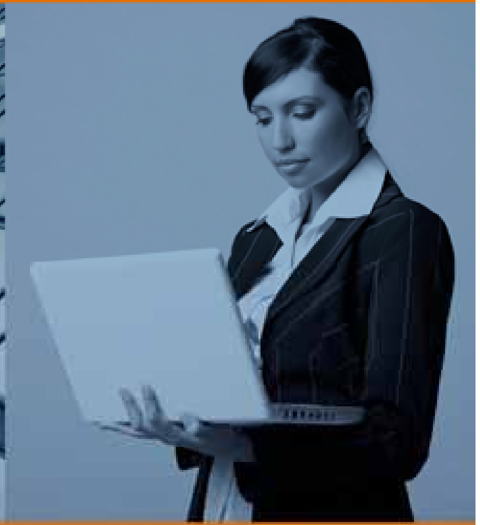


Test Case Mgmt and Bug Analysis

2008 hiSoft

Agenda

- Targets of Quality Assurance in SDLC
- Test Case Mgmt and Coverage
  - Common features of Test Case Mgmt Tool
  - Test Link
  - Code Coverage and Cobertura
- Bug Analysis Matrix

Target of QA

- ✓ Reduce cost
  - ✓bug cost
  - ✓testing cost
- ✓ Control software release risk

Bug Cost

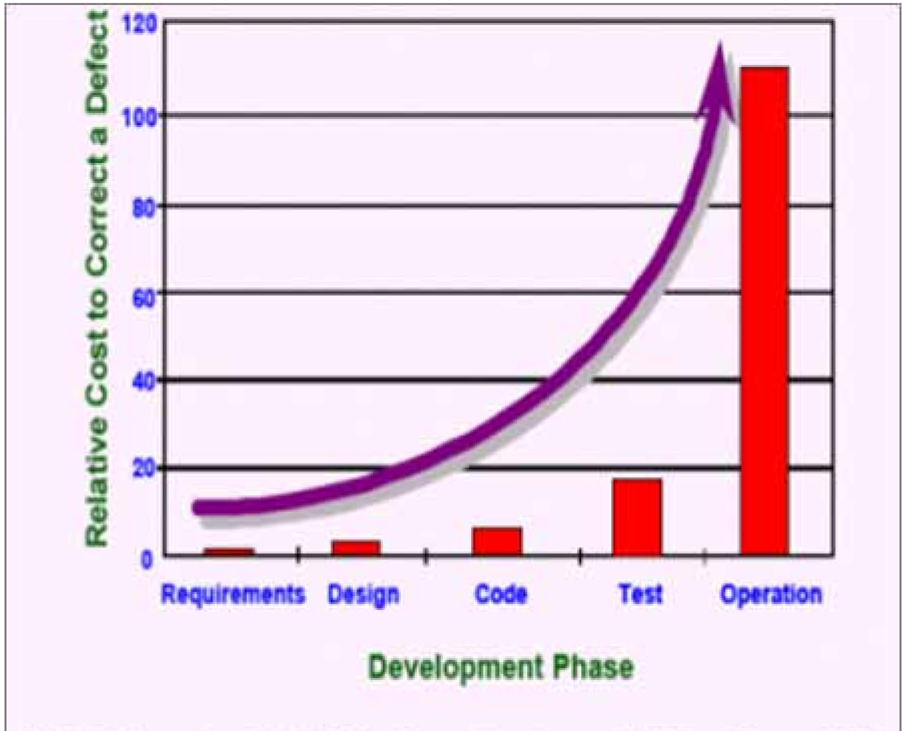
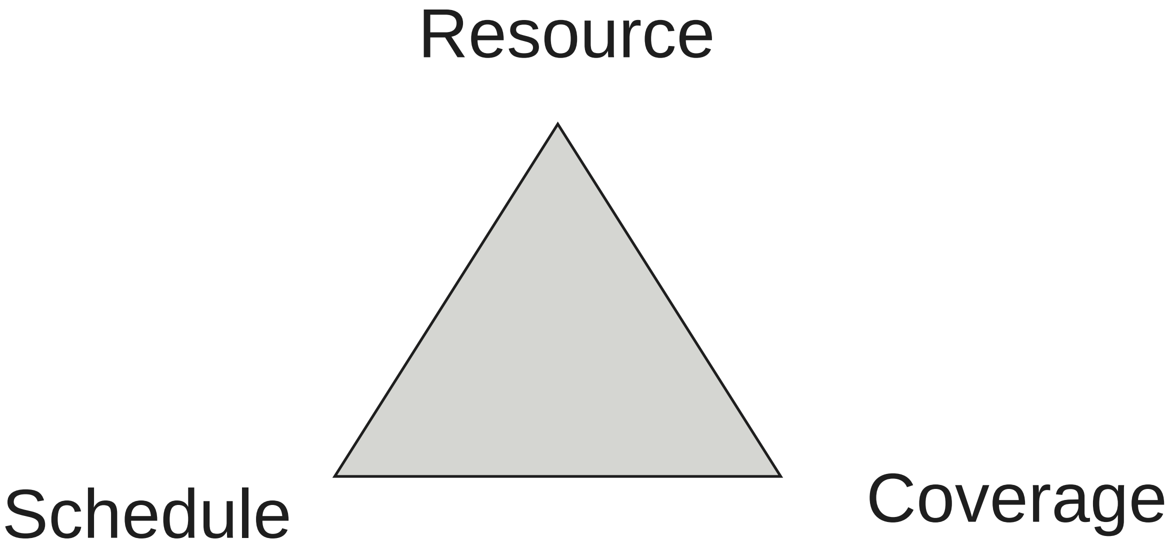


Exhibit 1. Increasing Cost of Defect Correction. (source: NSQE and Segue, 2004)

Bug cost control

- **Early and continuous** involvement from Quality Assurance
  - Requirement and Design Review
  - Code
    - API review
    - Unit Testing
    - Continuous integration
  - Testing
    - Smoke Testing
    - Integration, performance testing etc

Testing Cost



Resource

Schedule

Coverage

Testing cost control

- Dilemma
  - Less coverage (less cost) -> higher risk
  - More mature product -> More test cases
- Test More for Less
  - Prioritize TC
    - Critical path
    - New feature vs old feature
    - TC and bug correlation
  - Diff priority for different test cycles
  - Cross coverage
  - Automation and CI

Risk Control

- Predictability and Visibility

Test Case Mgmt Tool

- Case mgmt
  - Create/Edit/Version control
  - Grouping
    - Product, version, test group, test suite, etc
  - Keyword/Category
    - Priority, feature etc.
  - Searching
- Execution mgmt
  - Group test case per test plan
  - Test assignment
  - Test case execution status update

Test Case Mgmt Tool – Cont.

- Reporting
  - Execution status report
    - Passing/fail/block cases and rate
    - Matrix: by person, suite/group, priority, keyword
  - Progress report
    - Completion rate
    - Timing tracking
- Integration and others
  - Bug system integration
  - Requirement system integration

TestLink - Demo

Code Coverage

- **Code coverage** is a measure used in software testing. It describes the degree to which the source code of a program has been tested.
- **Coverage criteria**
  - Function
  - Statement
  - Condition
  - Path

Code Coverage – Common Practice

- Tool or library are available for most of common programming languages
- Coverage analysis
  - Coverage omission
  - A quantitative measure of code coverage
  - Identifying redundant test cases that do not increase coverage.
- Mechanism
  - Code instrument

Code Coverage – Common Practice

- Higher coverage does not always higher return

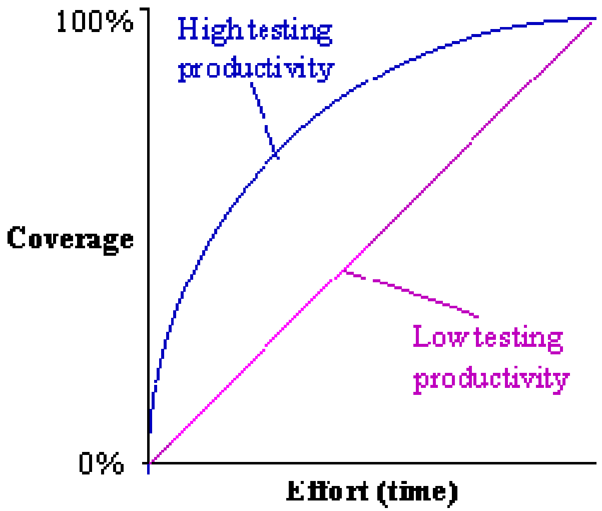


Figure 1 Coverage rate

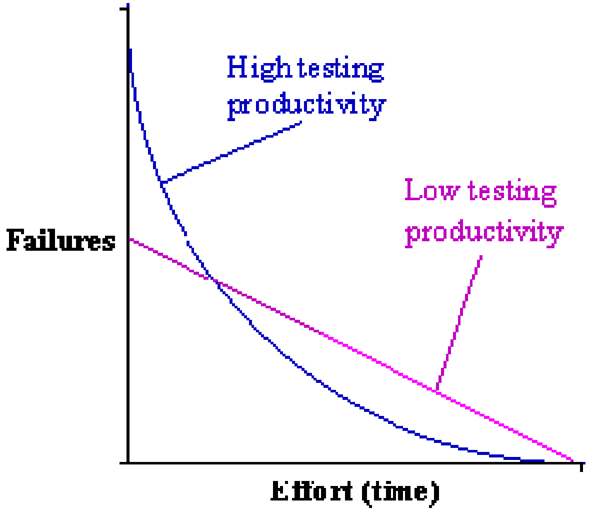


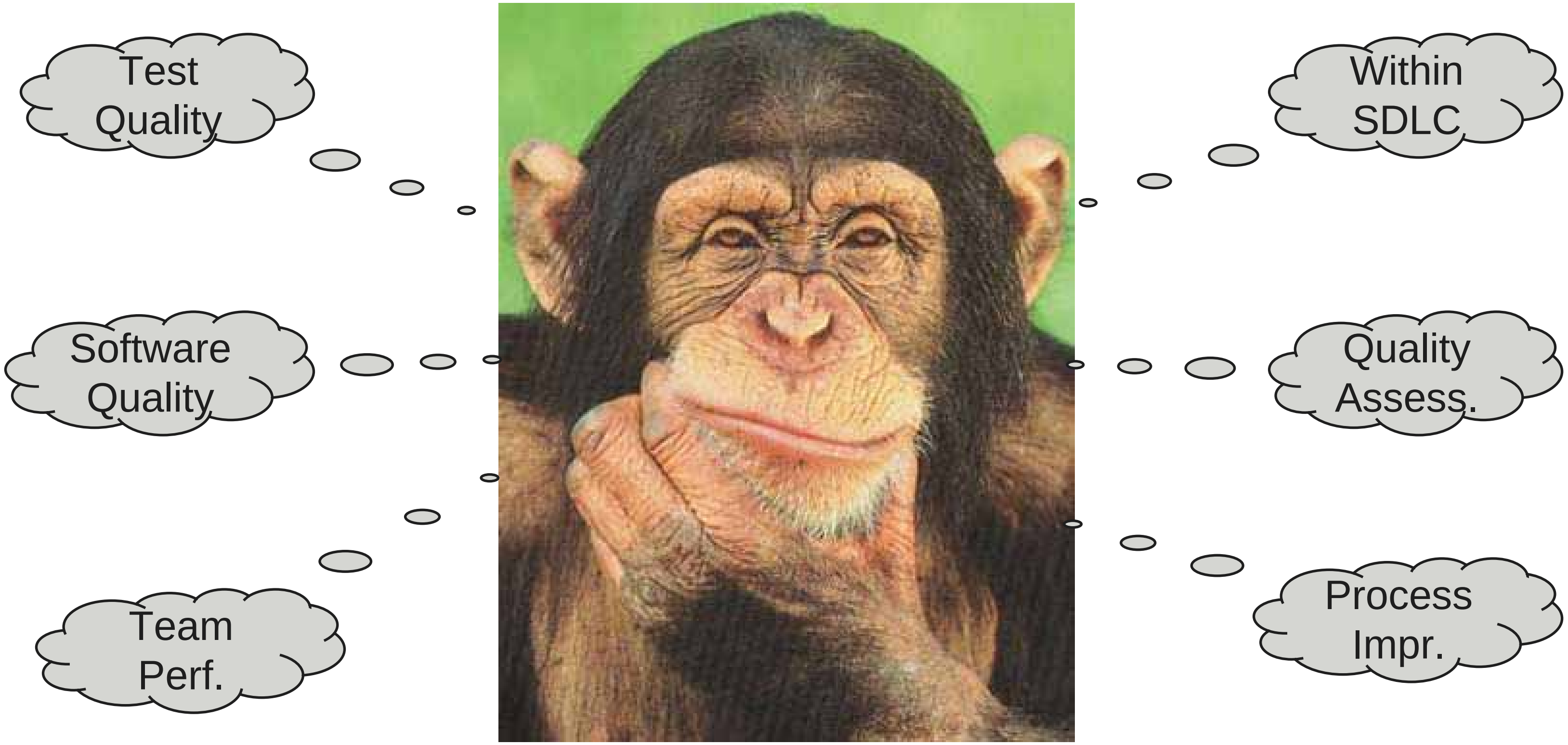
Figure 2: Failure discovery rate

Code Coverage – Tool

- Java
  - EMMA
  - Clover
  - Cobertura
- Demo
  - http://cobertura.sourceforge.net/sample/

Bug Analysis Matrix

- What QA manager is thinking



Test Quality

Software Quality

Team Perf.

Within SDLC

Quality Assess.

Process Impr.

Bug Analysis Matrix - 1

- Within SDLC
  - Are enough bugs found for specific function/OS/App category?
  - Should we adjust the coverage for specific function/OS/App category?

  - When we can release software with current bug trend?
  - What is the risky function/OS/App category etc?

  - What is the perf for QA (open bug, verify bug) and Dev (Fix bug)?
  - What is the perf bottom net that impact software release?

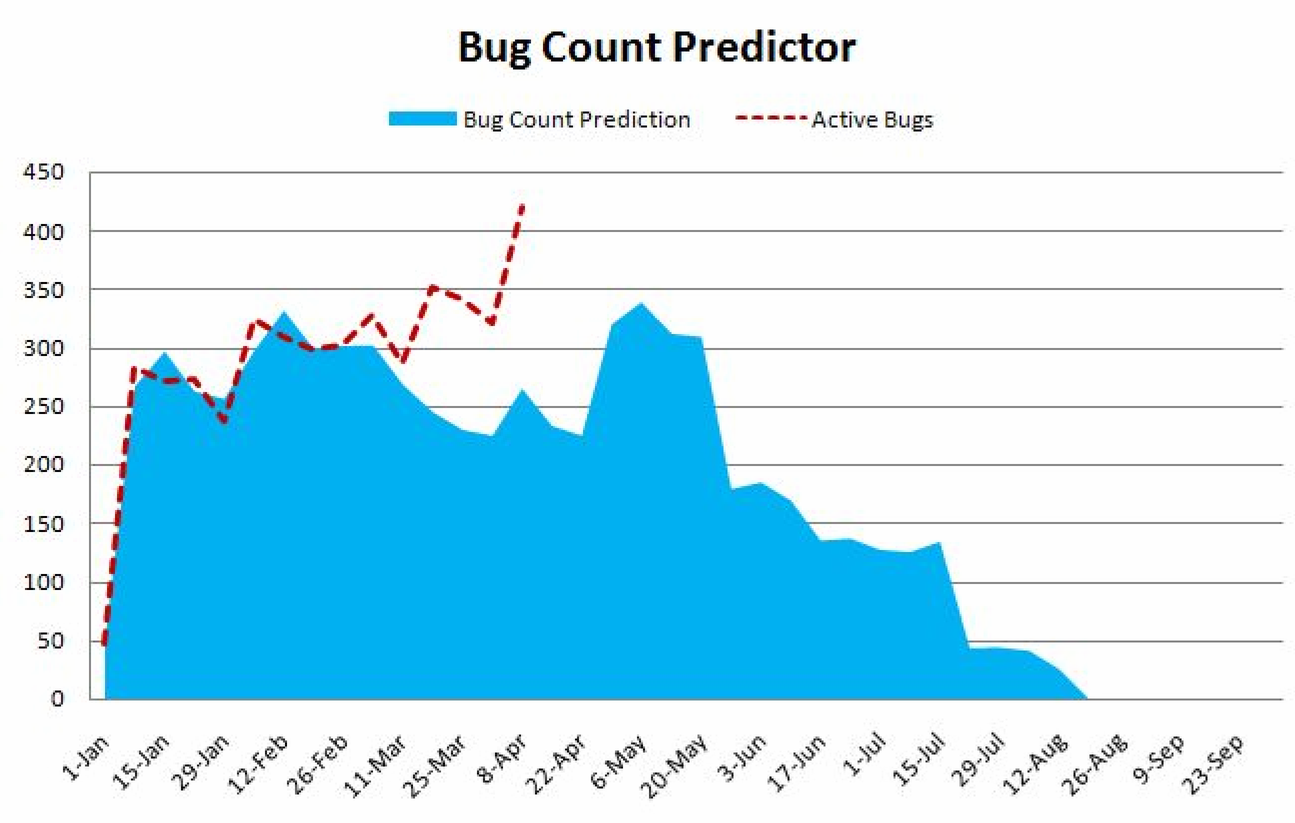
Bug Analysis Matrix - 2

- Quality Assessment
  - How well of Quality Assurance in each SDLC?
  - What is the quality of software vs previous release in specific category?
  - What is the performance of Dev in term of quality?

Bug Analysis Matrix - 3

- Process Improvement
  - Is there enough or too much coverage?
  - Is there any mis-alignment for the test cycles regarding priority?
  - Is there any way we can shorten test cycle?
  - How well the case are designed?
  - How much effort we should be put for automation?

Some Examples



hiSoft

Your Trusted Partner in China

Thank You!