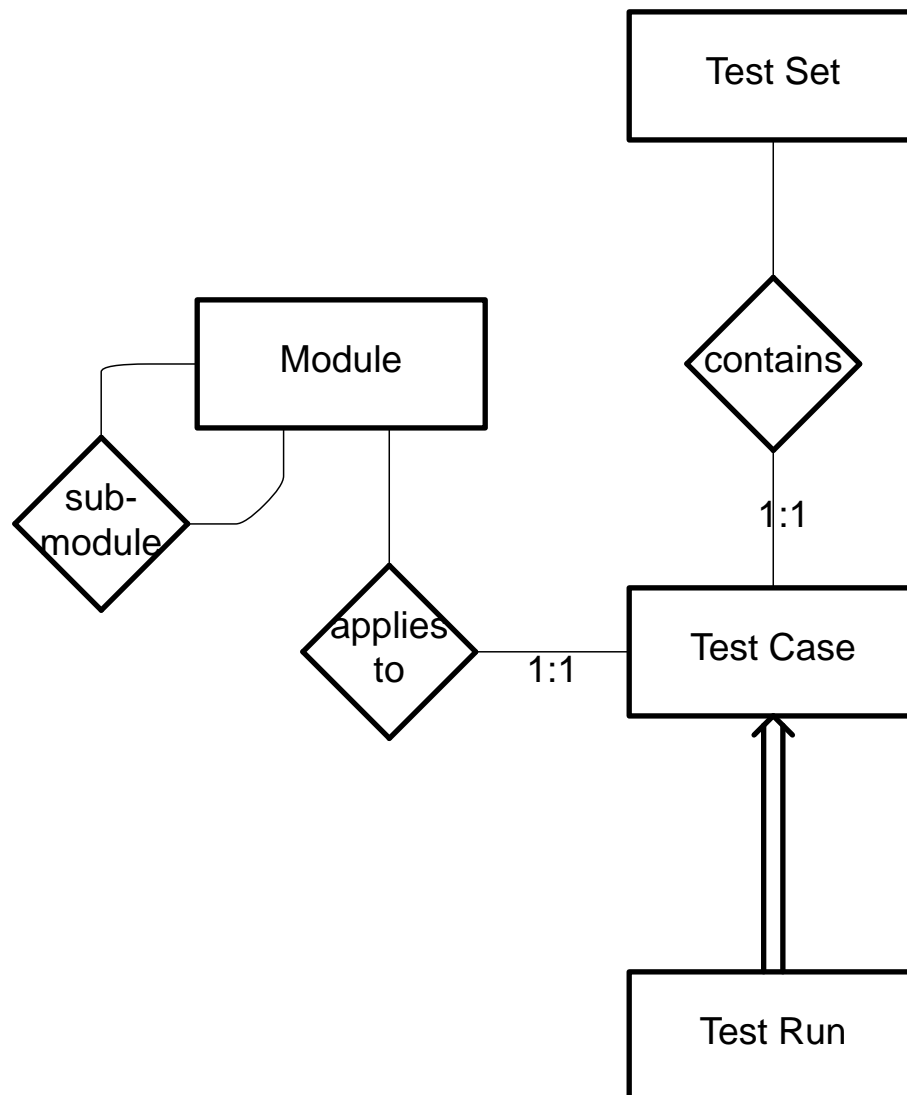
### Test Set

Collection of test cases

### Test Plan

Specification of test sets and testing process

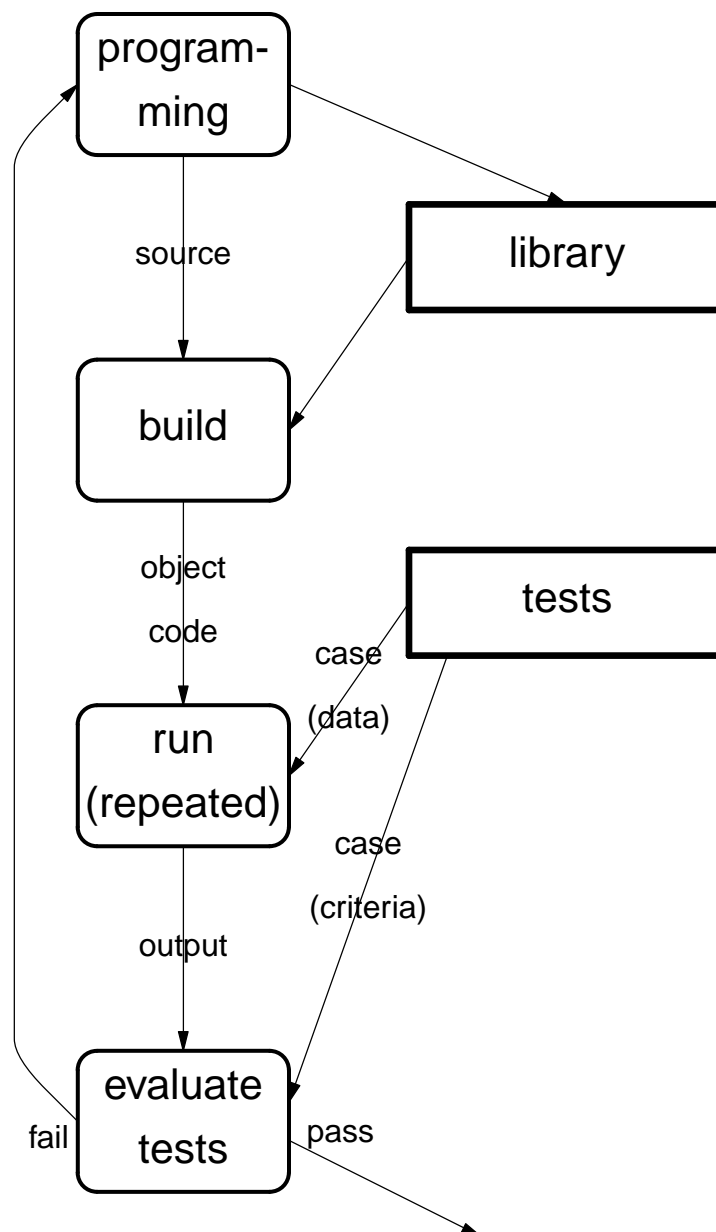
## Information Model for Testing



- “meta” hides lots of details
- allows for independent execution of cases
- “applies to” inferred for Set and Run

# Testing Process (Tactics)

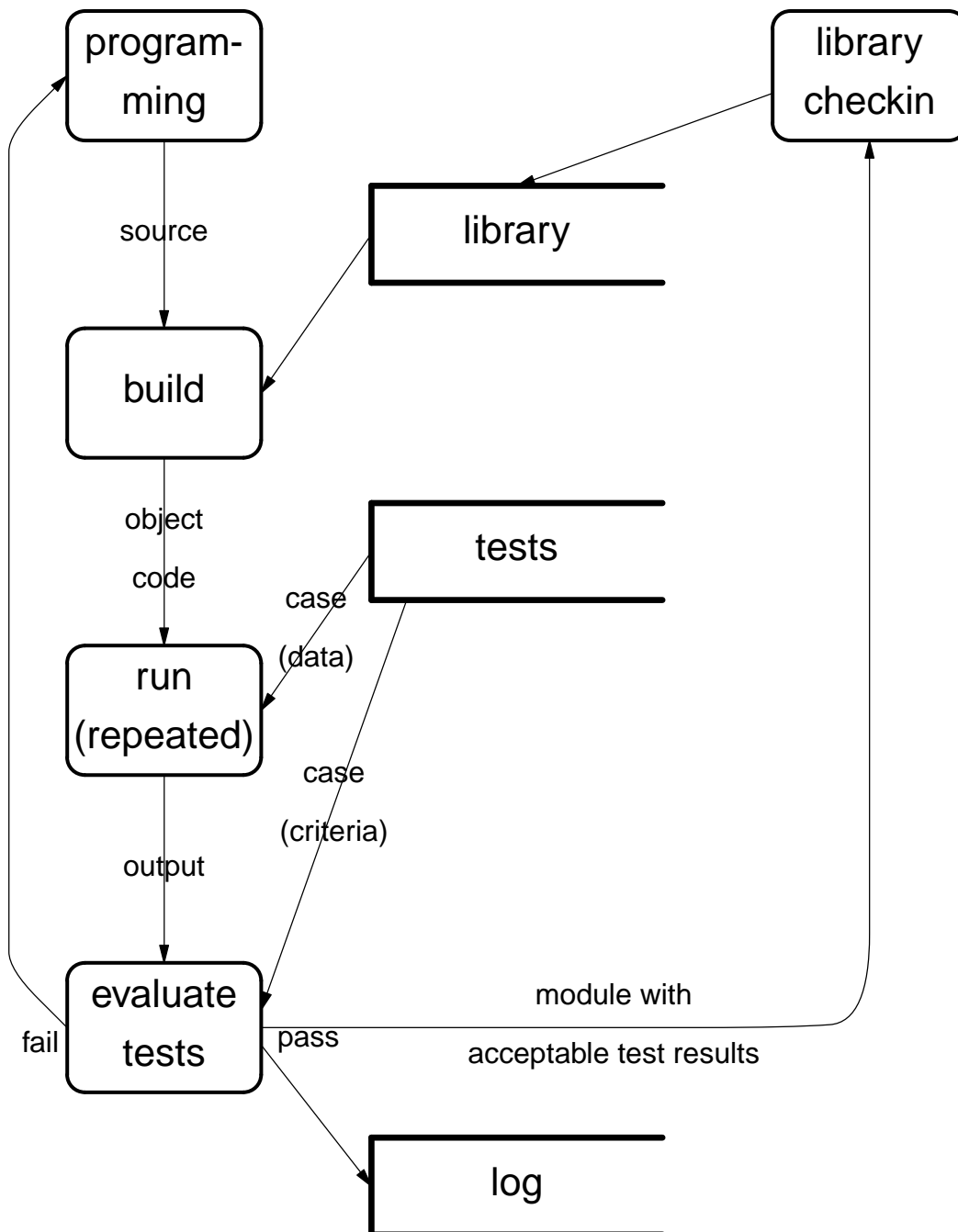
(traditional imperative code)



- Therefore need
  - ◇ test definitions
  - ◇ expected outcomes

## Testing Process with Librarian (Tactics)

(traditional imperative code with librarian)



- Correlates with version control
- Puts the *assurance* in QA

## Variations in Testing Process

Overall flow the same, specifics may differ:

- “compile & link” could be any make operation
- “run” has many variations

Well-defined process:

⇒

## Test Cases

A *test case* defines

- input to a single test run
- acceptance criteria

Test case input can be:

- single relation or data file
- collection of relations or data files
- human script
- “shell” script
- *etc.*

Acceptance criteria can be:

- expected outcome
- ways to compare expected and actual outcome
  - ◇ human
  - ◇ automated
- how to judge comparisons
  - ◇ human with possible computer assistance

## Testing Process – Strategy

- Set goals and objectives
- Develop a test strategy
- Specify testing
  - ◇ procedures (human + computer)
  - ◇ input  
test sets
  - ◇ outcomes+criteria
  - ◇ mechanisms
- Prepare test cases
- 
- Run tests
- Evaluate and report results
- Review test process for improvement

⇒ Test Plan



## Test Process Specifications

- Test execution process
  - ? degree of automation
- Test input mode
  - ? batch or online
- Test environment
  - ◇ setup/restore
  - ◇ framework
    - esp. for unit tests
    - a.k.a scaffolding
- Test output mode
  - online output difficult to capture
- Test evaluation
  - ? degree of automation

## Test Set Specification

- Specify
  - ◇ test target  
module, functionality, *etc.*
  - ◇ test objective  
utility, robustness, *etc.*
  - ◇ input mode
  - ◇ content characteristics
  - ◇ evaluation mode
  
- Multiple test sets can be covered by one specification

# Test Plan

 **Test Plan should only characterize tests,  not give the details of the test sets!**

Specify general testing processes

Specify test sets

- From Requirements Specification:  
for each functionality:
  - ◇ specify test sets
  
- From Preliminary Design  
for each module (code, form, *etc.*)
  - ◇ specify test sets