



STBC

The Economics of Testing

Objectives

- Understand where defects originate
- Learn about the 1:10:100 rule
- Learn how to make the most of your testing investment

Synopsis

This module discusses why software defects are costly to find and fix. You will also learn how early defect detection can save much cost in fixing defects.

Table of Contents

STBC –The Economics of Testing

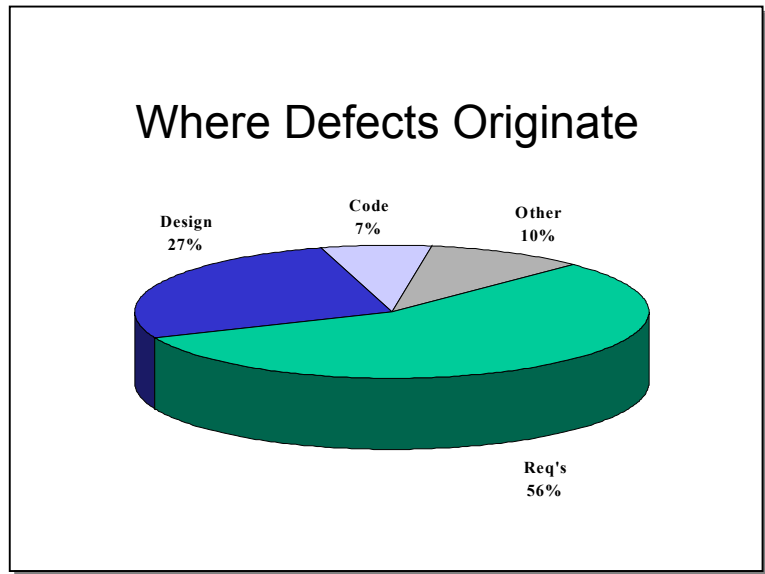
- Where defects originate
- Where testing resources are used
- The relative cost of fixing defects
- The bottom line

The Economics of Testing

There is a definite economic impact of software testing. One economic impact is from the cost of defects. This is a very real and very tangible cost.

Another economic impact is from the way we perform testing. It is possible to have very good motivations and testing goals while testing in a very inefficient way.

In this section, we will examine the economic impact of defects and ways to economize testing.



Where Defects Originate

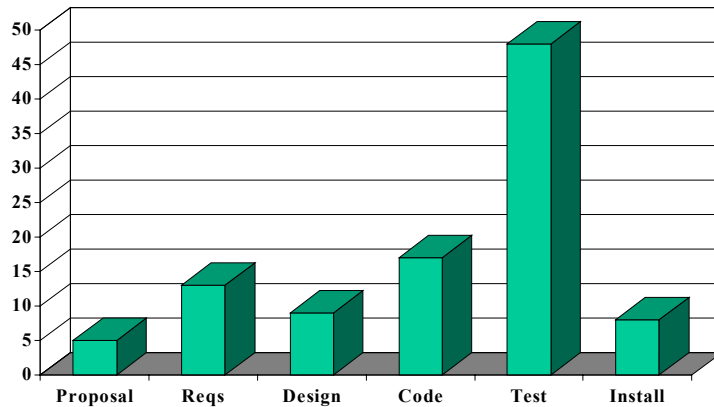
To understand the dynamics and costs of defects, we need to know some things about them. One of the most commonly understood facts about defects is that most defects originate in the requirements definition phase of a project. The next runner-up is the design phase.

Some problems in getting accurate, clear, and testable requirements are:

- Many people do not have a solid requirements gathering process
- Few people have been trained in or understand the dynamics of requirements
- Projects, people, and the world around us change very quickly
- The English language is ambiguous and even what we consider clear language can be interpreted differently by different people.

The figures in this pie chart were taken from a James Martin study and the numbers track very closely to measurements of typical software projects.

Where Testing Resources are Used

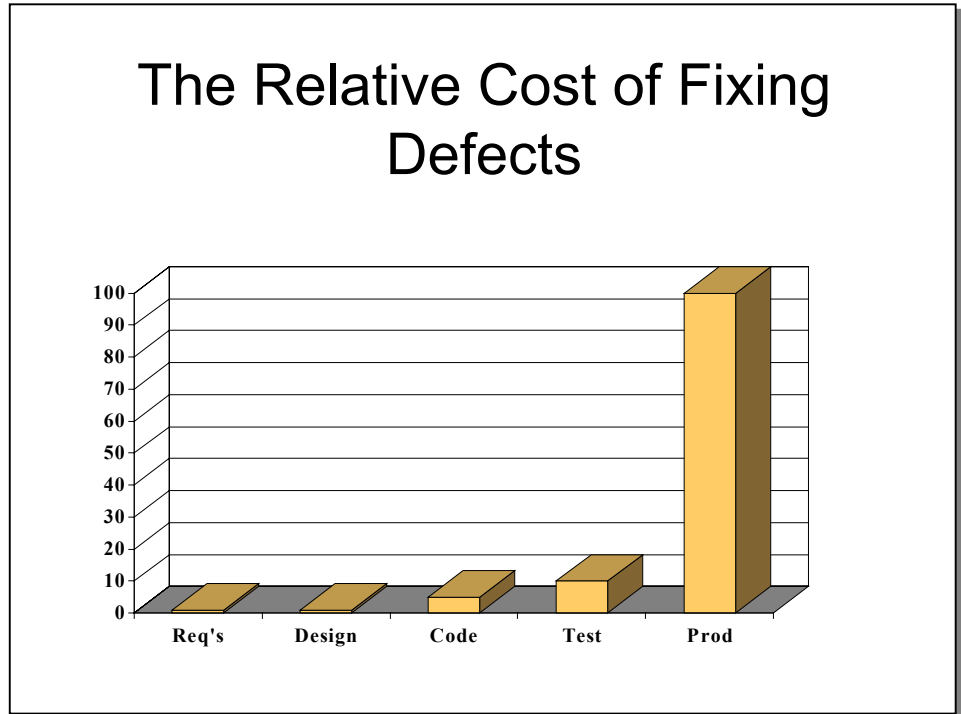


Where Testing Resources are Used

We saw that most defects originate in requirements and design, but most of the testing effort occurs in a traditional “testing” phase toward the end of the project. This is called the “big bang” approach from the concentration of effort at one big phase. “Big bang” could also describe the sound of the project as it fails.

The problem with the big bang approach to testing is that defects are not found until toward the end of the project. This is the most costly and risky time to fix defects. Some complex defects may even be impossible to fix.

The figures above were taken from surveys at the Quality Assurance Institute’s (QAI) annual software testing conference. To obtain the current survey, go to www.qaiusa.com.



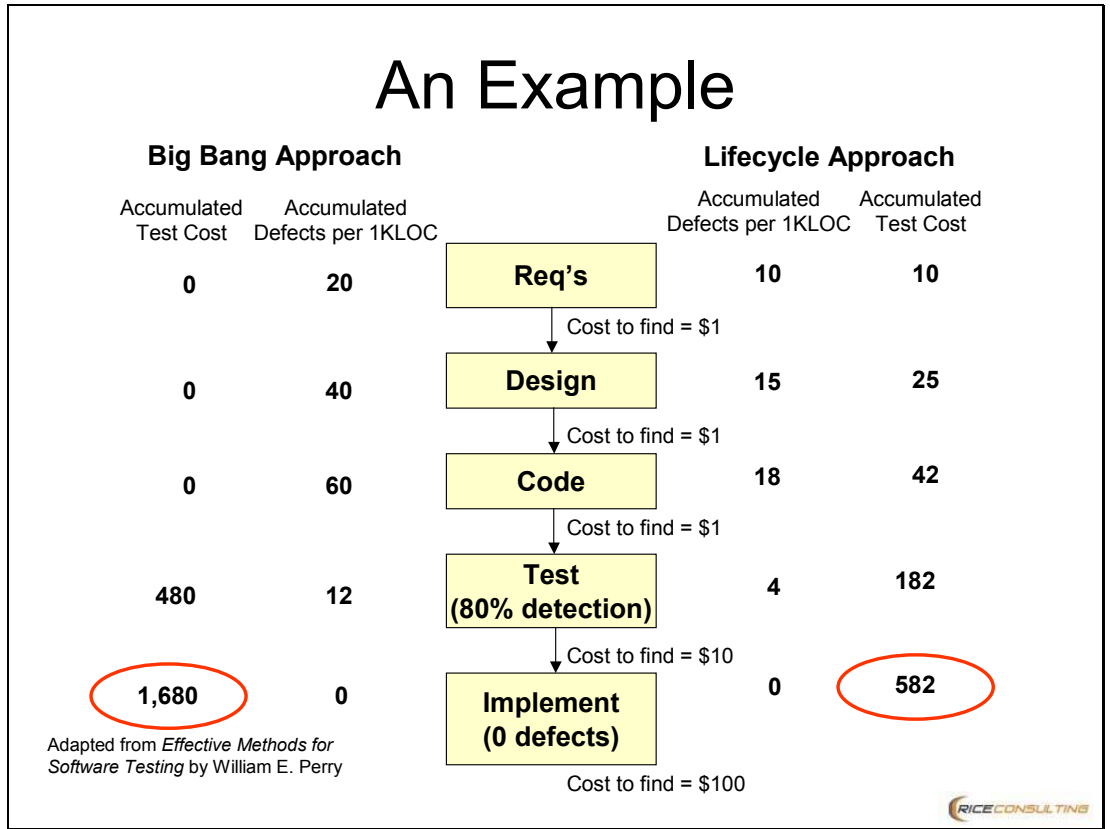
The Relative Cost of Fixing Defects

One of the most well known facts about software defects is that the longer they go undetected, the more expensive they are to fix. Although research differs on the exact ratios, the general rule is 1:10:100.

That is, if a defect costs one unit (hour, dollar, etc.) to fix in requirements and design, it costs 10 units to fix in testing (system/acceptance) and over 100 times to fix in production. Sometimes the cost to fix a defect in production costs much more than 100 times the cost of fixing it in the requirements phase.

This cost of defects doesn't even take into account the impact cost of defects. These costs could be attributed to lost revenue, reimbursements, fraud, lost customers, bad public relations, and litigation. In the case of safety critical systems, how can one put a cost value on a human life?

An Example



An Example

This slide shows two contrasting views of testing – the lifecycle approach where testing is performed throughout the project, and the big bang approach where testing is performed only at the end of the project. The final cost of the big bang approach is much greater than the lifecycle approach.

The Bottom Line

- Most defects are created in the early stages of a project
- Most defects are found in the later stages of a project
- It costs 10 to 100 times as much to fix a defect in the later phases of a project.



The Bottom Line

So, what does all of this mean? The main conclusion is that most people perform testing too late in the process. These people wonder why testing is so expensive and why their projects are often over budget.

If you really want to make your testing more efficient and reduce the overall cost of testing and defects, test early in the project and continue testing throughout the project. Most defects can be found by inspections even before a test is performed!

The Bottom Line

- If you want to reduce the cost of testing, spend time early in the system development (or purchase) process to make sure the requirements and design are correct.



This document may not be reproduced.