Using The Cost of Quality Approach for Software

May, 1999

Herb Krasner
President, Krasner Consulting
hkrasner@cs.utexas.edu
State of the Union in SW Quality

• experienced programmers are producing software with about 100 defects per 1000 SLOC
  - better commercial companies will try to remove at least 95% of these before delivery
• companies are routinely delivering systems of dubious quality
  - although there is much useful software, there is also much useless software
  - the US average for delivered quality is now about 4 Sigma
• successful large project performance rate is still low (~25%) but improving
  - runaway or failed projects are still common (e.g. the cost of canceled projects in 1995 for the US IT industry was estimated at $81B)
  - cost, schedule and functionality are under control for the CMM community; the CMM elite are way ahead of the pack
  - quality and customer satisfaction are the next big hurdles
• Y2K and other developments are pressing the issue of licensing and certification of SWEs
Five-Star Software Quality - Perspectives -

Customer Satisfaction
- Conformance to requirements/needs
- Customer and user satisfaction levels
- Fitness for use

Properties ("illities")
- Usability
- Reliability (MTBF)
- Portability
- Maintainability
- Complexity

Value to Stakeholders
- Profit
- Personnel satisfaction

Development Effectiveness
- Schedule & budget performance
- Features & functions delivered
- Process quality

Defectiveness
- Defect levels
- Defect severity
- Defect removal
Business Context for CoSQ

Improve Business Success Through Better Quality Software

Questions

How much does SW Quality Cost?
What are the benefits of Quality software?
How good is our SW Quality?

Measures

CoSQ & related ratios
reputation
market share
customer satisfaction
profit
time to market
problems
defectiveness
Definitions

Software Quality Costs
• a measure of the costs specifically associated with the achievement or non-achievement (nonconformance) of software product quality - encompassing all requirements established by the company, its customer contracts and society
  – Requirements - generally includes: wants, needs, constraints, etc. for both the product and related services
  – Nonconformance - generally, a deviation in one of the work products (e.g. SDP, SQAP, SCMP, SRS, SDD, STP, code, etc.) with respect to understood objectives, requirements, constraints and/or standards.
• Example terms associated with CoSQ
  – negative -> rework, wasted efforts, unacceptable software
  – positive -> preventing defects, validated software
• based on an emerging theory of the economics of software quality, reflecting the differences from manufacturing CoQ
• Non quality costs are the “normal” costs of creating SW
Comparative Cost of SW Quality

Relative Cost to Fix a Serious Defect

- **DPP & QIP in place**: 1
- **company finds & fixes it prior to release**: 10X
- **customer finds it in fielded system**: 100X
- **it leads to litigation**: ???

DPP = defect prevention process
QIP = quality improvement program
Cost of Software Quality
- Unvalidated Economic Theory -

![Graph showing the relationship between quality metric and costs](Image)

- Quality Metric
  - 0%
  - 100% (perfect)
- Non-conformance costs
- Quality achievement costs

CoSQ
CoSQ Model Categories - Overview

Software Production

CoSQ

- preventing poor quality (prevent nonconformances)
- dealing with nonconformances
- appraising quality (detecting nonconformances)

Software Creation

- e.g. quality goals, stds, training
- e.g. problem fixes, rework
- e.g. inspections, testing, SQA

Cost of SW Quality - 5/99 - Slide 8
<table>
<thead>
<tr>
<th>Major Category</th>
<th>Subcategory</th>
<th>Definition</th>
<th>Typical sub-items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dealing with nonconformances</td>
<td>External</td>
<td>quality problems detected prior to shipment</td>
<td>Pre-release defect mgt., rework, re-reviews, retesting, etc.</td>
</tr>
<tr>
<td></td>
<td>Internal</td>
<td>quality problems detected after shipment</td>
<td>Post release technical support, warentee work, recalls, litigation, etc.</td>
</tr>
<tr>
<td>Appraising the level of quality</td>
<td>Discovering the condition</td>
<td>discovering the level of nonconformances</td>
<td>Testing, SQA, inspections, reviews, etc.</td>
</tr>
<tr>
<td></td>
<td>Assuring the achievement</td>
<td>quality control gating</td>
<td>Product audits, go/no decisions to proceed, release to distribution</td>
</tr>
<tr>
<td>Preventing poor quality from occurring</td>
<td>Project/process interventions</td>
<td>TQM, PIP, SPI, org. learning, canceling projects</td>
<td>Training, process improvements, metrics collection and analysis</td>
</tr>
<tr>
<td></td>
<td>Quality basis mgt.</td>
<td>quality defn., goals, stds, thresholds, tradeoff analyses</td>
<td>Defining release criteria for acceptance, quality stds., quality goal model defn</td>
</tr>
</tbody>
</table>
CoSQ Can be Used to:

- provide cost/benefit justification for improvement initiatives
- provide cost data to demonstrate the relationship of employee efforts to the bottom line
- provide a basis for budgeting the quality management and assurance functions
- identify quality improvement candidates through causal analysis
- compare proposed process improvements and identify the most cost effective ones
- provide a (one) measure to compare the success of various projects, and/or organizations
- tune the quality costs on a particular project by altering the process prior to, or even in situ
- determine the potential cost/risk impact of specific quality tradeoff decisions on specific projects
Relating CoSQ and CMM Level

Cost as a Percent of Development

SEI CMM Levels

Legend
- TCoSQ
- Int NONC
- Ext NONC
- Appraisal
- Prevention

RES Cost Model Categories:

- **Nonconformance** - cost of rework
- **Appraisal** - cost of reviews, audits and testing
- **Prevention** - cost of preventing nonconformances
  - included their improvement initiative investment
- **Performance** - cost of building it right the first time (i.e. creation)

Changes experienced in average % of project time by cost type

<table>
<thead>
<tr>
<th></th>
<th>Performance</th>
<th>Nonconformance</th>
<th>Appraisal</th>
<th>Prevention</th>
</tr>
</thead>
<tbody>
<tr>
<td>1988</td>
<td>34%</td>
<td>44%</td>
<td>15%</td>
<td>7%</td>
</tr>
<tr>
<td>1990</td>
<td>55%</td>
<td>18%</td>
<td>15%</td>
<td>12%</td>
</tr>
<tr>
<td>1992</td>
<td>66%</td>
<td>11%</td>
<td>-</td>
<td>23%</td>
</tr>
<tr>
<td>1994</td>
<td>76%</td>
<td>6%</td>
<td>-</td>
<td>18%</td>
</tr>
</tbody>
</table>

see Haley, et. al., 1996 for details
Snapshot of a CMM Level 1 Performer

Legend
- Prevention
- Appraisal
- NONC
- Creation

% of total development cost (resources)

Current CoSQ Profile
Snapshot of a CMM Level 3 Performer

Legend
- Prevention
- Appraisal
- NONC
- Creation

% of Total Costs

This Year

<table>
<thead>
<tr>
<th></th>
<th>Prevention</th>
<th>Appraisal</th>
<th>NONC</th>
<th>Creation</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>49</td>
</tr>
</tbody>
</table>
CoSQ Improvement - New Leaders

- General Dynamics Defense Systems
- Motorola Cellular Systems
- Qualcomm
- Honeywell Building Control Systems
- Are there others out there ???
Honeywell BSCE Cost of Software Quality

Legend
- Prevention
- Testing
- Rework
- Total

1995
- Prevention: 1.16
- Testing: 1.023
- Rework: 0.685
- Total: 3.316

1996
- Prevention: 0.635
- Testing: 2.462
- Rework: 0.635
- Total: 3.782

Legend:
• 30% decrease in CoSQ
• Achieved CMM Level 2
• Implemented design & code reviews
• Earlier testing

• 80 person org.
• Do advanced building control systems
• In Germany

from D. Houston, CoSQ: Selling SPI to Managers, 1998, ASQ SW Quality Journal
Conclusions

• CoSQ is a useful technique whose time has come - because it uses the common metric of $\$, it helps unify business and technical decisions about software quality
• CoSQ will make the economics of software quality visible in your organization - thus allowing quality to participate in decisions equally with cost and schedule concerns
• CoSQ has been used successfully to measure the impact (ROI) of organizational software improvement programs
• CoSQ can be used to empower, as well as measure, an effective software quality improvement program
• Introducing CoSQ into your organization could cause a culture clash about the value of software quality
• CoSQ represents the “non normal” process which is usually hidden by conventional tracking mechanisms, possibly making it harder to implement
CoSQ Vs Performance Improvement

- SW Quality vs Business Performance
- Baselines vs Improvement Multiplier
- Years on the X-axis from 0 to 7
- Improvement Multiplier on the Y-axis from 0 to 10
CoSQ Reading References

- Krasner, H., 1997; The Cost of Software Quality: Empowering Improvement, Keynote Speech, 7th International Conference on Software Quality, see www.utexas.edu/coe/sqi/archive