Toward Estimating DevOps Effort with COCOMO® II

Jim Alstad  USC Center for Systems and Software Engineering

COCOMO Forum
Westin Arlington Gateway
Arlington VA
November 6, 2018

Jim Alstad  310/344-0894  jalstad@usc.edu
Agenda

- Agenda:
  - The challenge
  - Estimation for development projects
  - Estimation for maintenance projects
The Challenge

- At this Forum last year, Jo Ann Lane challenged attendees to develop software cost estimating models that applied to newer types of software life cycles, such as DevOps.
- This is my attempt to step up to that challenge:
  - Basic idea: get historical projects to serve as a basis for estimating new projects.
  - I give details of my specific approach, which loosely is to calibrate an estimating equation based on the historical projects.
  - Also, I cite a different approach using the same basic idea.
Estimation for New Development

• Concept: Averaging across a group of historical projects can provide some value in estimating new projects
• Assumption: The organization has a size measure for each project, and a means of estimating size on a new project
• Method:
  – Gather a group of representative completed projects
  – For these projects, compute \( A = \text{average "productivity"} = \text{total effort / total size} \).
  – Determine a set \( \text{CD} \) of COCOMO cost drivers that:
    • Seem to vary between projects; and
    • Seem to have an influence on effort
    • (could determine by running a correlation on effort).
  – Then for a new project, the estimate is \( \text{EstEffort} = A \times \text{EstSize} \times \text{product (EMi)} \)
    where the \( \text{EMi} \) range across cost drivers in \( \text{CD} \) and represent their estimated ratings.
Advanced Estimation for New Development

• Used where the projects range widely in size & effort
  – Say, more than one order of magnitude

• Extended method:
  – In addition to CD, determine a set SF of COCOMO scale factors that:
    • Seem to vary between projects; and
    • Seem to have an influence on effort
    • (could determine by running a correlation on log(effort)).
  – Determine average exponent E such that, on the same group of projects:
    • $E = \text{average value of } \frac{\log(\text{Effort}) - \log(A)}{\log(\text{Size})}$
  – Then for a new project, the estimate is
    \[ \text{EstEffort} = A \times (\text{EstSize}^\text{Exp}) \times \prod \text{(EMi)} \]
    where \[\text{Exp} = E - \text{ExpBase} + \text{sum (SFi)}\],
    \[\text{ExpBase} = \text{sum over SF of Nominal scale factors} .\]
Estimation for Maintenance

• This approach builds on top of the Estimation for Development material above

• My material is based directly on COCOMO II maintenance estimation, per sections 2.2.7 and 2.5 of the COCOMO II book.

• Method:
  – Determine maintenance size (in SLOC) as:
    \[ \text{Size} = (\text{Size added} + \text{Size modified}) \times MAF \] [Eq. 2.9];
    \[ MAF = 1 + (SU \times UNFM / 100) \] [Eq. 2.10]
  – Apply Estimation for Development, with these adjustments on certain cost drivers (when relevant):
    • Don’t use SCED or RUSE cost drivers
    • If using RELY, use Table 2.41 to look up effort multipliers (instead of the usual Table 2.17)
Historical Note

• There is a (Web-based) estimation tool called Agile COCOMO II at http://sunset.usc.edu/cse/pub/research/AgileCOCOMO/AgileCOCOMOII/Main.html
  – However, I didn’t see any documentation for it.

• The tool seems to operate by looking at an existing project, requesting information about COCOMO ratings where the new project differs from the existing one, and then adjusting the estimated effort via COCOMO II parameter values.

• A major difference between Agile COCOMO and this proposal: this proposal uses averages across a group of projects.