Maximize Software Development ROI With Quality Assurance

Showing the value of the Quality Process

Thibault Dambrine

Agenda

Software Quality Assurance ROI
  - Quantifying the Cost of Quality
  - Justifying a Software QA/QC Budget

Software Quality Principles for Managers 101
  - Quality Assurance vs. Quality Control
  - Quality Methodologies
Part 1
Measuring
Quality Assurance
Return on Investment (ROI)

Quality Definition

PMBOK ®
The degree to which a set of inherent characteristics [consistently] fulfills requirements
What is QUALITY worth to you?

• How does one quantify quality?

• How does one justify a QA/QC Staff?

Measuring the Cost of Quality

• Price Of Compliance (POC)
  - Cost of Prevention, QA/QC

• Price of Non-Compliance (PONC)
  - Cost of Internal Failures
  - Cost of External Failures

Price of Quality = POC + PONC
Price of Compliance - POC (1 of 2)

**Development Activities**

- Staff training
- Requirements analysis
- Early prototyping
- Fault-tolerant design
- Defensive programming
- Accurate internal documentation
- Proper Requirements
- Detailed Design Documents

Price of Compliance (POC) (2 of 2)

**QA/QC Activities**

- Design review
- Code inspection
- Unit testing
- End-to-End testing
- Regression Testing
- Beta testing
- Test automation
- Pre-release testing by staff
Price of Non-Compliance (PONC)

Internal **High Visibility** Costs

- Bug fixes
- Wasted in-house user time
- Developer fixing time
- Tester re-testing time
- Cost of late software product shipment
- Receivables potentially affected

Price of Non-Compliance (PONC)

External **Low Visibility** Costs

- Cost of decisions made based on bad data
- Lost Market Share
- Technical support calls
- Investigation of customer complaints
- Refunds and recalls
- Coding / testing of interim bug fix releases
- Shipping of updated product
- Added expense of supporting multiple versions of the product in the field
- PR work to soften drafts of harsh reviews
- Lost sales
- Lost customer goodwill – Reputation for producing buggy software
- Discounts to resellers to encourage them to keep selling the product
- Warranty costs
- Liability costs
- Government investigations – if company subject to regulatory rules
- Penalties
- All other costs imposed by law
#1 Cost of Quality Evaluation Pitfall: External or Low Visibility Costs

- Typically easy to overlook or minimize because hard to quantify
- Internal Costs often the only visible part of the PONC analysis - “iceberg effect”
- Also referred to as “SOFT COSTS” because hard to quantify

The 1-10-100 Quality Cost Rule

- Catching and Fixing Bugs at Your Workstation
- Repairing the Damage Caught by Customers
- Catching and Fixing Bugs Internally, but After they have left your Work Area
The Relative Cost of Fixing Bugs

The 1-10-100 Rule
Cost of Quality WITHOUT QA/QC

- 125 Bugs / Year, **80% caught by developers**
- $100.00 to fix a bug at developer level
- **NO QA/QC COST**

<table>
<thead>
<tr>
<th>Cost of resolving bug</th>
<th>Cost of Resolving a Bug Immediately</th>
<th>Cost of Resolving a Bug at QC</th>
<th>Cost of resolving a Bug once it reached the users</th>
<th>Total Cost of Bug Fixes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10-100 Rule</td>
<td>1</td>
<td>10</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Distribution of 125 Bugs</td>
<td>100 x $100 x 1</td>
<td>0 x $100 x 10</td>
<td>25 x $100 x 100</td>
<td>100</td>
</tr>
<tr>
<td>Distribution of Costs @ $100/bug</td>
<td>(100x$100 x 1)</td>
<td>(0x $100 x 10)</td>
<td>(25 x $100 x 100)</td>
<td>$250,000</td>
</tr>
<tr>
<td>NO QA/QC TEAM COSTS</td>
<td></td>
<td></td>
<td></td>
<td>$0</td>
</tr>
<tr>
<td>Total Cost</td>
<td></td>
<td></td>
<td></td>
<td>$260,000</td>
</tr>
</tbody>
</table>
The 1-10-100 Rule
Cost of Quality WITH QA/QC

- 125 Bugs / Year, 80% caught by developers
- $100.00 to fix a bug at developer level
- $75,000 QA/QC Costs/Year, 80% QC Catch

<table>
<thead>
<tr>
<th>Cost of resolving bug</th>
<th>Cost of Resolving a Bug Immediately</th>
<th>Cost of Resolving a Bug at QC</th>
<th>Cost of resolving a Bug once it reached the users</th>
<th>Total Cost of Bug Fixes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10-100 Rule</td>
<td>1</td>
<td>10</td>
<td>100</td>
<td>$155,000</td>
</tr>
<tr>
<td>Distribution of 100 Bugs</td>
<td>100 x $100 x 1</td>
<td>20 x $100 x 10</td>
<td>5 x $100 x 100</td>
<td>$80,000</td>
</tr>
<tr>
<td>Distribution of Costs @ $100/bug</td>
<td>(100 x $100 x 1)</td>
<td>$10,000</td>
<td>(20 x $100 x 10)</td>
<td>$20,000</td>
</tr>
<tr>
<td>+ Cost of QA/QC Team</td>
<td>$75,000</td>
<td>$75,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Cost</td>
<td></td>
<td></td>
<td></td>
<td>$155,000</td>
</tr>
</tbody>
</table>

Quality Assurance
Return On Investment (ROI)
Internal View

- Cost of quality without QA team: $260,000
- Cost of quality with QA team: $155,000
  
  $155,000

  Difference: $105,000

Money Spent on QA/QC for one year: $75,000
Money Saved with QA/QC for one year: $105,000

ROI using the 1-10-100 Rule for 1 year: 140%
The Quality Cost Curve

The Visibility Curve

The aim:
Keep that Line
As flat as possible
The Ford Pinto Case – Part 1

"The Pinto was not to weigh an ounce over 2,000 pounds and not cost a cent over $2,000."

The Ford Pinto Crash Data

Rear end Crash > 25 MPH:
Gas Tank Fire

Rear end Crash > 40 MPH:
Gas Tank Fire
+ DOORS JAMMED SHUT!
The Ford Pinto Case Part 3

compress.mov

http://www.youtube.com/watch?v=rcNeorjXMrE

The Danger of QA Cost Analysis:
The Ford Pinto Case Part 2

Ford’s Cost/Benefit Analysis
Relating to Pinto Model Rear-End Crash

The “BENEFIT”
(INTERNAL Cost Of Non-Compliance)

<table>
<thead>
<tr>
<th>Description</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gas tank related accidents</td>
<td>180 burn deaths, 180 serious burn injuries, 2100 burned vehicles</td>
</tr>
<tr>
<td>Unit Cost</td>
<td>$200,000 per death, $67,000 per injury, $700 per vehicle</td>
</tr>
<tr>
<td>Total Cost</td>
<td>$49.5 million</td>
</tr>
</tbody>
</table>

The “COST”
(Cost Of Compliance)

<table>
<thead>
<tr>
<th>Description</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recalling 11 million cars, 1.5 million light trucks to fix vehicles with this model of gasoline tank</td>
<td></td>
</tr>
<tr>
<td>Unit Cost</td>
<td>$11 per car, $11 per truck</td>
</tr>
<tr>
<td>Total Cost</td>
<td>(11,000,000 + 1,500,000) x $11 = $137 million</td>
</tr>
</tbody>
</table>
## QA Cost Analysis: What did Ford Miss?

| The “BENEFIT” or VISIBLE Cost of Non-Compliance | Gas tank related accidents - 180 burn deaths, 180 serious burn injuries, 2100 burned vehicles  
Unit Cost -- $200,000 per death, $67,000 per injury, $700 per vehicle  
Total Cost: - 180 x ($200,000) + 180 x ($67,000) + 2100 x ($700)  
Total: $49.5 million |
|---|---|
| The “INVISIBLE” or EXTERNAL Cost of Non-Compliance | - State of Indiana v. Ford Motor Co: Ford First American corporation ever indicted or prosecuted on criminal homicide charges  
- Lawsuits + Court Costs  
- Production stopped 5 months after trial  
- Lost Reputation  
- Small car market share lost  
Total: $BILLIONS, not millions! |

### The Danger of QA Cost Analysis: Missing the EXTERNAL COSTS!

<table>
<thead>
<tr>
<th>The VISIBLE or “INTERNAL” Cost Of Non-Compliance Recognized by Ford and labeled as “BENEFIT” [of not doing anything]</th>
<th>Total: $49.5 million</th>
</tr>
</thead>
<tbody>
<tr>
<td>The INVISIBLE or “EXTERNAL” Cost of Non-Compliance NOT RECOGNIZED BY FORD [of not doing anything]</td>
<td>Total: $BILLIONS</td>
</tr>
</tbody>
</table>

VS.

| The “COST” (Cost Of Compliance – doing the $11 repair) | Total: $137 million |
Part 2
Software Quality Principles for Managers 101

Software Quality

Quality Control / Quality Assurance

Pareto Rule

Quality Model Considerations
Quality Control / Quality Assurance

ISO Definition:

**Quality Control:**
The operational techniques and activities that are used to fulfill requirements for quality

**Quality Assurance:**
All those planned and systematic activities implemented to provide adequate confidence that an entity will fulfill requirements for quality

Quality Control

Characterized by:

– Tactical in nature
– Technical skills
– Attention to detail
– Front-line Quality Checking Activity
Quality Control Outputs

• Crossed-off Checklist - Pass/Fail

If Fail:

• Recommend Corrective Actions if needed
• Document Defects in Bug Track

Quality Assurance

Characterized by:

– Strategic: important or essential in relation to a plan of action
– Quality Planning
– Consistency of measurements
– What can be improved in the future?
“Bug Track” Considerations

• Document bugs properly – data, circumstances, screens prints, library lists, sequence of events etc.

• Categorize the bugs reported
  – What type of bug (data? Formula? Screen Flow?)
  – From what module?
  – Using what programming language?
  – Batch or Interactive Processing?
  – How much time has been spent on QC?
  – Was this a Design bug?
  – How much time has been spent on the Fix (if applicable)

QA Bang-for-the-Buck

The Pareto Rule (80/20) Rule

• Vilfredo Pareto (1848 – 1923) : “80% of the land in Italy is owned by 20% of the population”

QA Application of the 80/20 rule

• 80% of customer complaints arise from 20% of your products or services.
Pareto Chart Example:
Credit Application Rejection Reasons

20% of possible causes

80% Of the problems

Where do Defects Originate?

Ranking:
1. Incomplete Requirements
2. Ambiguous Requirements
3. Code Logic Defects
4. Defect Handling
5. Wrong Requirements

Source: Quality Assurance Institute
QA Bang-for-the-Buck
Discovering your own 80/20 Rules

• Ensure you know your bug track database

• Categorize problems in a meaningful way
  – By type of problem
  – By module
  – By time spent repairing

• Use your bug track database to find the 80/20’s
  – SQL
  – Microsoft XL

• Find where improving quality immediately will deliver best return for the QA investment

Find your Worst Offenders
SQL Example

```sql
SELECT MODULE, COUNT(*) MODULE_COUNT
FROM BUGTRACK_DB
GROUP BY MODULE
```

<table>
<thead>
<tr>
<th>MODULE</th>
<th>MODULE_COUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>GL</td>
<td>25</td>
</tr>
<tr>
<td>SHOP_FLOOR</td>
<td>7</td>
</tr>
<tr>
<td>SALES_REPORTING</td>
<td>3</td>
</tr>
</tbody>
</table>
Quality Management Systems, Methodologies
Worthwhile Reading

Quality Control Flow
Traditional Model: “The Waterfall”

- Project Requirements
- Software Developers
- Software
- Quality Control
**Waterfall Quality Cost Concentration**

![Waterfall Diagram](source: Quality Assurance Institute)

Source: Quality Assurance Institute

---

**The “V” Diagram New Flow**

QA/QC is applicable at ALL stages of software production:

- **Software Building**
  - Define Requirements
  - Verify Requirements
  - System Design
  - Verify Design
  - Code System
  - Verify Code

- **Pro-Active**
  - Validate Design
  - Integration Test
  - Validate Code
  - Unit Test

- **Quality Assurance**
  - Validate Business Needs
  - Acceptance Test

- **Plan QC Activities**
  - Quality Control Activities

Source: Quality Assurance Institute
ISO 20,000 Quality Standard and ITIL

- Developed to describe the capability of software contractors to provide software on time, within budget, and to acceptable standards
- Often used by Government or large companies

Capability Maturity Model (CMM)

- Method for Evaluating the Maturity of an Organization – 5 Levels
  1. Initial – Follows little or no rules
  2. Repeatable – Disciplined Process
  3. Defined – Standardized Disciplined Process
  4. Managed – Using precise measurements
  5. Optimizing – Quantitative feedback, continuous improvement
Quality Planning Resources: Six Sigma

- Origin of Six Sigma

  + or – 6 Standard deviations (sigma) from the mean

  **6 Sigma:** 3.4 defect/million

  By contrast:

  **3 Sigma:** 2,700 defects/million

More on Six Sigma at

http://www.isixsigma.com/
http://www.ge.com/sixsigma/

QC/QA Web Resources

http://satc.gsfc.nasa.gov/assure/agbsec3.txt
http://home.att.net/~iso9k1/tqm/tqm.html
http://www.isixsigma.com/
http://www.ge.com/sixsigma/
http://www.badsoftware.com/qualcost.htm
http://www.kaner.com/qualcost.htm
http://www.extremeprogramming.org/map/code.html
Quick Bugtrack Starter: BUGZILLA

- Bugzilla (bugtrack) [http://www.bugzilla.org/](http://www.bugzilla.org/)
- Used by
  - AMD
  - McGraw Hill Higher Education
  - Motorola
  - France Telecom
  - University of Minnesota
  - Indian Institute of Astrophysics

Points to Remember

- Sell the QA/QC VALUE PROPOSITION:

- Explain the Value Proposition
  - 1-10-100 Rule

- Explain the Quality Process
  - Why Requirements are critical
  - QA vs QC
  - Quantified knowledge can lead to improvement
Questions