Quantitative Software Management

Using Metrics to Develop a Software Project Strategy

Donald M. Beckett

QSM, Inc.
2000 Corporate Ridge, Suite 900
McLean, VA 22102
(360) 697-2640, fax: (703) 749-3795
e-mail: don_beckett@qsm.com
http://www.qsm.com
Outline

• Overview
• Measurement, Expense or Investment
• State of the Industry: Project Estimation
• Staffing and Schedule
• Understanding Trade-offs
• Conclusion
• Questions?
Overview

Does this sound familiar?
Measurement: Expense or Investment

- **Software measurement (and process improvement) are viewed as expenses: Overhead**
  - Lean, agile organizations want to reduce overhead
  - But, how do organizations become “lean & agile”?  
  - Measurement activities not directly related to revenue generation

- **Part of cost of doing business**
  - 3 – 5% on average
  - Project management averages 14%
Measurement: Expense or Investment

• What does software measurement provide?
  1. Knowledge of an organization’s capabilities
  2. Identifies patterns and trends (Strengths to leverage and weaknesses to correct)
  3. Insight into projects in time to make effective mid-stream corrections
  4. Ability to benchmark against competition or “the industry” in quality, productivity, and time to market
  5. Quantitative basis for evaluating project and organizational performance

• Improves ability to meet commitments, avoid pitfalls, and evaluate trade-offs
State of the Industry: Project Estimation

- **Software estimates are not** project plans

- **Estimates contain uncertainty about two key components:**
  - Scope of the requirements (project size)
  - Team productivity

- **Estimates define a range of possible outcomes**
  - Trade-offs can be examined
  - Impossible solutions identified
The Cone of Uncertainty

- Not enough information is available early in the development lifecycle to make accurate estimates
- Precision is not accuracy
Actual vs. Estimated Effort

Effort Growth

Percent

Smaller At Size Larger

Actual vs. Estimated Effort
Actual vs. Estimated Schedule

Schedule Growth

Percent

Smaller | At Size | Larger

Actual vs. Estimated Schedule
Actual vs. Estimated Size

Size Growth

Percent

0 10 20 30 40 50 60 70 80 90 100

Smaller At Size Larger

Actual vs. Estimated Size

© Quantitative Software Management, Inc. #10
Change Happens

- Software development is more of a discovery process than a manufacturing one

- Software projects tend to grow as the implications of requirements are better defined

- Change must be
  - Managed (Change control process)
  - Anticipated
  - Its implications quantified
In Summary

• Average schedule growth is 8%
• Average cost/effort growth is 16%
• Average size growth is 15%
• So how can we use this information to create more accurate estimates?
Modeling Increased Size

- **Create best project estimate based on proposed size**
  - Use historically based productivity
  - Account for project constraints (staff, effort, schedule)

- **Create estimate based on 15% size growth**
  - Does this account for projected schedule & effort growth?
500 FP Project

9.4 months duration
37 person months effort
50% probability
500 FP Project

Evaluate Probability of Current Estimate

Life Duration (Months) Risk Profile - Probability demo
<500 FP project>

Life Effort (PM) Risk Profile - Probability demo
<500 FP project>

Likely outcomes 10.2 months schedule, 43 effort months

© Quantitative Software Management, Inc. #15
15% Growth (575 FP)

- **Duration**: 10.2 months
- **Effort**: 46 person months
- **Cost**: $603.7K

Milestones:
- 0 - CSR
- 1 - SRR
- 2 - LLDR
- 3 - CUT
- 4 - E
- 5 - STC
- 6 - UAT
- 7 - FCR
- 8 - 99R
- 9 - 99.9R

46 person months effort
15% Growth (575 FP)

Averages close to numbers predicted for effort and schedule growth (10.2 duration and 43 staff months of effort)
Staffing & Schedule

What is “normal” variability?

Schedule varies by a factor of 3.5 from +1σ to -1σ

Effort varies by a factor of 8 from -1σ to +1σ
How Should Project Effort Be Expended
A Case Study

- 838 projects that had data reported for Analysis/Design as well as Construction and Test phases
- Average Effort applied to Analysis/Design = 20%
- 474 projects in the sample used <= 20% analysis/design effort
  - Average Analysis/Design Effort = 11%
- 364 projects in the sample used > 20% analysis/design effort
  - Average Analysis/Design Effort = 33%
- Size profiles of samples very similar
Observations

• **Projects with <20% effort in Requirements and Design**
  - Took 12% longer to complete
  - Averaged 5.6% more effort (median 24.4% greater)
  - Had an average staff 14.6% higher

• **But these projects did excel at one thing:**
  - Found 63.7% more defects in systems test
  - Had 127% more defects in the first 12 months after delivery
Understanding Trade-offs

Size = Effort^a \times \text{Time}^b \times \text{Productivity}

where \ a = \frac{1}{3} \ \text{and} \ b = \frac{4}{3}

Additional schedule has a much larger impact on a software project than increased effort
The Estimating Conundrum

Schedule / Effort Tradeoff

Uncertainty about Size and Productivity creates uncertainty about the Duration-Effort curve
Conclusions

• **Measurement is an integral part of management**

• **Information required to make precise estimates is unavailable at project start-up**
  – Estimate uncertainty decreases rapidly with more information

• **Project estimates historically understate effort, schedule, & size**
  – Estimating based on a larger size or at a higher assurance level can account for this

• **The trade-off between schedule & cost/effort is non-linear**
Conclusions

- Effort spent in Analysis & Design pays big dividends
  - Reduces overall project effort (cost$$$$)
  - Reduces overall project schedule
  - Improves project quality
Questions
?

© Quantitative Software Management, Inc. #25