Software Cost Estimating

Body of Knowledge

Estimation Maturity

Estimated training time: 60 minutes

Slides are used with permission from © Galorath Inc. All rights reserved.
Acknowledgements

• Esteban Sanchez (Author), Ton Dekkers, Dan Galorath and Brian Glauser (Reviewers)

• To Galorath Incorporated for authorizing the use of Galorath’s Estimation Maturity Framework and facilitating illustrative outputs of the SEER Models
Objectives

• Introduce the Students to the concept of Estimation Maturity:
  • Background and Definition
  • Importance
  • Key Concepts

• Present an Estimation Maturity Framework in detail
  • Foundation Concepts of the Framework
  • Estimation Maturity Levels

• Illustrate how to use the Estimation Maturity Framework to measure and improve the Estimation Process

• Show how to use the Maturity Assessment to justify investment in Estimation Process Improvement
Lesson Outline

• Introduction and Background
  • Definition of Estimation Maturity and Key Terms
  • Why is Estimation Maturity Important?

• Estimation Maturity Framework
  • Estimation Process Areas
  • Estimation Maturity Levels
    • Level 0 – No Practice
    • Level 1 – Ad-Hoc (Informal)
    • Level 2 – Introduction to Formal
    • Level 3 – Implemented
    • Level 4 – Refined
    • Level 5 – Continuously Improved

• Continuous Improvement
  • Using the model to continuously improve the Estimation Process
  • Justifying Organization Investment
Understanding Estimation Maturity

• An indication of the degree to which an organization has adopted estimation industry best practices to continuously improve its estimation process through consistent measurement and analysis

• Key Terms:
  • Industry Best Practices
  • Continuous Improvement
  • Measurement and Analysis

• Estimation Maturity is measured/rated against a framework or model (Similar to SEI Capability Model Maturity Integration)
Estimation Industry Best Practices

- Developed through experience and research
- Collaborative knowledge from multiple sources
  - Organizations
  - Academic
  - Literature
- Can be customized to fit a specific organization and its environment
Continuous Improvement

- Ongoing effort to improve a product, service or process

- Lessons learned and data are gathered at the end of the process and fed back for improvement and calibration

- Based on the traditional Deming Cycle

- The objective is to constantly learn and adjust from experience
Measurement & Analysis

• Measure and collect software metrics (Effort, Cost, Schedule, Size, Maintenance, Defects)
• Analyze the data to identify trends and correlations
• Identify gaps and adjust the estimation process accordingly
Why is Estimation Maturity Important?

Estimation Maturity is synonymous with:

- Adoption of Industry Best Practices – What has brought good results to others
- Formal sizing techniques and estimation methods (Function Points, parametric models, historical data collection)
- Consistent and scalable processes

Mature organizations are able to deliver better estimates:

- Accurate cost and schedule estimates
- Based on Industry Data - Justifiable
- Refined using historical data – Reflects the organization trend

A good estimate is a key condition for project success. Good estimates are achieved via a mature estimation process.
Consequences of Low Estimation Maturity

- Poor Implementations (quality is sacrificed)
- Insufficient Development Team - Emergency Staffing (Not very effective if project is already late)
- Unrealistic Milestones
- Cost overruns caused by underestimating project requirements
- Project Management with little or no control on the project (lack of an accurate baseline to track performance)
- Extra Management (Micromanagement and Bureaucracy)
- Team frustration (stress is generated and quality is compromised)
- Customer dissatisfaction

Business decisions made early with minimum knowledge and maximum uncertainty
Estimation Maturity Model

- **6 levels of Estimation Maturity**

  - **Level 0**: Informal or no estimating
  - **Level 1**: Direct Task Estimation, Spreadsheets, Ad Hoc Process
  - **Level 2**: Formal Sizing (e.g. function points), Direct Task Estimation, Simple model (Size * Productivity), Some measurement & analysis, Informal Process
  - **Level 3**: Formal Sizing, Robust Parametric estimation, Estimate vs. actual capture, Rigorous measurement & analysis, Parametric planning & Control, repeatable process
  - **Level 4**: Formal sizing, Repeateable process, Robust parametric estimating, Rigorous measurement & analysis, Parametric estimation with tracking & control, Process improvement via lessons learned
  - **Level 5**: Formal sizing, Repeateable process, Robust parametric estimating, Rigorous measurement & analysis, Parametric estimation with tracking & control, Continuous process improvement
1. Establish Estimate Scope
2. Establish Technical Baseline, Ground Rules and Assumptions
3. Collect Data
4. Estimate and Validate Software Size
5. Prepare Baseline Estimate
6. Review, Verify and Validate the Estimate (*)
7. Quantify Risks and Risk Analysis
8. Generate a Project Plan
9. Document Estimate and Lessons Learned (*)
10. Track Project Throughout Development

* BoE for Software 74R-13 can be used as a template for Estimate Validation and Documentation

The Ten Process Areas Framework is Galorath Inc. Proprietary. © 2017. All Rights Reserved.

© ICEAA
Estimation Maturity
• Organizations are rated on each Estimation Process Area

Image courtesy of Galorath Inc. © 2017. All Rights Reserved.
Rating Criteria

- Rating is done based on a series of well-defined evaluation criteria

<table>
<thead>
<tr>
<th>3. Collect data</th>
<th>1.7</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
<th>Level 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sizing Data is converted to a common language of interest if necessary</td>
<td>3- Implemented</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Core information is obtained using a form customized for each job</td>
<td>2- Some Formal</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameters not relevant to the current estimate are removed from estimate survey</td>
<td>1- Ad-Hoc</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Parameters available in provided documentation are removed from the estimate survey</td>
<td>5- Continual Imp.</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Collected data is grouped into relevant categories which are then assigned unique identifiers which describe the attribute (sizesizing, productive productive, etc.)</td>
<td>1- Ad-Hoc</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Collected data is further identified in terms of its description, whether it is required at the start of the estimate or whether it will evolve as the estimate proceeds, and from whom the information must be gathered.</td>
<td>3- Implemented</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Data is normalized via a well-documented process to a standard set of activities, phases, etc.</td>
<td>4- Refined</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Data points are compared to established metrics to determine whether it is reasonable</td>
<td>0- No Practice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Image courtesy of Galorath Inc. © 2017. All Rights Reserved.
Level 0 – No Practice

• No estimation process at all
• Or Estimation performed in an inconsistent manner

GUESSING is the most widely used estimation method
Estimating without a Process

• Estimate are a wild GUESS done by developers or Project Managers

• Inaccurate estimates reduce project success
  • Poor Implementations
  • Customer Dissatisfaction
  • Lack of credible baseline
  • Project is likely to overrun cost and/or schedule

• Organizations suffer from cancelled/failed projects, resulting in significant economical impacts

• Poor estimates/plans are a root cause of project risk
Level 1 – Ad-Hoc (Informal)

- Home-grown methods such as spreadsheets and Ad-Hoc processes
- Inconsistent and hardly repeatable
Basic Estimation Tribes

- **Napkins** – Ad-Hoc, hero driven estimators... past successes are legendary... napkin scribbling taken as gospel
- **Guts** – Estimates based on feelings/experiences. Trusted regardless of actual results
- **Home-Grown Spreadsheets** – Former Napkins and Guts translating tribal knowledge into spreadsheets
  - Bestows mathematical accuracy and empirical integrity on home-grown estimation algorithms
  - Tribal estimation knowledge can and does work – But it comes with high risk and cost
  - Rarely repeatable and sporadically consistent
  - Heroic energy is kept in reserve and used to mitigate risk
  - Knowledge is never institutionalized and never leaves Desktop PC files
  - Knowledge is gone when heroes retire or leave
Manual Estimates: Introduce Human Errors

- Manual Task Estimates yield **SIGNIFICANT** error
- Desire for “credibility” motivates **over-estimation**
  - Need to provide an estimate with high confidence – This results in “padding” or excessive contingencies
  - Must spend all available budget to be “Reliable”
- Technical pride sometimes causes **under-estimation**
  - Optimistic bias from developers and experts
  - Pressure from upper management or sales department to lower the numbers
  - Targets are often used as estimates
- Metrics collection can help overcome these issues
Level 2 – Introduction to Formal

- First steps in adopting a formal sizing technique
- Simple CERs (Cost Estimation Relationships) and primitive use of parametric models
- Processes are Informal and not institutionalized

Estimation Bias mitigation begins at this level by introducing a formal sizing approach
Many Sizing techniques are available nowadays.

Organizations must choose based on goals and project nature.

<table>
<thead>
<tr>
<th>Software Type in this column</th>
<th>Is best characterized by this sizing technique(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Lines</td>
</tr>
<tr>
<td>Traditional Information Technology</td>
<td>X</td>
</tr>
<tr>
<td>Embedded/Algorithmic Processing</td>
<td>X</td>
</tr>
<tr>
<td>Code Generator</td>
<td>X</td>
</tr>
<tr>
<td>COTS Integration</td>
<td>X</td>
</tr>
<tr>
<td>Non-line based</td>
<td>X</td>
</tr>
<tr>
<td>Cloud</td>
<td>X</td>
</tr>
</tbody>
</table>

Use ranges to best quantify size early.
Estimation should use more than Simple Productivity Measures

- Using simple Size over Productivity may not adequately forecast the effort for a new software
  - Unless the software is very similar
- Additional software characteristics should be considered to describe the software and its environment
  - Quality
  - Reuse
  - Testing and Quality Assurance
  - Staffing
  - Technology and Environment (e.g., Requirements Volatility)

Productivity measures are generally based on Size... So Low Maturity Organizations may lag here too
Level 3 – Implemented

- Processes are defined => Consistency and Repeatability
- Measurement and Analysis of Estimates vs Actuals
- Formal Sizing + Parametric Estimation = Key for better estimates

Estimation Bias mitigation is solid in this level with the combination of Formal Sizing and Robust Parametric Estimation.
Fundamental Metrics for Estimation, Planning and Control

- **Size**
  - Volume, mass
  - Units: SLOC, Function Points, Use Cases
  - New and Reused
  - COTS and Packages

- **Technology**
  - Productivity potential, Efficiency (e.g. tools and personnel)

- **Complexity**
  - Programming languages
  - Interfaces, algorithms, data

- **Time**
  - Duration, Schedule
  - Units: Calendar Months, Weeks

- **Effort**
  - Work, Labor
  - Units: Staff Months, Staff Hours

- **Cost**
  - Budget, Money
  - Units: $ or other currencies

- **Staffing**
  - Manpower loading
  - Units: FTEs, people

- **Defects**
  - Reliability, Quality
  - Units: Defect count, resolution time
Estimation Lifecycle
When do we build estimates?

- Traditional Estimate “Tollgates”
  - During Feasibility
  - At Concept
  - After Requirements
  - After Design
  - At the end (Post Mortem)

Estimation is not a one-time task. The estimate must be revised throughout the lifecycle as more (and better) information becomes available.

Image courtesy of Galorath Inc. © 2017. All Rights Reserved.
Estimation Lifecycle

When do we build estimates (Agile)?

- Agile Estimate Tollgates
  - Backlog Planning
  - Before each release
  - Before each sprint

Agile estimates are no different. The Development Method is part of the solution, not the problem.

Image courtesy of Galorath Inc. © 2017. All Rights Reserved.
Estimating Project Risk

- Level 3 (and above) Organizations Identify, Measure and Estimate Risk

Project

- Monte Carlo Simulation

Work Element

- Probabilistic Distribution (S Curve)
- Confidence Levels

Parameter (Inputs)

- Input Ranges
- Three Point Inputs
Estimation Processes are Defined

- Define and Document the estimation processes
  - Workflows
  - Checklists
  - Roles and Responsibilities
  - Entry and Exit Criteria
  - Include interactions with other areas of the organization
- Train and Inform the Users
- Ensure processes are followed (Estimation Quality Assurance)
Level 4 – Refined

- Estimation Processes and tools are defined throughout the organization (i.e. Institutionalized)
- Rigorous Measurement and Analysis
- Estimation Process improved via Lessons Learned and Data Collection
Lessons Learned Reviews

• Capture lessons at the end of an estimate and at the end of the project
• Document missing or incomplete information
• Capture risks, issues and problems addressed
• Document key decisions made during the estimation process – Capture both, the Good and the Bad
• Document dynamics that occurred during the estimation:
  • Interactions of the estimation team
  • Interfaces with stakeholders
  • Trade-offs made to address issues

Capture Lessons Learned ASAP while memories are still “Fresh”. Every software project is an opportunity to improve the estimation process.
Measurement & Analysis

• Collect Metrics (Effort, Cost, Duration, Size, Defects)
• Compare your estimates against the metrics (trend lines, correlations)
• Identify gaps and adjust the estimation process and tools accordingly
Using Data to compare against the Industry

• Measuring and Collecting data will also allow you to “see where you are” compared to the Industry
  • Understand the size of the applications and software portfolio
  • How much am I spending in Development and Maintenance compared to the Industry?
  • How good is my productivity when compared to the Industry?
  • Am I being Cost Efficient?
  • How is the quality of my software (defect density) compared to the Industry?
• Mature organizations need to be able to answer these types of questions

Improving performance is vital for survival. But... Performance can only be improved once it can be measured

© ICEAA Estimation Maturity 32
Level 5 – Continuously Improved

- Quantitative Targets are established based on the strategic goals of the organization
- Continuous Process Improvement is oriented towards these targets
- Detailed performance measures are collected and analyzed
- Total Cost of Ownership Estimates are used for Strategic Business Decisions
Total Cost of Ownership (TCO) Costs

- Software Development
- Software Maintenance/Ongoing Support
- IT Infrastructure & Services (this can represent 60+% of TCO)
Making Business Decisions via TCO and ROI

Quantify Benefits
- Savings
- Additional Revenue

Estimate TCO
- Development
- Maintenance
- Infrastructure

Determine if Project is Worth
- Sufficient ROI
- Aligned with Strategy
Core Metric: Value Provided by the Software

- Spend where you obtain the most value
  - Value = Savings or additional revenue due to implementing the software

- Software Fails to add value too often
  - Users enamored with the concept
  - Concept Deployed (somewhere else)
  - Little or No Value contributed to the company
  - Bad Assumptions: Saving one minute per day of employee filling in their time card represents huge savings per year

- Many projects never produce positive ROI

Level 5 Organizations can accurately measure ROI and make better business decisions
Continuous Improvement

• Every Project is an opportunity to improve the Estimation Process
• Collect Lessons Learned at the end of every estimate and also at project closure
• Build a repository of project actuals
• And more importantly: Do something with the data!
  • Calibrate the Estimation Process and Tools to better reflect historical trends
• Level 5 Organizations show up to 2% variability between estimates and actuals
Using the Estimation Maturity Model

- Assess Current State
- Define Goal
- Perform GAP Analysis
- Develop Implementation Plan
- Execute the Plan
- Capture Metrics/Data
- Self-Assessment
- Process Improvement Plan
- What needs to be improved (processes, tools, training)

Image courtesy of Galorath Inc. © 2017. All Rights Reserved.
Justifying Investment

• Develop a Business Case for the Organization

• Focus on two main aspects:
  • **The Problem:** What is wrong and how much is it costing us?
  • **The Solution:** What can be improved and how much will it save or how much extra revenue will it generate?

• Make sure to handle Stakeholder’s expectations:
  • Project Managers
  • Project Teams
  • Estimators
  • Executives (Decision makers)
  • Customers

This one is typically a stronger motivator for investors
What is wrong and how much is it costing us?

- **Project Overruns**
  - Statistics on failed/challenged projects
  - How much $ do they represent to the organization
- **Missed Milestones**
  - Impact of unachieved milestones
  - Dependencies with other projects
- **Quality**
  - Cost of technical debt
  - Customer dissatisfaction
  - Reputation
  - Low Maintainability
- **Team Moral**
  - Stress
  - Demotivation

Decision Makers want to see $ figures

Quantity and measure as much as you can to show $

Some aspects will remain Qualitative (i.e. Team Demotivation)
What can be improved and how much will it save/generate?

- Adoption of Formal Sizing Techniques
  - Function Points (any flavor, IFPUG, NESMA, COSMIC)
  - Use Cases
  - Lines of Code

- Adoption of Parametric Estimation Tools
  - Several options available in the Market (SEER, QSM, PRICE)

- Invest in Data Collection
  - Industry Data available Off-The-Shelf (ISBSG)
  - Organizational Data must be collected – But we need a process in place for this

- Process Definition
  - Workflows
  - Checklists
  - Criteria
  - Roles and Responsibilities

Again, the key is to assign a monetary value to each of these things
Remember to Address Stakeholder’s needs

• All Stakeholders are important, but the decision makers are the MOST important

• Apply the 80/20 Rule:
  • One or two Executives have the power to make the decision

• Focus on showing the value to the stakeholders and the Organization (this is better done using $s)
Conclusions

• An Estimate is a key component for Project Success
• Higher Estimation Maturity allows an organization to produce better estimates
• Estimation Maturity Improvement is an ongoing process
• Measurement and Data Collection are important contributors to Estimation Maturity Improvement
• The Business Case to justify investment in Estimation Maturity must focus on showing the value (ideally in $) to the organization
Summary of Key Points

- What is an estimate?

- Definition of Estimation Maturity
  - Industry Best Practices
  - Continuous Improvement
  - Measurement and Analysis
  - Maturity Levels

- Importance of Estimation Maturity

- Estimation Maturity Levels

- Importance of Measurement and Data Collection

- Using the Estimation Maturity Framework to justify investment
Bibliography


• Galorath, D & Evans, M (2006) Software Sizing, Estimation and Risk Management. When Performance is Measured Performance Improves


Thanks for attending this workshop

Estimation Maturity

Slides are used with permission from © Galorath Inc. All rights reserved.
Questions

Estimation Maturity

Slides are used with permission from © Galorath Inc. All rights reserved.