



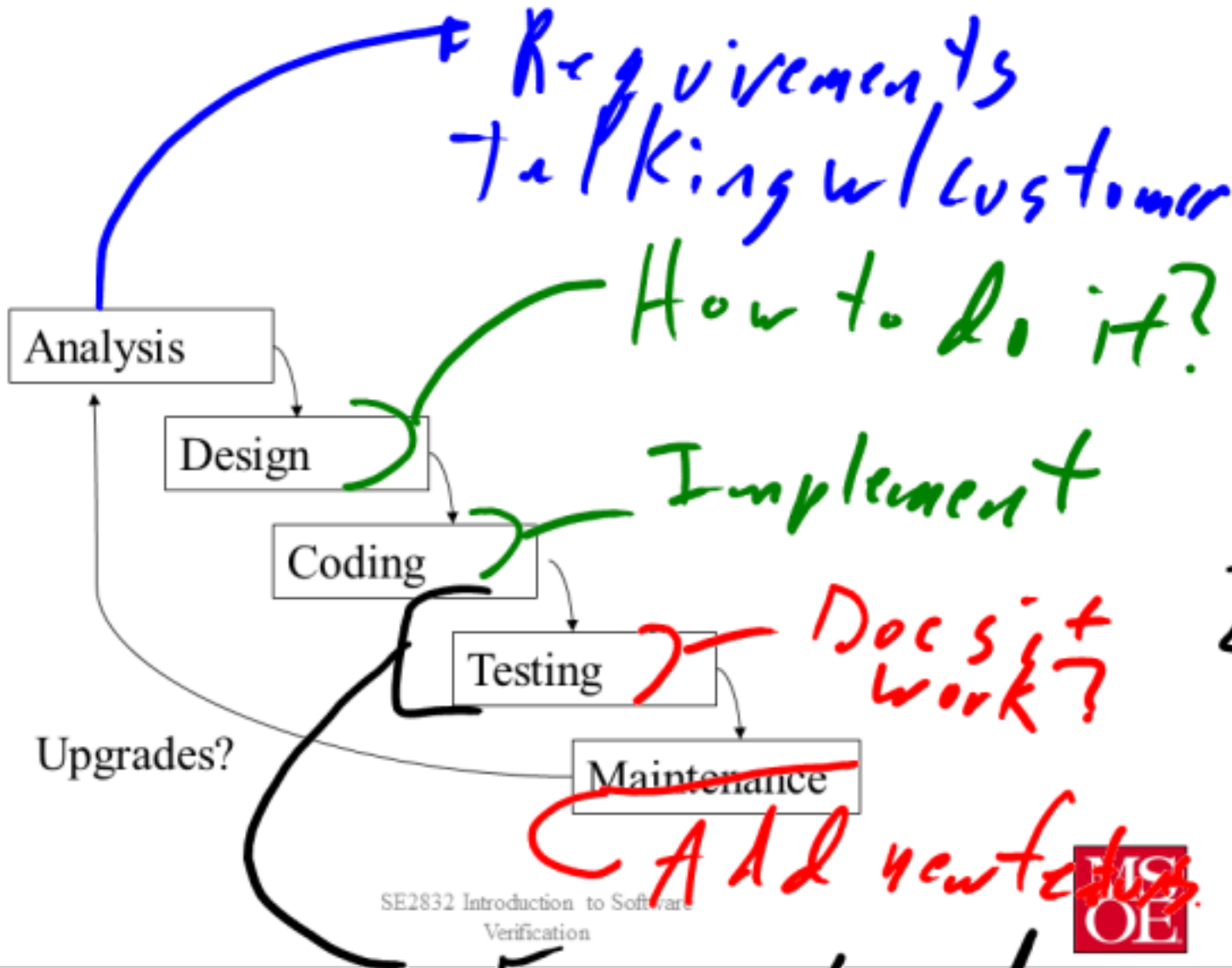
Test Activities

Lecture Objectives:

- 1) List the activities conducted by test engineers
- 2) Draw the software V Model and explain the relationships between testing activities and design activities.
- 3) Explain the relationship between acceptance testing, system testing, integration testing, module testing, and unit testing.
- 4) Explain Beizer's Testing Levels and Test Process Maturity
- 5) Define the relationship between testing and debugging.
- 6) Explain the relationship between the cost of fixing a defect and the phase in which the defect is discovered.

- Traditional approach

Waterfall Model



1. Change will happen.

2. One product.

First idea of quality

Test Design in Context

- *Test Design* is the process of designing input values that will effectively test software
- Test design is one of several activities for testing software
 - Most mathematical
 - Most technically challenging

Discrete
Math

Test Engineer & Test Managers

- Test Engineer: An software professional who is in charge of one or more technical test activities

- designing test inputs
- producing test values
- running test scripts
- analyzing results
- reporting results to developers and managers

- test cases

- JUnit

- How big of a problem?

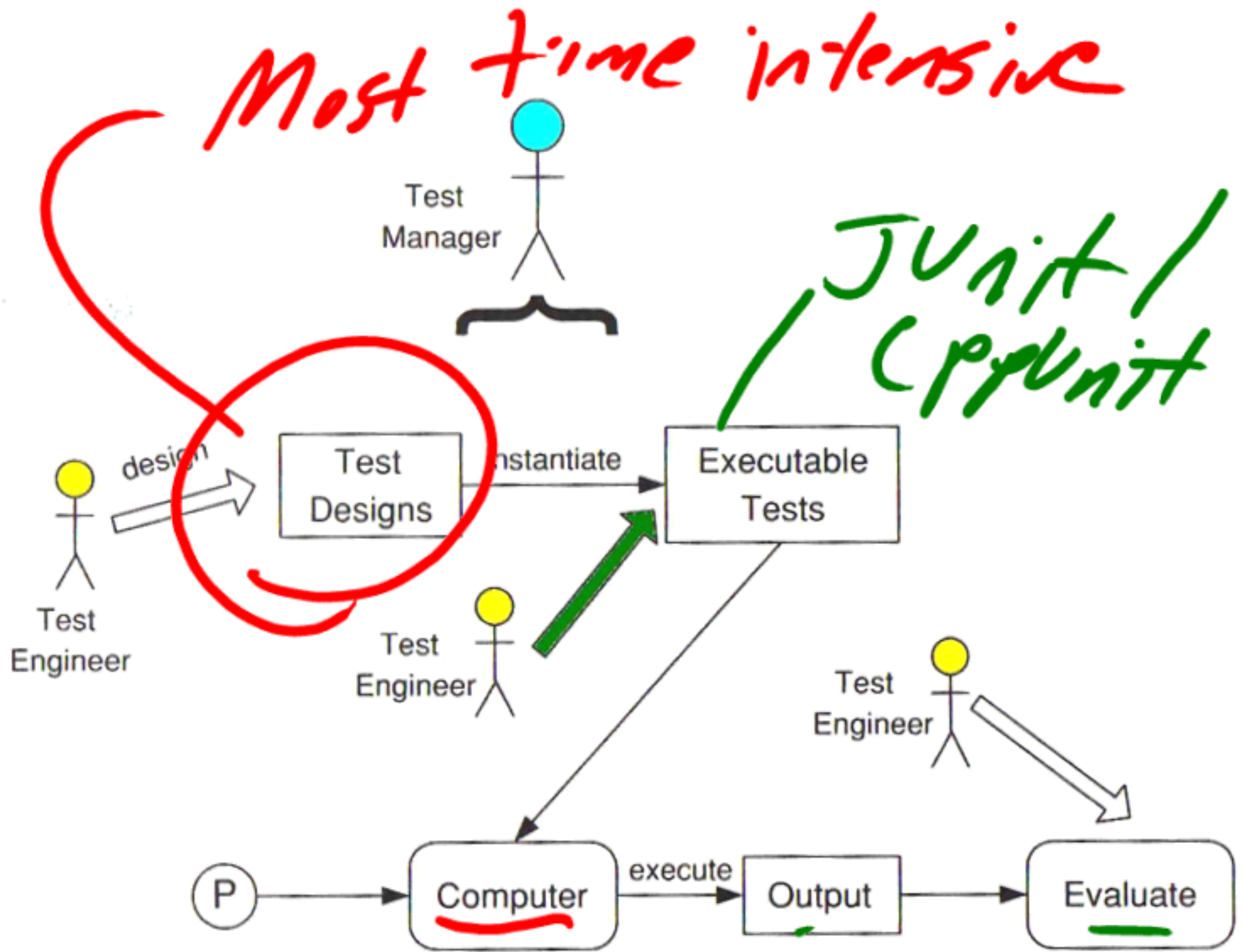
AAA

- Test Manager: In charge of one or more test engineers



- sets test policies and processes
- interacts with other managers on the project
- otherwise helps the engineers do their work

Facilitating

Activities of Test Engineers



1. Test Design – (a) Criteria-Based

- This is the most technical job in software testing
- Requires knowledge of :
 - Discrete math 
 - Programming 
 - Testing
- Requires much of a traditional CS degree
- This is intellectually stimulating, rewarding, and challenging
- Test design is analogous to software architecture on the development side
- Using people who are not qualified to design tests is a sure way to get ineffective tests

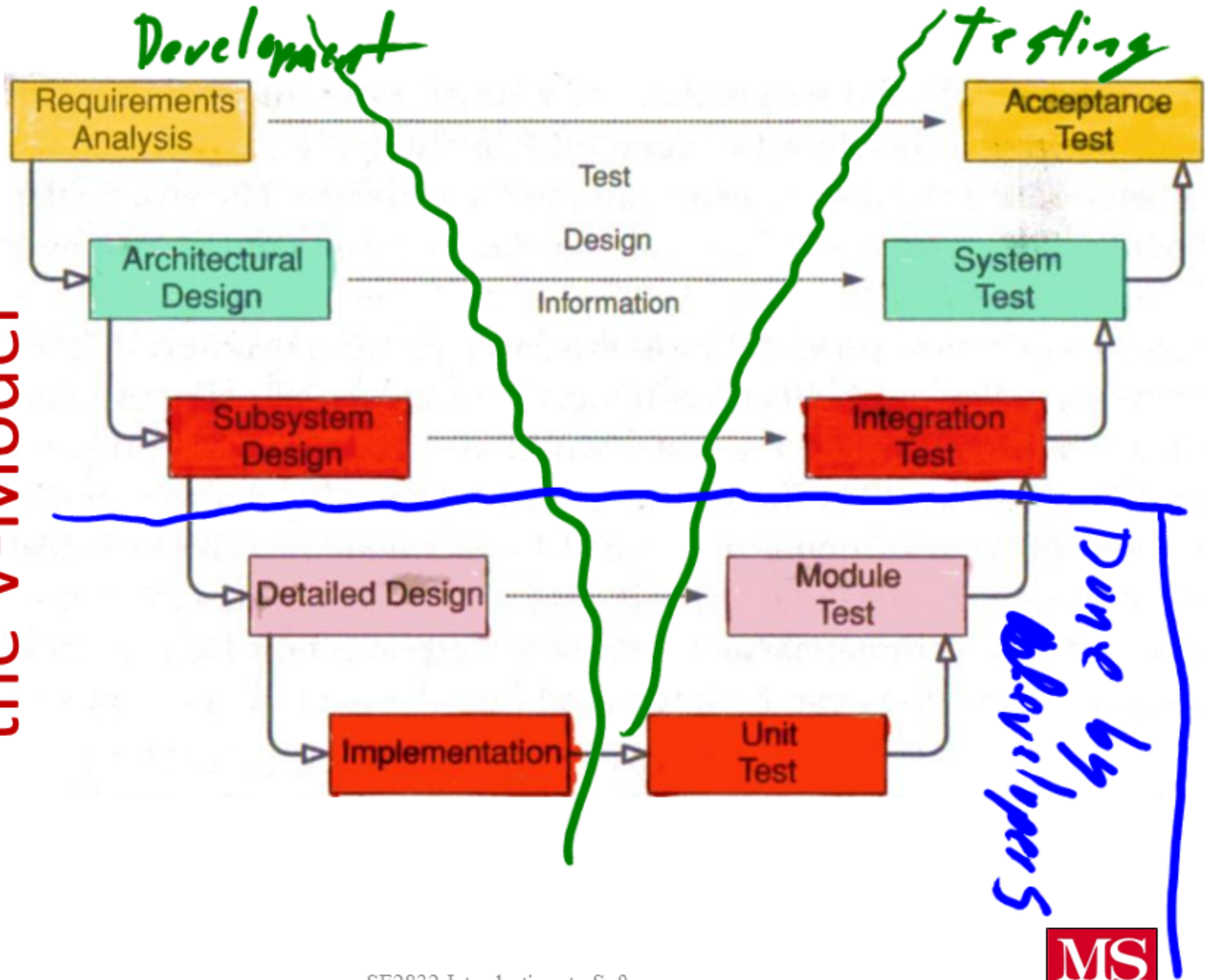
SE

1. Test Design – (b) Human-Based

- This is much **harder** than it may seem to developers
- Criteria-based approaches can be blind to special situations
- Requires **knowledge** of : *Problems*
 - Domain, testing, and user interfaces
- Requires almost **no traditional CS**
 - A background in the **domain** of the software is essential
 - An **empirical background** is very helpful (biology, psychology, ...)
 - A **logic background** is very helpful (law, philosophy, math, ...)

Software Development and

the V-Model



Testing Activities

- Acceptance Testing
 - Assess the software with regards to the requirements
- System Testing
 - Assess the software with respect to the architectural design
- Integration Testing
 - Assess software with regards to subsystem design
- Module Testing
 - Assess the software with respect to detailed design

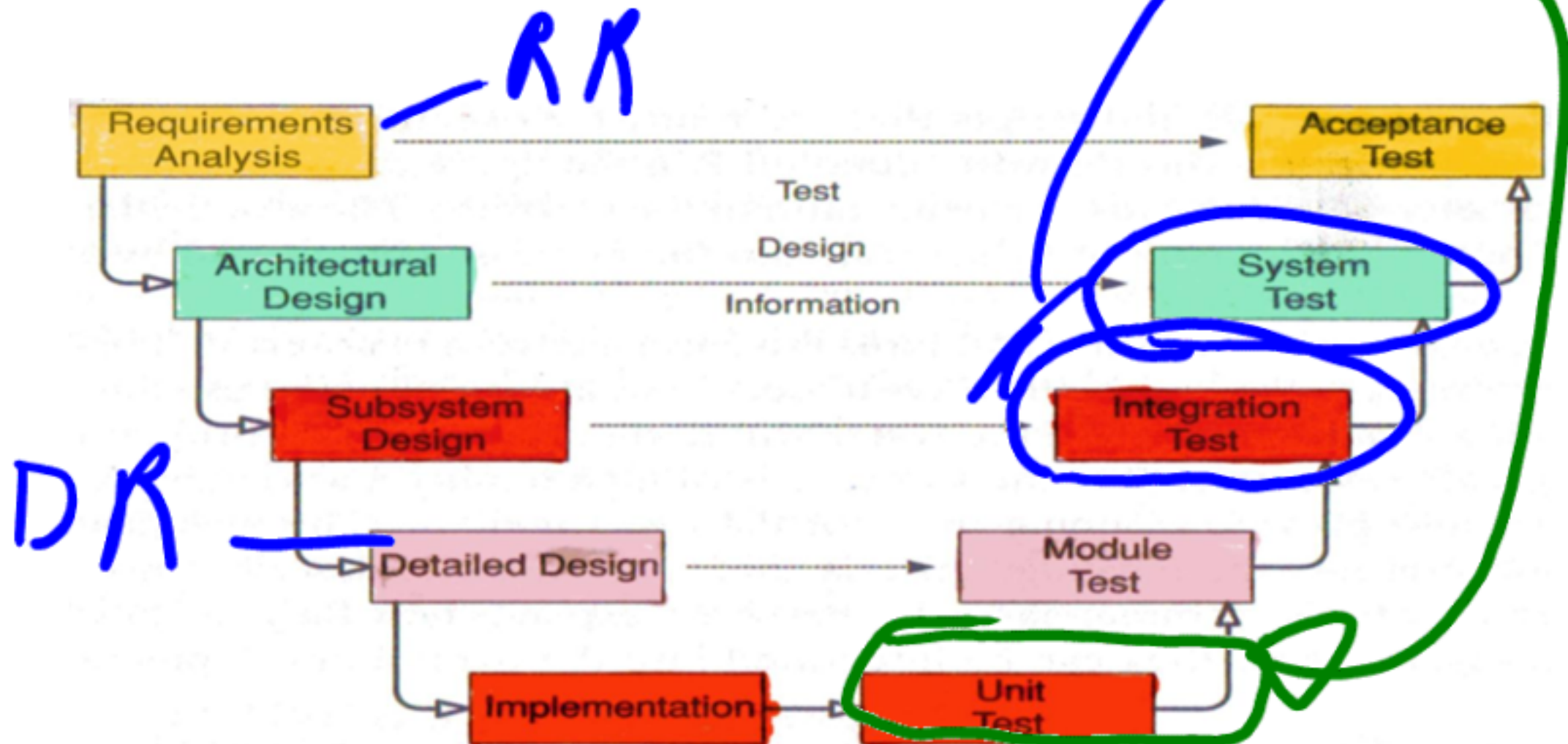
Definitions

- Unit
 - One or more contiguous program statements with an appropriate identifier
 - i.e. procedure, function, method
- Module
 - A collection of related units assembled into a ~~file~~, package, or ~~class~~

Usually.
Tested by the
developer.

Using different levels of testing

- Different levels of testing should look for different types of faults
 - Case study: Mars lander of September 1999
 - One module used English units
 - One module used Metric units
 - When should this problem have been caught?
 - Zune failure
 - When should this problem have been caught?



Why have you tested your programs?

Discussion: Why have you tested your programs?

Using Ubiquitous presenter, select one of the following which best represents why you have tested your programs.

- A. To debug my program
- B. To prove that the software works
- C. To prove that the software doesn't work.
- D. To reduce the risk of using the software.
- E. To develop higher quality software.

Test Process Maturity

SE / Developers

L4: Testing is a mental discipline that helps all IT professionals develop higher quality software.

L3: The purpose of testing is not to prove anything specific, but to reduce the risk of using the software.

L2: The purpose of testing is to show that the software doesn't work.

L1: The purpose of testing is to show that the software works.

L0: There is not different between testing and debugging.

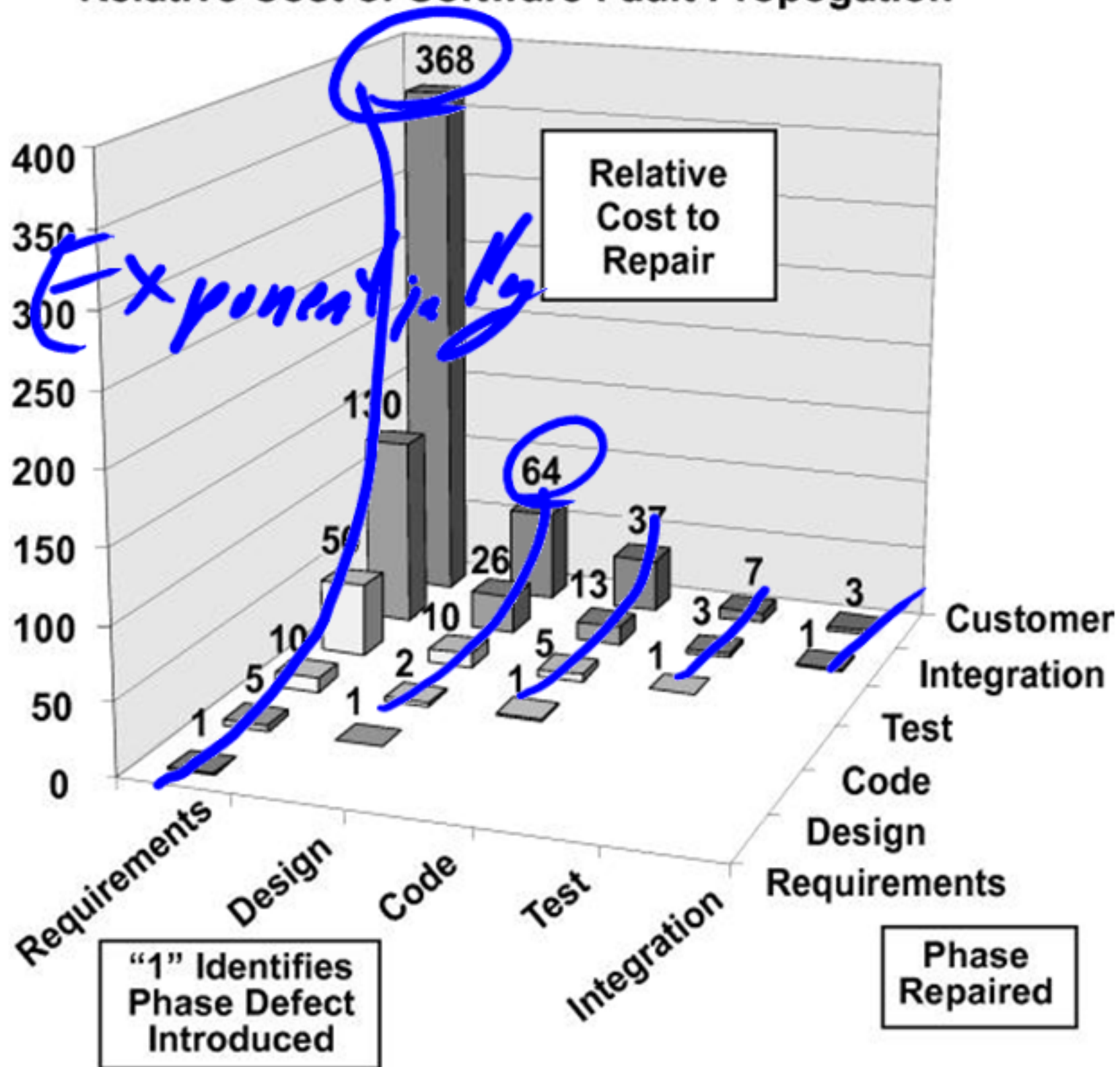
Definitions

- Testing
 - Finding inputs that cause the software to fail
→ Figuring out what breaks SW
- Debugging
 - The process of finding a fault given a failure
↳ Something goes wrong
What caused it?

Relative Cost of Software Fault Propagation

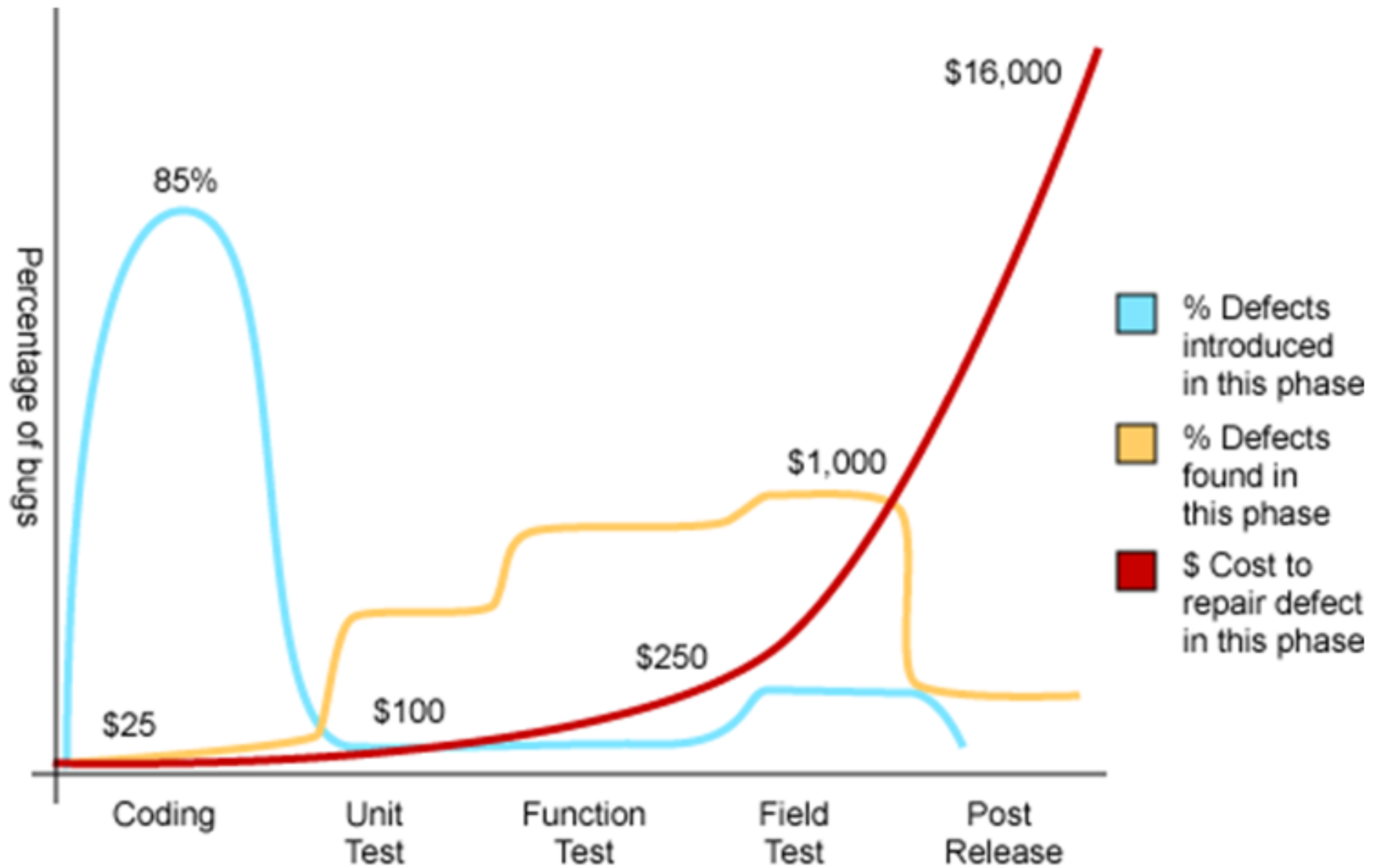
Cost of Software Fault

Propagation



Software Fault Propagation

– Alternate Data



Source: *Applied Software Measurement*, Capers Jones, 1996