Quicker and Better Quality Improvement Business Cases with Bayesian Belief networks and Six Sigma

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA  15213

Ben Linders, Affiliate SEI / Ericsson Quality Mgr
Bob Stoddard, Senior Member Tech Staff SEI

E-SEPG: Monday June 11, 2007
Contents / Agenda

Introduction
Six Sigma Methods
Exercise 1
Defect Modeling
Exercise 2
Conclusions
Introduction
Quality improvement needed in many organizations

Business case required

- Identification of problem areas
- Selected improvement
- Quantified costs & benefits

Problem: No data available

- Measurement programs are costly
- Long lead time
Solution

Requirements

- Value/result driven
- Comprehensible, easy to use
- Objective & reliable
- Industry Standard Compatible (Benchmarking)
- Re-use best practices

Technologies

- Six Sigma
- GQIM, Balanced Scorecard
- Bayesian Belief Networks
- Cost of Quality, Root Cause Analysis
Two step approach

Quality Factor Model

- Expert opinion, extended with data
- Quick Quality Scan
- Rough Prediction Fault Slip Through
- Improvement Areas

Defect Prediction Model

- Data, tuned with expert opinion
- Detailed Prediction Fault Slip Through
- Improvement Business Case
Collaboration

Ericsson Netherlands: 
Market Unit Northern Europe & Main R&D Center

R&D: Value Added Services
- Strategic Product Management
- Product marketing & technical sales support
- Provisioning & total project management
- Development & maintenance
- Customization
- Supply & support

+/- 1300 employees, +/- 350 in R&D

SEI Pittsburgh, PA: 
Software Engineering Measurement & Analysis

Modern Measurement Methods
- Goal Driven Measurement
- Managing Projects with Metrics
- Measuring for Performance-Driven Improvement -I, -II
- Understanding CMMI High Maturity Practices
- Client Support & Research
- Training Development & Delivery

Software Engineering Institute | Carnegie Mellon
Ben Linders & Bob Stoddard
June 11, 2007
© 2006 Carnegie Mellon University
Affiliate Assignment

Joint effort: Ericsson (Ben Linders) and SEI (Bob Stoddard)

- Time, money, materials
- Knowledge & experience

Deliverables Ericsson

- Defect data & benchmarks
- Improved decisions skills
- Business case & Strategy 2007:
  - Early phases: Improvements
  - Late test phases: Reduction

Research contribution

- Apply Six Sigma business cases
- Verify technology (CoQ, RBT, FST, etc)
Six Sigma Methods
DMAIC Roadmap

Define
- Define project scope
- Establish formal project

Measure
- Identify needed data
- Obtain data set
- Evaluate data quality
- Summarize & baseline data

Analyze
- Explore data
- Characterize process & problem
- Update improvement project scope & scale

Improve
- Identify possible solutions
- Select solution
- Implement (pilot as needed)
- Evaluate

Control
- Define control method
- Implement
- Document
The collaboration included an implementation of DMAIC to reduce Fault Slip Thru. This tutorial highlights the Analyze and Improve phase activities.
Basic Statistical Prediction Models

- **ANOVA & MANOVA**
  - Y (Continuous)
  - X (Discrete)

- **Chi-Square & Logit**
  - Y (Discrete)
  - X (Discrete)

- **Correlation & Regression**
  - Y (Continuous)
  - X (Continuous)

- **Logistic Regression**
  - Y (Discrete)
  - X (Continuous)
Example ANOVA Output

Escaping Defects versus Quality Check Method

We predict a range of escaped defect density for each type of quality check.

One-way ANOVA:

Escaped Defect Density versus Quality Check

<table>
<thead>
<tr>
<th>Quality Check</th>
<th>DF</th>
<th>SS</th>
<th>MS</th>
<th>F</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>330</td>
<td>238.306</td>
<td>0.722</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>334</td>
<td>377.825</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

S = 0.8498 R-Sq = 36.93% R-Sq(adj) = 36.16%

<table>
<thead>
<tr>
<th>Level</th>
<th>N</th>
<th>Mean</th>
<th>StDev</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Test</td>
<td>94</td>
<td>3.7298</td>
<td>0.6768</td>
</tr>
<tr>
<td>Inspection</td>
<td>10</td>
<td>4.5164</td>
<td>0.8615</td>
</tr>
<tr>
<td>Walkthrough</td>
<td>88</td>
<td>4.8568</td>
<td>0.8417</td>
</tr>
<tr>
<td>Informal w/Peer</td>
<td>37</td>
<td>5.6081</td>
<td>1.1186</td>
</tr>
<tr>
<td>Email Comments</td>
<td>6</td>
<td>6.6500</td>
<td>1.2755</td>
</tr>
</tbody>
</table>
Example Regression Output

Regression Analysis: Defect Density versus ReqsVolatility, YearsDomainExperience

The regression equation is
Defect Density = 0.484 + 0.480 ReqsVolatility - 0.0242 YearsDomainExperience

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Coef</th>
<th>SE Coef</th>
<th>T</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>0.48367</td>
<td>0.03957</td>
<td>12.22</td>
<td>0.000</td>
</tr>
<tr>
<td>ReqsVolatility</td>
<td>0.47963</td>
<td>0.09511</td>
<td>5.04</td>
<td>0.000</td>
</tr>
<tr>
<td>YearsDomainExperience</td>
<td>-0.024215</td>
<td>0.001941</td>
<td>-12.48</td>
<td>0.000</td>
</tr>
</tbody>
</table>

\[ s = 0.00893207 \quad R-Sq = 85.9\% \quad R-Sq(adj) = 84.8\% \]

Analysis of Variance

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>SS</th>
<th>MS</th>
<th>F</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>2</td>
<td>0.0126076</td>
<td>0.0063038</td>
<td>79.01</td>
<td>0.000</td>
</tr>
<tr>
<td>Residual Error</td>
<td>26</td>
<td>0.0020743</td>
<td>0.0000798</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>28</td>
<td>0.0146819</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Use of Design of Experiments

Essentially a sophisticated method of sampling data to conclude relationships

Provides more confidence in possible cause-effect relationships

Enables us to define a small, efficient set of scenarios which we can then include in surveys of experts

Results help to populate relationships in the Bayesian Belief Network (BBN) model
Example of Design of Experiments

Welcome to Minitab, press F1 for help.

Fractional Factorial Design

Factors: 5 Base Design: 5, 8 Resolution: III
Runs: 8 Replicates: 1 Fraction: 1/4
Blocks: 1 Center pts (total): 0

* NOTE * Some main effects are confounded with two-way interactions.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Why Use Monte Carlo Simulation?

- Allows modeling of variables that are uncertain (e.g. put in a range of values instead of single value)
- Enables more accurate sensitivity analysis
- Analyzes simultaneous effects of many different uncertain variables (e.g. more realistic)
- Eases audience buy-in and acceptance of modeling because their values for the uncertain variables are included in the analysis
- Provides a basis for confidence in a model output (e.g. supports risk management)

“All Models are wrong, some are useful” – increases usefulness of the model in predicting outcomes
Crystal Ball uses a random number generator to select values for A and B.

Crystal Ball then allows the user to analyze and interpret the final distribution of C!

Crystal Ball causes Excel to recalculate all cells, and then it saves off the different results for C!
Why Use Optimization Modeling?

Partners with Monte Carlo simulation to automate tens of thousands of “what-ifs” to determine the best or optimal solution

Best solution determined via model guidance on what decisions to make

Easy to use by practitioners without tedious hours using analytical methods

Uses state-of-the-art algorithms for confidently finding optimal solutions

Supports decision making in situations in which significant resources, costs, or revenues are at stake
Several Example Tools

@RISK

The world's most powerful risk analysis tool. Take into account all possible scenarios using Monte Carlo simulation. Work directly in Excel, create presentation-quality graphs, use distribution fitting, and more!

@RISK for Project

Analyze cost and schedule risks in Microsoft Project using Monte Carlo simulation.

• STANDARD
• PROFESSIONAL
A Bayesian network is a probabilistic graphical model, also known as a Bayesian Belief Network (BBN) or belief network.

A Bayesian network is represented by a graph, in which the nodes of the graph represent variables, and the edges represent conditional dependencies.

The joint probability distribution of the variables is specified by the network's graph structure. The graph structure of a Bayesian network leads to models that are easy to interpret, and to efficient learning and inference algorithms.

From Wikipedia, the free encyclopedia
Nodes can represent any kind of variable, be it a measured parameter, a latent variable, or a hypothesis. They are not restricted to representing random variables; this is what is "Bayesian" about a Bayesian network.

Bayesian networks may be used to diagnose and explain why an outcome happened, or they may be used to predict outcomes based on insight to one or more factors.

From Wikipedia, the free encyclopedia
Example of Bayesian Belief Model

**STEP 1:**
Develop a model that depicts leading indicator nodes and/or root cause nodes for each node to be predicted. In this diagram, the flow is from left to right and thus, each child node (nodes with incoming lines) may be predicted with the parent nodes (nodes sending a line to a child node).

**STEP 2:**
Model the relationship between each “child” node and the associated “parent” nodes for the child using:

Regression & ANOVA with Objective Data
Design of Experiments with Subjective Data
Examples of BBN Tools

“AGENARISK” http://www.agena.co.uk/

Bayesian Network and Simulation Software for Risk Analysis and Decision Support

“NETICA” http://www.norsys.com/

NORSYS
SOFTWARE CORP.

NORSYS makes advanced Bayesian belief network and influence diagram technology practical and affordable.

“HUGIN” http://www.hugin.com/

.NETICA” http://www.norsys.com/

Hugin Training Course

Our Hugin courses in Bayesian networks, have now been scheduled for 2007. Join our next training course in Copenhagen scheduled for February 27th - March 1st.

White Paper
Defect Modeling
History Defect Modeling

2001
- Defect Model defined, pilot in first project

2002/2003
- Improved based on project feedback
- First release quality estimates
- Industrialize model/tool, use in all major projects

2004/2005
- Targets: Project portfolio management
- Process Performance & Cost of Quality

2006/2007
- Process Improvement Business Cases
  SW Engineering Economics, Six Sigma
- Defect Prediction
Project Defect Model

Why?
- to control quality of the product during development
- improve development/inspection/test processes

Business Value:
- Improved Quality
- Early risks signals
- Better plans & tracking
- Lower maintenance
- Save time and costs
- Happy customers!
Process Performance

Project Data

- Insertion Rates
- Detection Rates
- Defect Distribution
- Fault Slip Through
- Post Release Defects

Process View

- Performance of design & test processes
- Benchmarking
- Best Practices & Improvement Areas
Main value to gain:

- Increase appraisal effectiveness
- Decrease failure costs

Improve performance & Invest in Prevention

- Cost determinators, and their results
- Relationships between cost categories (ROI)
Economic Model

Understand costs of defects
Process & project performance
Dialog managers & developers
Use operational data
Manage under uncertainty & incomplete data

Technologies
- Cost of Quality
- Bayesian Belief Networks
- Real Options
- Lean Six Sigma
Figure 1: Roadmap for research in software engineering economics.
Quality Prediction

Current Model: Estimation

• Extrapolate past performance
• Based on inserted/detected defects
• Plan & track

Wanted: Prediction

• Causes of defects
• What if Scenarios
• Decision taking

All models are wrong
Some models are useful

Deming
Step 1: Quality Factor Model

Bayesian Belief Network

- Phases
- Quality Factors
- Expert opinion
- Prediction of Quality Impact

Managerial: Line, project & Process Management

Technical: Requirements, Design, Implementation, Inspection, Test
Step 1: Quality Prediction

Management Factors

Defect Insertion

Defect Detection
Step 1: Management Factors

Management Context for Technical Activities

Direct:
- Project Management
- Process Management

Indirect:
- Strategic & Operational Line Management
Step 1: Defect Insertion

Technical Activities where defects inserted

- Root Cause Analysis
- Defect Prevention
Step 1: Defect Detection

Technical Activities where defects detected

- Early Detection
- Economy of Test
- Release Quality
Step 2: Defect Causes & Effect

From Estimation to Prediction

- Design Process
  - Competence, skills
  - Tools, environment

- Test Process
  - Competence, skills
  - Test Capacity
  - Tools, environment

- Defects Inserted
  - (documentation, code)

- Defects Detected
  - (Inspection, test)

- Resident Defects in Design Base

- Detection Rate
  - Fault Slip Through
  - Defect Classification

- Defect Density

- Resident Defects in Delivered Product

- Defect Level

- (Un)happy customers

- Defect Density

- Process
- Inputs and outputs
- Influencing factors
- Measurement
Step 2: Defect Prediction

Fault Slip Through

Defect found in a (later) test phase that should have been found earlier

“Should”: More Cost effective (economical)

Predict Defect Reduction

- Determine process impact
- Simulate quality change
- Predict savings

Pilots

- Agile
- Model Driven Development
Conclusions
Conclusions

Two step approach

• Quality Factors
• Defect Prediction

Benefits

• Quick Improvement Scope
• Better Business Case: Value for Business

Future

• Pilot results
• Industry Standards
The Software Engineering Institute Affiliate Program provides sponsoring organizations with an opportunity to contribute their best ideas and people to a uniquely collaborative peer group who combine their technical knowledge and experience to help define superior software engineering practices.

Affiliates: [http://www.sei.cmu.edu/collaborating/affiliates/affiliates.html](http://www.sei.cmu.edu/collaborating/affiliates/affiliates.html)