

EXHIBIT 10

COCOMO II

Model Definition Manual

Version 2.1

© 1995 – 2000 Center for Software Engineering, USC



Copyright Notice

This document is copyrighted, and all rights are reserved by the Center for Software Engineering at the University of Southern California (USC). Permission to make digital or hard copies of part of all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation of the first page. Abstracting with credit is permitted. To copy otherwise, to republish, to post on Internet servers, or to redistribute to lists requires prior specific permission and/or fee.

Copyright © 1995 – 2000 Center for Software Engineering, USC

All rights reserved.

Warranty

This manual is provided “as is” without warranty of any kind, either express or implied, including, but not limited to the implied warranties of merchantability and fitness for a particular purpose. Moreover, the Center for Software Engineering, USC, reserves the right to revise this manual and to make changes periodically without obligation to notify any person or organization of such revision or changes.

1. Introduction

1.1 Overview

This manual presents two models, the Post-Architecture and Early Design models. These two models are used in the development of Application Generator, System Integration, or Infrastructure developments [Boehm et al. 2000]. The Post-Architecture is a detailed model that is used once the project is ready to develop and sustain a fielded system. The system should have a life-cycle architecture package, which provides detailed information on cost driver inputs, and enables more accurate cost estimates. The Early Design model is a high-level model that is used to explore of architectural alternatives or incremental development strategies. This level of detail is consistent with the general level of information available and the general level of estimation accuracy needed.

The Post-Architecture and Early Design models use the same approach for product sizing (including reuse) and for scale factors. These will be presented first. Then, the Post-Architecture model will be explained followed by the Early Design model.

1.2 Nominal-Schedule Estimation Equations

Both the Post-Architecture and Early Design models use the same functional form to estimate the amount of effort and calendar time it will take to develop a software project. These nominal-schedule (NS) formulas exclude the cost driver for Required Development Schedule, SCED. The full formula is given in Section 3. The amount of effort in person-months, PM_{NS} , is estimated by the formula:

$$PM_{NS} = A \times \text{Size}^E \times \prod_{i=1}^n EM_i$$

Eq. 1

$$\text{where } E = B + 0.01 \times \sum_{j=1}^5 SF_j$$

The amount of calendar time, $TDEV_{NS}$, it will take to develop the product is estimated by the formula:

$$TDEV_{NS} = C \times (PM_{NS})^F$$

Eq. 2

$$\text{where } F = D + 0.2 \times 0.01 \times \sum_{j=1}^5 SF_j$$
$$= D + 0.2 \times (E - B)$$

The value of n , the number of effort multipliers, EM_i , is 16 for the Post-Architecture model effort multipliers, EM_i , and 6 for the Early Design model. SF_j stands for the exponential scale factors. The values of A , B , EM_1, \dots, EM_{16} , SF_1, \dots, SF_5 for the COCOMO II.2000 Post-Architecture model are obtained by calibration to the actual parameters and effort values for the 161 projects currently in the COCOMO II database. The values of C and D for the

COCOMO II.2000 schedule equation are obtained by calibration to the actual schedule values for the 161 project currently in the COCOMO II database.

The values of A, B, C, D, SF₁, ..., and SF₅ for the Early Design model are the same as those for the Post-Architecture model. The values of EM₁, ..., and EM₆ for the Early Design model are obtained by combining the values of their 16 Post-Architecture counterparts; the specific combinations are given in Section 3.2.2.

The subscript NS applied to PM and TDEV indicates that these are the nominal-schedule estimates of effort and calendar time. The effects of schedule compression or stretch-out are covered by an additional cost driver, Required Development Schedule. They are also included in the COCOMO II.2000 calibration to the 161 projects. Its specific effects are given in Section 4.

The specific milestones used as the end points in measuring development effort and calendar time are defined in Section 6, as are the other definitions and assumptions involved in defining development effort and calendar time. Size is expressed as thousands of source lines of code (SLOC) or as unadjusted function points (UFP), as discussed in Section 2. Development labor cost is obtained by multiplying effort in PM by the average labor cost per PM. The values of A, B, C, and D in the COCOMO II.2000 calibration are:

$$\begin{array}{ll} A = 2.94 & B = 0.91 \\ C = 3.67 & D = 0.28 \end{array}$$

Details of the calibration are presented in Section 7, which also provides formulas for calibrating either A and C or A, B, C, and D to one's own database of projects. It is recommended that at least A and C be calibrated to the local development environment to increase the model's accuracy.

As an example, let's estimate how much effort and calendar time it would take to develop an average 100 KSLOC sized project. For an average project, the effort multipliers are all equal to 1.0. E will be set to 1.15 reflecting an average, large project. The estimated effort is $PM_{NS} = 2.94(100)^{1.15} = 586.61$.

Continuing the example, the duration is estimated as $TDEV_{NS} = 3.67(586.6)^{(0.28+0.2 \times (1.15-0.91))} = 3.67(586.6)^{0.328} = 29.7$ months. The average number of staff required for the nominal-schedule development is $PM_{NS} / TDEV_{NS} = 586.6 / 29.7 = 19.75$ or about 20 people. In this example, an average 100 KSLOC software project will take about 30 months to complete with an average of 20 people.

4. Schedule Estimation

The initial version of COCOMO II provides a simple schedule estimation capability similar to those in COCOMO 81 and Ada COCOMO. The initial baseline schedule equation for the COCOMO II Early Design and Post-Architecture stages is:

$$\text{TDEV} = [C \times (\text{PM}_{\text{NS}})^{(D+0.2 \times (E-B))}] \times \frac{\text{SCED}\%}{100} \quad \text{Eq. 14}$$

where $C = 3.67$, $D = 0.28$, $B = 0.91$

In Equation 14, C is a TDEV coefficient that can be calibrated, PM_{NS} is the estimated PM *excluding* the SCED effort multiplier as defined in Equation 1, D is a TDEV scaling base-exponent that can also be calibrated. E is the effort scaling exponent derived as the sum of project scale factors and B as the calibrated scale factor base-exponent (discussed in Sections 3.1). SCED% is the compression / expansion percentage in the SCED effort multiplier rating scale discussed in Section 3.2.1.

Time to Develop, TDEV, is the calendar time in months between the estimation end points of LCO and IOC for MBASE/RUP or SRR and SAR for Waterfall lifecycle models (see Section 6.2). For the waterfall model, this goes from the determination of a product's requirements baseline to the completion of an acceptance activity certifying that the product satisfies its requirements. For the MBASE/RUP model discussed in Section 6, it covers the time span between LCO and IOC milestones.

As COCOMO II evolves, it will have a more extensive schedule estimation model, reflecting the different classes of process models a project can use. The effects of reusable and COTS software; the effects of applications composition capabilities; and the effects of alternative strategies such as Rapid Application Development are discussed in [Boehm et al. 2000; also at <http://sunset.usc.edu/COOCMOII/suite.html>].