

USC COCOMO II

User's manual

University of Southern California

This manual is compatible with USC-COCOMO II. 2000 version 3.

Copyright Notice

This document is copyrighted, and all rights are reserved by University of Southern California. This document may not in whole, or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent.

Copyright © 1995-2003

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, 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.

Trademark Acknowledgement

USC has made every effort to supply trademark information about company names, products and services mentioned in this document. Trademarks indicates below were derived from various sources.

UNIX is a registered trademark of AT&T Bell laboratories

Motif is a trademark of Open Software Foundation, Inc.

FrameMaker and Frame Technology are registered trademarks of Frame Technology Corporation.

Sun Microsystem and Sun Workstation are registered trademarks, and OpenWindows, Sun-3, Sun-4, and SPARCstation are trademarks, of Sun Microsystems, Inc.

X Window System is a trademark of the Massachusetts Institute of Technology.

MS Window 95, 98, 2000, NT, XP are a trademark of Microsoft Corporation

Printed in the United States of America.

Acknowledgements

COCOMO 81, Version 1.0:

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Alfredo Arcilla, Joyce Balimbin, Gina Gaborno, Larry Klei, Robert Kosai, Deseree Moon, Jason Pan, Thomas Quayle, Isaiah Simmons, Scott Zechiel

COCOMO 81, Version 1.1:

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Ing-Jer Huang

COCOMO 81, Version 10.0

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – M. Susan Lane, Ping Luo, Lorna Zorman

COCOMO 2.0, Version 2.0

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Wiryadi Adidharma, Sen-Ho change, Shu-fen Cheng, Yu-Chuan Lin, Steve K. Luk, Shawne Robison, Tuan Ton

COCOMO 2.0, Version 2.0.5

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Thomas Majchrowski, Suppachai Sangtonkhamsuk, Lloyd Manglapus

COCOMO II, 1997.2

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Jongmoon Baik, Jungwon Park, James Chou, Dana Flora-Adams, SangHyun, and Eunsook Bang

COCOMO II, 1998.0

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Jongmoon Baik and CS655 students

COCOMO II, 2000.0

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Jongmoon Baik and Keun Lee

COCOMO II, 2000.2

Principal Investigator – Dr. Ellis Horowitz
Student Designers, Testers and programmers – Keun Lee

COCOMO II, 2000.3

Principal Investigator – Dr. Ellis Horowitz

Student Designers, Testers and programmers – Keun Lee

Note – This manual is sufficient for the X Windows, MS Windows 95/98/2000/NT/XP, and Java versions of USC COCOMO II. 2000.3 programs. Though there are some user interface differences between these systems, the core component of all versions is identical.

Some of the material used in this manual has been taken from Software Engineering Economics, by Barry Boehm, Prentice-Hall, with permission of the author.

Content

Chapter 1: Introduction 1

1.1 What is COCOMO?	1
1.1.1 Effort Estimation Equation	2
1.1.2 Schedule Estimation Equation	3
1.1.3 Scale Factors	3
1.1.4 Sizing methods	4
1.1.5 FP: Counting with Unadjusted Function Points	7
1.1.6 AAF: Adaptation Adjustment Factor	8
1.1.7 Effort Multipliers	8
1.2 Navigating COCOMO	9
1.3 Begin Using COCOMO	18
1.4 Running COCOMO: Windows, Sparc or Java	19

Chapter 2: File Menu 20

2.1 New	21
2.2 Project Load	22
2.3 Project Save	23
2.4 Project Save As	24
2.5 Model Load	25
2.6 Model Save	27
2.7 Model Save As	28
2.8 Make Report	29
2.9 Export	30
2.10 Save Screen	31
2.11 Print Screen	32
2.12 Print Preview	32
2.13 Print Setup	33
2.14 Exit	34

Chapter 3: Edit Menu 36

3.1 Add module	36
3.2 Clear All Module	36
3.3 Snapshot	37
3.4 Undo	39
3.5 Cut	39
3.6 Copy	40
3.7 Paste	40

Chapter 4: Parameters Menu 41

4.1 Product	42
4.2 Platform	42
4.3 personnel	43
4.4 Project	43
4.5 User Defined EAF	45
4.6 Early Design	46
4.7 Scale Factor	46
4.8 Equation	47
4.9 Person Month	49
4.10 Function Point	49

Chapter 5: Calibrate menu 50

5.1 File Load	51
5.2 File Save	51
5.3 File Save As	51
5.4 Project	53
5.5 Computer	54

Chapter 6: Phase Distribution 55

6.1 Waterfall	56
6.1.1 Project Phase Distribution	56
6.1.1.1 Overall Project Phase Distribution	56
6.1.1.2 Plans and Requirements Project Phase Distribution	59
6.1.1.3 Programming Project Phase	59
6.1.1.4 Product Design Phase	60
6.1.1.5 Integration and Test Project Phase	61
6.2 MBASE	68
6.2.1 Project Phase Distribution	68
6.2.1.1 Overall Project Phase Distribution	69
6.2.1.2 Inception Project Phase Distribution	69
6.2.1.3 Elaboration Project Phase	71
6.2.1.4 Construction Design Phase	71
6.2.1.5 Transition Project Phase	73
6.2.2. Module Phase Distribution	73
6.2.2.1 Overall Module Phase Distribution	75
6.2.2.2 Inception Module Phase Distribution	76
6.2.2.3 Elaboration Module Phase Distribution	76
6.2.2.4 Construction Phase Distribution	78
6.2.2.5 Transition Module Phase Distribution	78

Chapter 7: Maintenance 79

7.1 Project maintenance	81
7.2 Module Maintenance	85

Chapter 1: Introduction

1.1 What is COCOMO?

COCOMO (COConstructive COSt MOdel) is a screen-oriented, interactive software package that assists in budgetary planning and schedule estimation of a software development project. Through the flexibility of COCOMO, a software project manager (or team leader) can develop a model (or multiple models) of projects in order to identify potential problems in resources, personnel, budgets, and schedules both before and while the potential software package is being developed.

The COCOMO software package is based upon the software cost and schedule estimation model: COConstructive COSt MOdel version II (COCOMOII). This is the newly revised version of the original COConstructive COSt MOdel (COCOMO) first published by Dr. Barry Boehm in his book Software Engineering Economics, Prentice-Hall (1981), and Ada COCOMO (1989) predecessors. The current model is described in [Boehm et al. 1995]

The primary objectives of the COCOMOII.1998 effort are:

- To develop a software cost and schedule estimation model tuned to the life cycle practices of the 1990's and 2000's.
- To develop software cost database and tool support capabilities for continuous model improvement.
- To provide a quantitative analytic framework, and set of tools and techniques for evaluating the effects of software technology improvements on software life cycle costs and schedules.

The full COCOMOII model includes three stages. Stage 1 supports estimation of prototyping or applications composition efforts. Stage 2 supports estimation in the Early Design stage of a project, when less is known about the project's cost drivers. Stage 3 supports estimation in the Post-Architecture stage of a project.

This version of USC COCOMOII implements stage 3 formulas to estimate the effort, schedule, and cost required to develop a software product. It also provides the breakdown of effort and schedule into software life-cycle phases and activities from the original COCOMO manual. These are still reasonably valid for waterfall model software projects, but need to be interpreted for non-waterfall projects.

1.1.1 Effort Estimation Equation

(EQ 1-1)

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

, where $E = B + 0.01 \times \sum_{j=1}^5 SF_j$ and $PM_{\text{Auto}} = \frac{\text{Adapted SLOC} \times (AT/100)}{ATPROD}$

$$\text{Size} = (1 + \frac{REVL}{100}) \times (\text{New KSLOC} + \text{Equivalent KSLOC})$$

$$\text{Equivalent KSLOC} = \text{Adapted KSLOC} \times AAM \times (1 - AT/100)$$

$$AAM = \frac{AA + AAF \times (1 + [0.02 \times SU \times UNFM])}{100}, \text{ for } AAF \leq 50$$

$$= \frac{AA + AAF \times (SU \times UNFM)}{100}, \text{ for } AAF > 50$$

$$AAF = (0.4 \times DM) + (0.3 \times CM) + (0.3 \times IM)$$

Symbol	Description
A	Constant, currently calibrated as 2.94
AT	Percentage of components that are automatically translated
ATPROD	Automatic translation productivity
B	Scaling base-exponent for Effort that can be calibrated
E	Scaling exponent for Effort
EM	Effort Multipliers: RELY, DATA, CPLX, RUSE, DOCU, TIME, STOR, PVOL, ACAP, PCAP, PCON, APEX, PLEX, LTEX, TOOL, SITE, SCED
PM	Person Months effort from developing new and adapted code
PM _{Auto}	Person Months effort from automatic translation activities
SF	Scale Factors: PREC, FLEX, RESL, TEAM, PMAT
SLOC	Source Lines of Code
AA	Percentage of Assessment and Assimilation
CM	Percent Code Modified
DM	Percent Design Modified
IM	Percent of Integration Required for the Adapted Software
KSLOC	Thousands of Source Lines of Code
REVL	Percentage of Requirements Evolution and Volatility
SU	Software understanding (zero if DM = 0 and CM = 0)
UNFM	Programmer Unfamiliarity with software

1.1.2 Schedule Estimation Equation

Determine time to develop (TDEV) with an estimated effort, PM, that excludes the effect of the SCED effort multiplier:

(EQ 1-2)

$$TDEV = [C \times (PM_{NS})^F] \times \frac{SCED\%}{100}$$

$$F = D + 0.2 \times [E - B]$$

Symbol	Description
B	The scaling base-exponent for the effort equation
C	Schedule coefficient that can be calibrated
D	Scaling base-exponent for Schedule that can be calibrated
E	The scaling exponent for the effort equation
F	Scaling exponent for Schedule
PM _{NS}	Person Months estimated without the SCED cost driver (Nominal Schedule) and without PM _{Auto}
SCED%	Required Percentage of Schedule Compression relative to Nominal Schedule
TDEV	Time to Develop in calendar months

1.1.3 Scale Factors

Equation 1-2 defines the exponent, B, used in Equation 1-1. Table 1.1 provides the rating levels for the COCOMOII scale drivers. The selection of scale drivers is based on the rationale that they are a significant source of exponential variation on a project's effort or productivity variation. Each scale driver has a range of rating levels, from Very Low to Extra High. Each rating level has a weight, W, and the specific value of the weight is called a scale factor. A project's scale factors, W_i, are summed across all of the factors, and used to determine a scale exponent, B, via the following formula:

(EQ 1-3)

$$B = 0.91 + 0.01 \times \sum_{j=1}^5 SF_j$$

For example, if scale factors with an Extra High rating are each assigned a weight of (0), then a 100 KSLOC project with Extra High ratings for all factors will have $SF_j = 0$, $B = 1.01$, and a relative effort $E = 100^{1.01} = 105 PM$. If scale factors with Very Low rating are each assigned a weight of (5), then a project with Very Low (5) ratings for all factors will have

$SF_j = 5$, $B = 1.26$, and a relative effort $E = 331 PM$. This represents a large variation, but the increase involved in a one-unit change in one of the factors is only about 4.7%.

Scale Factors (SF _j)	Very Low	Low	Nominal	High	Very High	Extra High
PREC	thoroughly unprecedented	largely unprecedented	Somewhat unprecedented	generally familiar	largely familiar	thoroughly familiar
FLEX	rigorous	occasional relaxation	some relaxation	general conformity	some conformity	general goals
RESL ¹	little (20%)	some (40%)	often (60%)	Generally (75%)	mostly (90%)	full (100%)
TEAM	very difficult interactions	some difficult interactions	Basically cooperative interactions	largely cooperative	highly cooperative	seamless interactions
PMAT	Weighted average of “Yes” answers to CMM Maturity Questionnaire					

Table 1.1: Scale Factors for COCOMO.II Early Design and Post-Architecture Models

1.1.4 Sizing Methods

SLOC: Lines of Code Counting Rules

In COCOMOII, the logical source statement has been chosen as the standard line of code. Defining a line of code is difficult due to conceptual differences involved in accounting for executable statements and data declarations in different languages. The goal is to measure the amount of intellectual work put into program development, but difficulties arise when trying to define consistent measures across different languages. Breakage due to change of requirements also complicates sizing. To minimize these problems, the Software Engineering Institute (SEI) definition checklist for a logical source statement is used in defining the line of code measure. The Software Engineering Institute (SEI) has developed this checklist as part of a system of definition checklists, report forms and supplemental forms to support measurement definitions [Park 1992] [Goethert et al. 1992].

Figure 1-1 shows a portion of the definition checklist as it is being applied to support the development of the COCOMOII model. Each checkmark in the “Includes” column identifies a particular statement type or attribute included in the definition, and vice-versa for the excludes. Other sections in the definition clarify statement attributes for usage, delivery, functionality, replications and development status. There are also clarifications for language specific statements for ADA, C, C++, CMS-2, COBOL, FORTRAN, JOVIAL and Pascal.

¹. % significant module interfaces specified,% significant risks eliminated.

Some changes were made to the line-of-code definition that departs from the default definition provided in [Park 1992]. These changes eliminate categories of software, which are generally small sources of project effort. Not included in the definition are commercial-off-the-shelf software (COTS), government-furnished software (GFS), other products, language support libraries and operating systems, or other commercial libraries. Code generated with source code generators is not included though measurements will be taken with and without generated code to support analysis.

The “COCOMOII line-of-code definition” can be calculated in several ways. One way is to use the software program, Amadeus[Amadeus 1994] [Selby et al. 1991]. Another software program is code count.

Figure 1-1 Definition Checklist for Source Statements Counts

Measurement unit		Physical source lines			
Logical source statements					
Statement type	Definition <input type="checkbox"/>	Data Array		Includes	Excludes
<i>When a line or statement contains more than one type, classify it as the type with the highest precedence.</i>					
1 Executable		Order of precedence →	1	<input type="checkbox"/>	
2 Nonexecutable					
3 Declarations			2	<input type="checkbox"/>	
4 Compiler directives			3	<input type="checkbox"/>	
5 Comments					
6 On their own lines			4		<input type="checkbox"/>
7 On lines with source code			5		<input type="checkbox"/>
8 Banners and non-blank spacers			6		<input type="checkbox"/>
9 Blank (empty) comments			7		<input type="checkbox"/>
10 Blank lines			8		<input type="checkbox"/>
11					
12					
How produced	Definition <input type="checkbox"/>	Data array		Includes	Excludes
1 Programmed				<input type="checkbox"/>	
2 Generated with source code generators					<input type="checkbox"/>
3 Converted with automated translators				<input type="checkbox"/>	
4 Copied or reused without change				<input type="checkbox"/>	
5 Modified				<input type="checkbox"/>	
6 Removed					<input type="checkbox"/>
7					
8					
Origin	Definition <input type="checkbox"/>	Data array		Includes	Excludes
1 New work: no prior existence				<input type="checkbox"/>	
2 Prior work: taken or adapted from					
3 A previous version, build, or release				<input type="checkbox"/>	
4 Commercial, off-the-shelf software (COTS), other than libraries					<input type="checkbox"/>
5 Government furnished software (GFS), other than reuse libraries					<input type="checkbox"/>
6 Another product					<input type="checkbox"/>
7 A vendor-supplied language support library (unmodified)					<input type="checkbox"/>
8 A vendor-supplied operating system or utility (unmodified)					<input type="checkbox"/>
9 A local or modified language support library or operating system					<input type="checkbox"/>
10 Other commercial library					<input type="checkbox"/>
11 A reuse library (software designed for reuse)				<input type="checkbox"/>	
12 Other software component or library				<input type="checkbox"/>	
13					
14					

1.1.5 FP: Counting with Unadjusted Function Points

The function point cost estimation approach is based on the amount of functionality in a software project and a set of individual project factors [Behrens 1983][Kunkler 1985][IFPUG 1994]. Function points are useful estimators since they are based on information that is available early in the project life cycle. A brief summary of function points and their calculation in COCOMOII is as follows.

Function points measure a software project by quantifying the information processing functionality associated with major external data input, output, or file types. Five user function types should be identified as defined in the Table2.

External Input (Inputs)	Count each unique user data or user control input type that (i) enters the external boundary of the software system being measured and (ii) adds or changes data in a logical internal file.
External Output (Outputs)	Count each unique user data or control output type that leaves the external boundary of the software system being measured.
Internal Logical File (Files)	Count each major logical group of user data or control information in the software system as a logical internal file type. Include each logical file (e.g., each logical group of data) that is generated, used, or maintained by the software system.
External Interface Files (Interfaces)	Files passed or shared between software systems should be counted as external interface file types within each system.
External Inquiry (Queries)	Count each unique input-output combination, where an input causes and generates an immediate output, as an external inquiry type.

Table 2: User Function Types

Each instance of these function types is then classified by complexity level. The complexity levels determine a set of weights, which are applied to their corresponding function counts to determine the Unadjusted Function Points quantity. This is the Function Point sizing metric used by COCOMOII. The usual Function Point procedure involves assessing the degree of influence (DI) of fourteen application characteristics on the software project determined according to a rating scale of 0.0 to 0.05 for each characteristic. The 14 ratings are added together, and added to a base level of 0.65 to produce a general characteristics adjustment factor that ranges from 0.65 to 1.35.

Each of these fourteen characteristics, such as distributed functions, performance, and reusability, thus have a maximum of 5% contribution to estimated effort. This is inconsistent with COCOMO experience; thus COCOMO.II uses Unadjusted Function Points for sizing, and applies its reuse factors, cost driver effort multipliers, and exponent scale factors to this sizing quantity.

1.1.6 AAF: Adaptation Adjustment Factors

Adaptation of Existing Code

COCOMO is not only capable of estimating the cost and schedule for a development started from "scratch", but it is also able to estimate the cost and schedule for products that are built upon already existing code. Adaptation considerations have also been incorporated into COCOMO, where an estimate for KSLOC will be calculated. This value will be substituted in place of the SLOC found in the equations already discussed. This adaptation of code utilizes an additional set of equations that are used to calculate the final count on source instructions and related cost and schedule. These equations use the following values as components:

- Adapted Source Lines of Code (ASLOC). The number of source lines of code adapted from existing software used in developing the new product.
- Percent of Design Modification (DM). The percentage of the adapted software's design that received modification to fulfill the objectives and environment of the new product.
- Percent of Code Modification (CM). The percentage of the adapted software's code that receives modification to fulfill the objectives and environment of the new product.
- Percent of Integration Required for Modified Software (IM). The percentage of effort needed for integrating and testing of the adapted software in order to combine it into the new product.
- Percentage of reuse effort due to Software Understanding (SU).
- Percentage of reuse effort due to Assessment and Assimilation (AA).
- Programmer Unfamiliarity with Software (UNFM)

These components are brought together in Figure 1-6. The AAF is the adaptation adjustment factor. The AAF is the calculated degree to which the adapted software will affect overall development.

1.1.7 Effort Multipliers

There are a number of contributing factors to a project's delivery time and effort. Development productivity was found to be affected by additional factors that were found to fall under the headings: product attributes, platform attributes, personnel attributes, and project attributes.

Product attributes refer to the constraints and requirements placed upon the project to be developed. These included

- Required software reliability (RELY)
- Database size (DATA)
- Documentation match to life-cycle needs (DOCU)
- Product complexity (CPLX)
- Required Reusability (RUSE)

Platform attributes refer to the limitations placed upon development effort by the hardware and operating system being used to run the project. These limitations are listed below.

- Execution time constraint (TIME)
- Main storage constraint (STOR)
- Platform volatility (PVOL)

Personnel attributes refer to the level of skills that are possessed by the personnel. The skills in question are general professional ability, programming ability, experience with the development environment and familiarity with the project's domain. These skills are characterized below.

- Analyst capabilities (ACAP)
- Applications experience (AEXP)
- Programmer capabilities (PCAP)
- Platform experience (PEXP)
- Programming language experience (LEXP)
- Personnel Continuity (PCON)

Project attributes refer to the constraints and conditions under which project development takes place. The issues that affect development are:

- Use of software tools (TOOL)
- Multisite Development (SITE)

These 16 factors are incorporated into calculating an estimated effort and schedule. Each of the factors has associated with it up to six ratings. These ratings are *very low*, *low*, *nominal*, *high*, *very high*, and *extra high*. Each rating has a corresponding real number based upon the factor and the degree to which the factor can influence productivity. A rating less than 1 denotes a factor that can decrease the schedule and effort. A rating greater than 1 denotes a factor that extends the schedule or effort. Lastly, a rating equal to 1 does not extend nor decrease the schedule and effort (this rating is called *nominal*).

These 16 factors (or effort multipliers) are incorporated into the schedule and effort estimation formulas by multiplying them together (see Figure 1-7 for the COCOMO dialog box). The numerical value of the i^{th} adjustment factor (there are 16 of them) is called EM_i and their product is called the adjustment factor or EAF. The actual effort, PM_{total} is the product of the nominal effort times the EAF.

In addition to the 16 EAF factors there are two user defined factors named $USR1$ and $USR2$. Their initial values are all set to 1. They may be redefined by using the Parameters-User Defined EAF menu item.

A final effort multiplier, Required Development Schedule (SCED) is treated separately as it operates at the overall project level rather than potentially varying from module to module.

FIGURE 1-2 **Estimate Development Effort**

$$PM_{\text{total}} = (SCED) \times PM_{\text{nominal}} \times \prod_{i=1}^{18} EM_i$$

1.2 Navigating COCOMO

This software is a stand-alone software system intended for a single user. The software is user interactive in that it attempts to interface well with a user's needs, using extensive mouse interaction wherever possible.

On the screen in Figure 1-3 is the CLEF (Component Level Estimation Form). This is where all of the entered information will be displayed. The top of the screen shows all of the subfunctions which the user may call. The choices appear in pop down menus according to the major headings of Project, Model, and Phase.

In order to efficiently use COCOMO, you must become familiar with the Component Level Estimating Form (CLEF). The different sections that are to be discussed have been given a corresponding number. These sections are given a descriptive label as a point of reference as well as a summary of their contents and functions.

The sections found in Figure 1-3 and their descriptions are as follows:

FIGURE 1-3 COCOMO CLEF

The screenshot shows the COCOMO CLEF software interface. The title bar reads "USC-COCOMO II, 2000, 3 - Untitled". The menu bar includes File, Edit, View, Parameters, Calibrate, Phase, Maintenance, and Help. The toolbar contains icons for file operations and help. The main form area includes:

- Project Name:** A text field containing "<sample>" (callout 3).
- Scale Factor:** A text field containing "18.97" (callout 13).
- Schedule:** A button (callout 14).
- Project Notes:** A button (callout 4).
- Development Model:** A dropdown menu currently showing "Post Architecture" (callout 15).
- Table Headers:** A table with 13 columns: X, Module Name, Module Size, LABOR Rate (\$/month), ERF, Language, NOM Effort DEV, EST Effort DEV, PROD, COST, INST COST, Staff, and RISK (callout 16).
- Summary Section:**
 - Total Lines of Code:** A text field containing "0" (callout 11).
 - Hours/PM:** A text field containing "152.00" (callout 12).
 - Estimated Effort Sched PROD COST INST Staff RISK:** A table with three rows: Optimistic, Most Likely, and Pessimistic. All values are 0.0 except for RISK in the Most Likely row, which is 0.0 (callout 17).

Callouts 6, 7, 8, 9, and 10 point to the table columns. Callout 22 points to the status bar which displays "Ready".

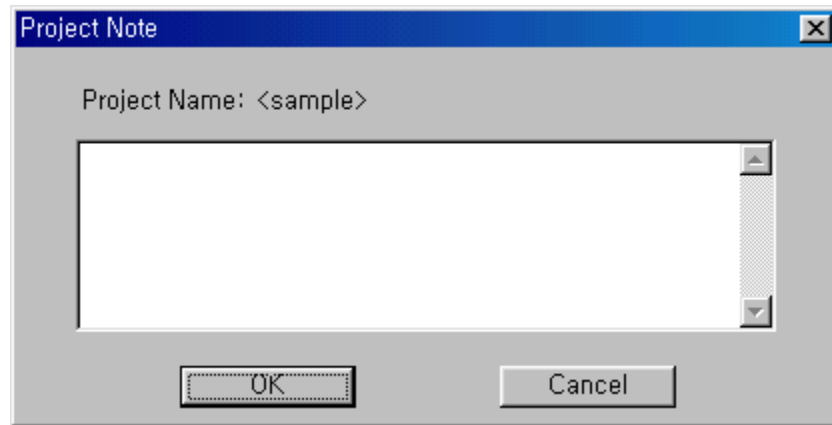
1. **Main Menu bar** - This area contains the menu selection of the main functions of COCOMO. These selections are *File*, *View*, *Edit*, *Parameters*, *Calibrate*, *Phase Distribution* and *Help*. *File*, *View*, *Edit*, *Parameters*, *Calibrate*, and *Phase Distribution* are discussed in chapters 2, 3, 4, 5, and 6 respectively. *Help* is the selection used to receive on-line assistance with the available functions.

2. **Tool bar** - This area contains image buttons like other windows applications for New Project, Open Project, Save Project, Delete Module, Copy & Paste, Insert clipboard content, Insert a module, and About functions.

3. **Project Name** - This editable field displays the name of the currently displayed project. To edit the name click twice upon this field and proceed to edit name. Upon completion of editing press the "Return" key. The Default name of a new project is "<sample>".

4. **Project Note** - This button displays the Project Note Dialog Box as shown in Figure 1-4. In this box, user can write brief description of the project.

FIGURE 1-4 **Project Note**



5. **X** - This column is reserved for identifying a module. Pressing upon this field for a given module will mark the desired module. Marking is denoted by an x that appears in this column. Only one module can be marked at a time. Modules are marked in order to perform module deletion, cutting, copying or pasting.

6. **Module Name Column** - This column is used to house the name of each module located in the Module Area. The module name can be changed by clicking twice on the desired module name box and entering the changes into the module name field. Upon completion of editing press "Return".

7. **Module Size (SLOC) Column** - This column is used to house the SLOC of each module located in the Module Area. The value for SLOC can be computed in one of three ways. One, the value can be entered directly in the SLOC field as shown in Figure 1-5. Two, by using the function point model as shown in Figure 1-6. Three, by using Adaptation Adjustment Factor as shown in Figure 1-7. Upon completion click on OK. There is a limit to the range of input. The inputted value for SLOC must be within the range 0 - 9,999,999.

Note - COCOMO is not calibrated for Total SLOC < 2000.

FIGURE 1-5 SLOC Dialog Box - Source Lines of Code (SLOC)

SLOC Input Dialog - <sample>

Sizing Method

- ☒ SLOC
- ☐ Function Points
- ☐ Adaptation and Reuse

Breakage
% of code thrown away due to requirements evolution and volatility

REVL

Module Size in SLOC

Language

SLOC

OK Cancel Help

FIGURE 1-6 SLOC Dialog Box - Function Points (FP)

SLOC Input Dialog - <sample>

Sizing Method

☐ SLOC
☒ **Function Points**
☐ Adaptation and Reuse

Breakage
 % of code thrown away due to requirements evolution and volatility
 REVL

Module Size in Function Points

Language

Ratio Type : ☒ Jones ☐ David

Calculation Method : ☒ Using Table ☐ Input Calculated Function Point

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Inputs	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	0
Outputs	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	0
Files	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	0
Interfaces	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	0
Queries	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	0
Total Unadjusted Function Points				0
Equivalent Total in SLOC				0

There are two Ratio types, Jones and David. Jones is a set of the function point multipliers provided by Caper Jones. David is a set of the function point multipliers provided by David Consulting Groups. Select one of two types to estimate the SLOC using function point.

Also there are two methods to estimate SLOC using function point, 'Using Table' and 'Input Calculated Function Point'.

FIGURE 1-7 SLOC Dialog Box - Adaptation Adjustment Factor (AAF)

SLOC Input Dialog - <sample>

Sizing Method

- ☐ SLOC
- ☐ Function Points
- ☒ Adaptation

Breakage
% of code thrown away due to requirements volatility
BRAK

Adaptation

Initial SLOC	<input type="text" value="0"/>
% Design Modified	<input type="text" value="100"/> %
% Code Modified	<input type="text" value="100"/> %
% Integration Modified	<input type="text" value="100"/> %
Software Understanding (SU)	<input type="text" value="30"/> SU
Assesment & Assimilation (AA)	<input type="text" value="4"/> AA
Unfamiliarity with Software	<input type="text" value="0.4"/> UNFM
% Components Automatically Translated	<input type="text" value="0"/> %
Automatic Translation Productivity	<input type="text" value="2400"/>
Computed Adaptation Adjustment Factor	100
Computed ASLOC	0

OK Cancel Help

7. Labor Rate Column - This column contains the amount of money at which a developer working on a particular module would be paid per month. The labor rate can be edited by clicking on the corresponding Labor Rate box and entering the new value via the edit area. The range on labor rate is between \$0 and \$99,999.

8. Effort Adjustment Factor (EAF) Column - This column displays the product of the cost drivers for each specific module. By clicking on this field a dialog box appears (see Figure 1-7). This box displays all of the cost drivers, inter cost drivers and their current ratings. The cost drivers are divided into the groupings: *Product*, *Platform*, *Personnel* and *Project*. The inter cost drivers are rated as 0%, 25%, 50%, and 75 %. The ratings for each multiplier can be changed by cycling through the available ratings until the desired rating is displayed. As the cost driver ratings are changed the total product of the cost drivers is displayed in the upper right hand corner of the dialog box along with the module name. The final rating of a cost driver is calculated using this formula for the interpolation.

Final rating = (Next cost driver rating - Current cost driver rating) * Current inter cost driver / 100

9. Totals Area - This area houses the calculated results of all of the modules combined. Within this area is the total SLOC count for the module, the total nominal effort (PM), the total nominal productivity (SLOC/PM), the total estimated effort (EST PM), the total estimated productivity (Prod), the total estimated project cost, the estimated cost per instruction, the total estimated FSWP and the total estimated schedule for project completion (see each individual column for more information). The latter six quantities have not only a most likely estimate but also an optimistic estimate (no less than this, 90% of the time) and a pessimistic estimate (no greater than this, 90% of the time).

10. Status bar - This window displays a short definition of the column headings clicked upon and also displays a short description of the result of the last function initiated by the user.

11. Schedule Button - This button displays the Schedule Dialog Box as shown in Figure 1-8.

12. Scale Factor Button - This button displays the Scale Factor Dialog Box as shown in Figure 1-9.

13. Risk Column - This column contains the Total risk level for each specific module. By clicking on this field a dialog box appears (see Figure 1-10). This box displays all of the risk levels for the chosen module. The total risk of a module is computed as:

$\text{total_risk} = \text{schedule_risk} + \text{product_risk} + \text{personnel_risk} + \text{process_risk} + \text{platform_risk} + \text{reuse_risk};$

$\text{total risk of a module} = \text{total_risk} / 373 * 100;$

For the definitions of schedule risk, product risk, platform risk, personnel risk, process risk, and reuse risk, see [Madachy 1997].

14. Full-time SoftWare Personnel (FSWP) Column - This column houses the calculated most likely estimate for the number of full-time developers that would be needed to complete a module in the estimated development time.

15. Instruction Cost Column - This column contains the calculated most likely cost per instruction. This number is calculated from Cost/SLOC in each module.

FIGURE 1-7 EAF Dialog Box

EAF - <sample>

base + Incr % = rating

Product:	RELY	DATA	DOCU	CPLX	RUSE
base	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%

Platform:	TIME	STOR	PVOL
base	NOM	NOM	NOM
Incr%	0%	0%	0%

Personnel:	ACAP	AEXP	PCAP	PEXP	LTEX	PCON
base	NOM	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%

Project:	TOOL	SITE
base	NOM	NOM
Incr%	0%	0%

User:	USR1	USR2
base	NOM	NOM
Incr%	0%	0%

EAF is also affected by Schedule

EAF: 1.00

OK Cancel Help

FIGURE 1-8 Schedule Dialog Box

Schedule

Schedule..... NOM 1.00

0%

OK Cancel Help

16. **Cost Column** - This column contains the calculated most likely estimate of the development cost for a particular module.

17. **Productivity (PROD) Column** - This column contains the calculated result of the module's individual SLOC divided by the module's most likely effort estimate.

18. **Estimated Person-Month (EST PM) Column** - This column holds the module's most likely effort estimate obtained from multiplying Effort Adjustment Factor (EAF) by Nominal Person Month (NOM PM).

19. **Nominal Person-Month (NOM PM) Column** - This column holds the module's most likely effort estimate without incorporating the Effort Adjustment Factors (EAF).

FIGURE 1-9 **Scale Factor Dialog Box**

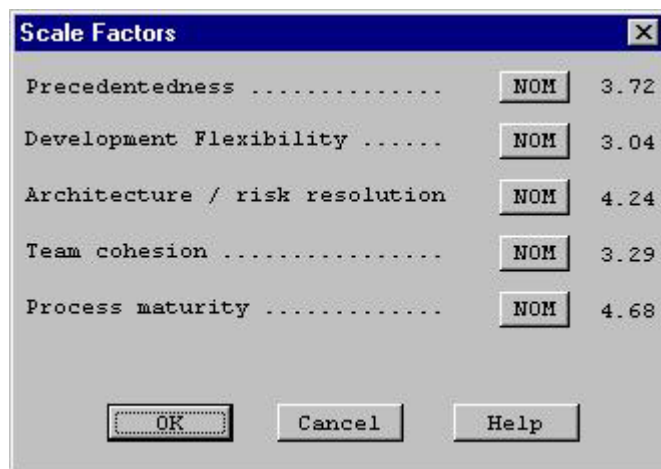
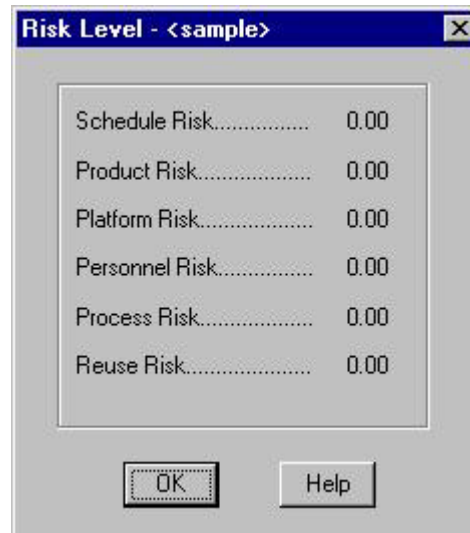


FIGURE 1-10 Risk Level Dialog Box


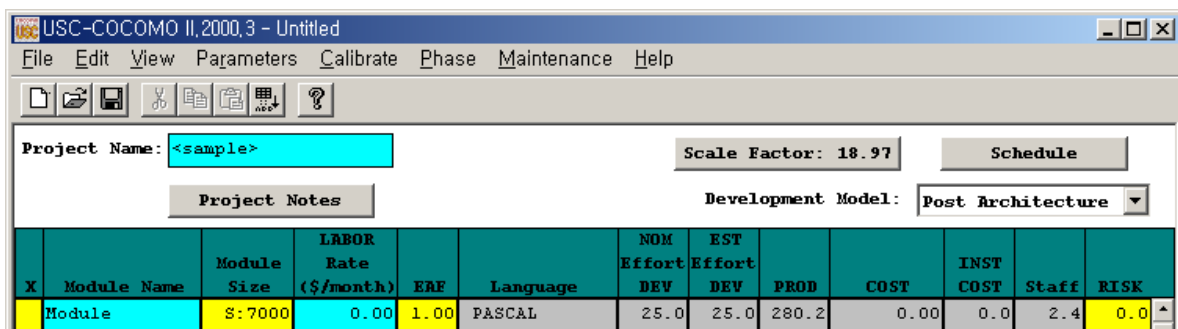
The dialog box is titled "Risk Level - <sample>". It contains a list of risk categories, each followed by a value of 0.00:

- Schedule Risk..... 0.00
- Product Risk..... 0.00
- Platform Risk..... 0.00
- Personnel Risk..... 0.00
- Process Risk..... 0.00
- Reuse Risk..... 0.00

At the bottom of the dialog box are two buttons: "OK" and "Help".

1.3 Begin Using COCOMO

To begin entering a new module, either click on the "Add Module" button on the Tool bar or on the pulldown menu item (Edit|Add Module). At this point, a new module will appear in the CLEF with all values set to their respective defaults. Double click upon the module name field in order to give the new module a name. Upon typing the module name press "Return.". A value for SLOC and Labor rate may also be given by clicking on the respective field and editing appropriately (see Figure 1-11).

FIGURE 1-11 Create Sample Module and give values to SLOC and Labor Rate


The screenshot shows the USC-COCOMO II, 2000, 3 - Untitled application window. The menu bar includes File, Edit, View, Parameters, Calibrate, Phase, Maintenance, and Help. The toolbar contains icons for file operations and help. The main window displays the following information:

- Project Name:** <sample>
- Scale Factor:** 18.97
- Schedule:** (button)
- Project Notes:** (button)
- Development Model:** Post Architecture (dropdown menu)

Below this information is a table with the following data:

X	Module Name	Module Size	LABOR Rate (\$/month)	ERF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Module	S:7000	0.00	1.00	PASCAL	25.0	25.0	280.2	0.00	0.0	2.4	0.0

NOTE - In order to change any of the editable fields, just click on the desired field twice and begin editing the field. Upon completing editing, either hit the "Return" key, or click on OK. All of the final results can be found at the bottom of the CLEF in the Totals area (see Figure 1-12).

FIGURE 1-12 Totals area after calculations have been completed

Total Lines of Code:		7000	Estimated		Effort	Sched	PROD	COST	INST	Staff	RISK
Hours/PM:		152.00	Optimistic		20.0	9.5	350.2	0.00	0.0	2.1	
			Most Likely		25.0	10.2	280.2	0.00	0.0	2.4	0.0
			Pessimistic		31.2	11.0	224.1	0.00	0.0	2.8	

Ready

1.4 Running COCOMO: Windows, Sparc or Java

Currently, there are three implementations of COCOMOII: a Windows 95/NT version, a Sun Microsystems Sparcstation version, and a Java version.

To download any these versions, you should enter this in a web browser:

<http://sunset.usc.edu/COCOMOII/cocomo.html>

and scroll down to the section labeled **COCOMOII Downloads(Software and Documentation)**

or

ftp://ftp.usc.edu/pub/soft_engineering/COCOMOII/ where you will see 7 files:

c98sunos.tar.gz, c98windows.zip, c98java.tar.gz, usersman.ps, usersman.pdf, modelman.ps, modelman.pdf

To run java cocomo(Netscape Navigator3.0 or Microsoft Internet Explorer3.0 or higher), you should visit:

http://sunset.usc.edu/j_cocomo/cocomo.html

Chapter 2: File Menu

The COCOMO file types include: project file, model file, report file, calibration file, and csv file. The first three are discussed here. The others are discussed in later chapters.

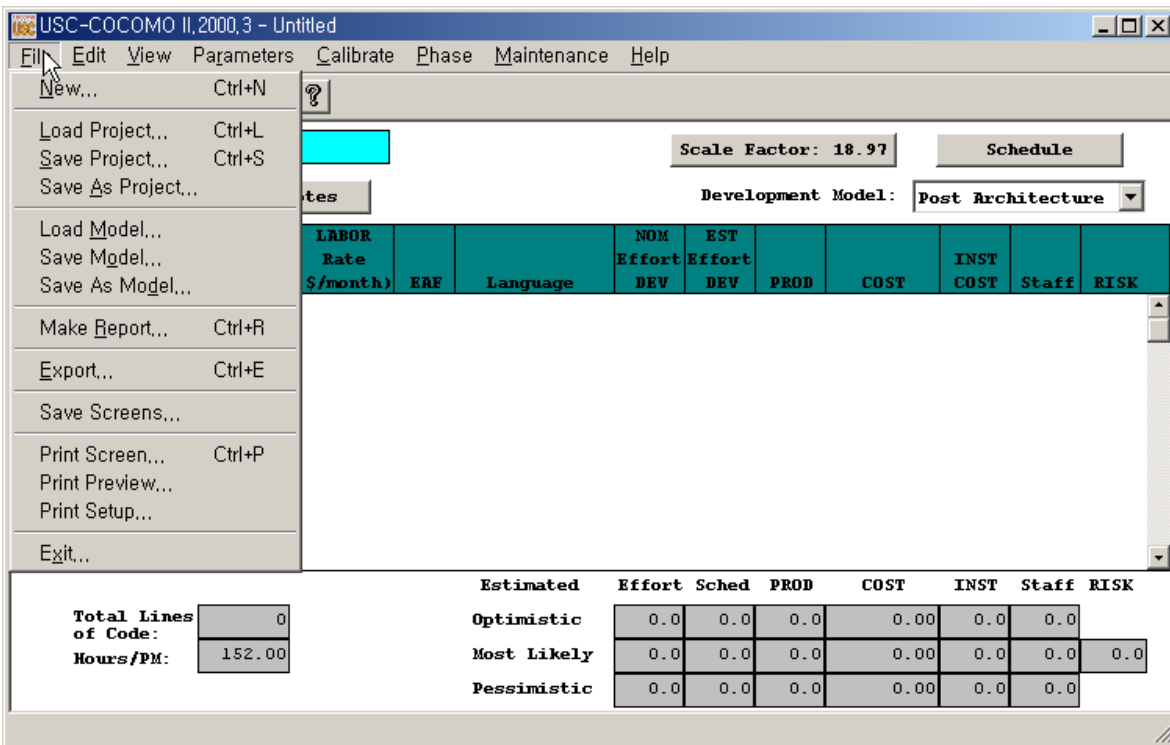
The project file in COCOMO stores a project's data, which include project name, project scale factors, project schedule constraint, module name, SLOC, labor rate, effort adjustment factors (EAF), and COCOMO related calculation results. The COCOMO system gives all project files an ".est" extension.

Regarding the model file, as we mentioned in chapter one, COCOMO incorporates 17 predictor factors, or cost driver attributes, which are grouped into four categories: software product attributes, platform attributes, personnel attributes, and project attributes. Each of these cost drivers attributes determines a multiplying factor, which estimates the effect of the attribute in software development effort. There are also two user defined EAF factors plus the project-level required development schedule EAF factor. Besides these cost drivers, COCOMO also has scale factors. These multiplying factors and effort estimating equations constitute the model of a project. As we said previously, COCOMO has assigned default values and equations for the annually calibrated default model. Each time a COCOMO project is created, its effort estimate is based on the default model. COCOMO provides flexibility in changing the values of multiplying factors and effort estimating equation, or schedule estimating equation. An adjusted model can no longer be considered default values, and therefore it will be lost if COCOMO is exited without saving it in a model file. Upon saving this model file, these altered values can be applied to another project by loading the saved model file. The COCOMO system gives all model files a ".mod" extension.

The report file is a summary report of the COCOMO project. This report contains all entered and calculated values of a project. These files are given a ".rpt" extension.

The File menu option will enable you to create, retrieve, save, or print COCOMO files.

To select the File menu and its options, click on File with the mouse. The File menu will appear as Figure 2-1.

FIGURE 2-1 File Menu

2.1 New

The New option creates a new project file in the COCOMO working window, replacing any previous project file in the working window.

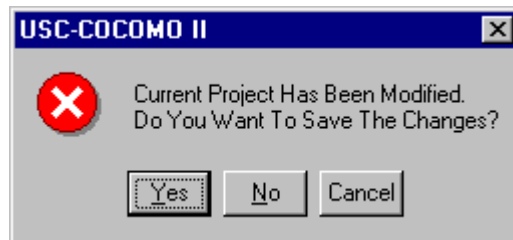
To Create a New Working File

1. Choose New from the File menu with mouse.

The working window will now be clear; the previous project file in the working window has been removed.

Note: New can be selected anytime; however, if the previous project file or model file has been modified, a warning dialog box will appear and requests confirmation. (as seen in Figure 2-2)

FIGURE 2-2 **Warning Dialog Box**



2. If the modifications on the previous file are not to be saved, choose Yes, otherwise choose No. If the No is selected, a Save File dialog will appear. (See Save Project and Save Model respectively)

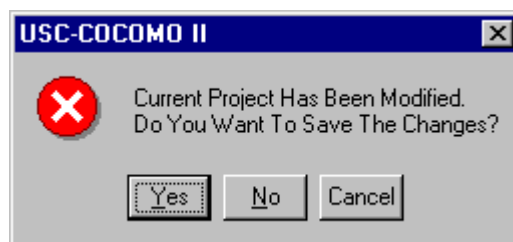
2.2 *Load Project*

The Load Project option is used to retrieve a project file as well as loading it on the working window.

To Retrieve or Load a Project File

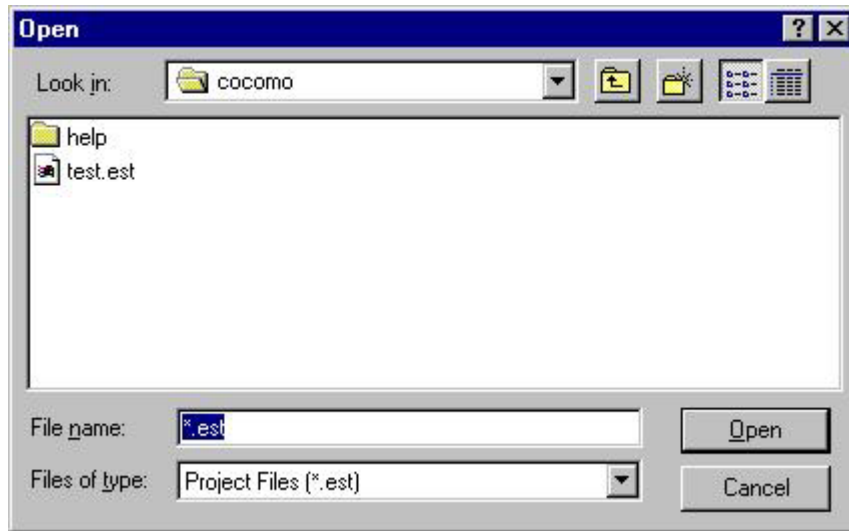
1. Choose Load Project from the File menu with the mouse.
2. If a previous project file has been modified in the working window, the dialog box as in Figure 2-3 will appear.

FIGURE 2-3 **Warning Dialog Box**



3. If the previous project file is to be saved, choose Yes, then a Save File dialog box will appear. (See Save Project). If the modified file is not to be saved, choose No.

4. The Load Project dialog box will appear as seen in Figure 2-4.

FIGURE 2-4 Load Project Dialog Box

The file name of a COCOMO project has a default format with ".est" as an extension. With this window, the desired project file can be selected from the Files scroll list for loading. If the desired project file does not exist in the scroll list, it is necessary to choose an appropriate directory.

5. Choose desired directory for file loading
6. When the desired file is shown on the Files list, click it, and click the "OK" button to initiate project loading.
7. After a project file is loaded, its file name will be displayed on the PROJECT FILE field at upper left corner on the working window, and all modules and related items will be displayed in the CLEF area. If the number of modules is beyond the window scope, the scroll bar can be used to look at all items.

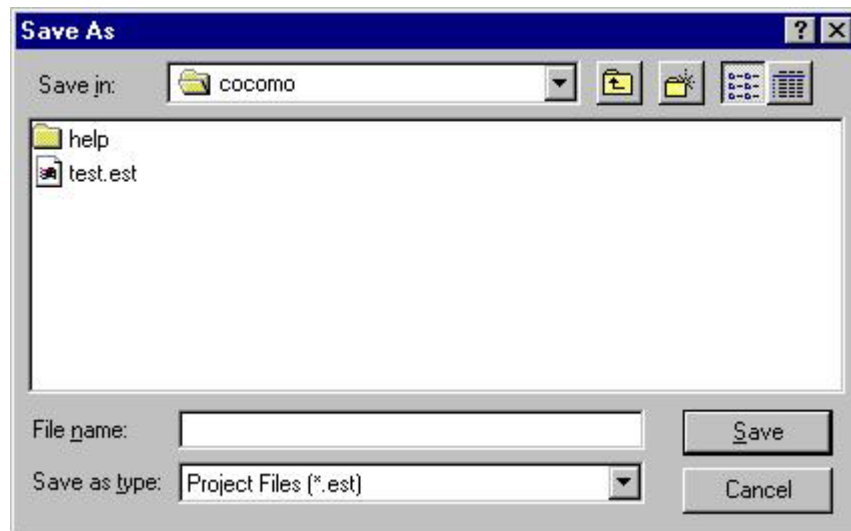
2.3 Save Project

The Save Project option is used to store the results of the current COCOMO project as a file with ".est" extension.

To Store the Results of Current Project

1. Choose Save Project from the File menu with the mouse. If the current project is loaded from a previously stored project file, the Save Project will overwrite the same project file with the current project.
2. If the current project is a new one, i.e., being created by the New command, the Project Save dialog box will appear, as seen in Figure 2-5.

FIGURE 2-5 Save Project Dialog Box



3. Look at the Files scroll window. If the file saving is to update (overwrite) a existing project file, the desired filename should be found in the Files scroll list. If the filename can not be found from current list, change the directory from the Directories scroll list until the desired filename is being shown. When the desired filename is on the list, click it.
4. If the file saving is to store a new project file, choose the desired directory, then type in a new filename.
5. After the desired filename is selected or inputted, click the OK button to initiate project saving.

2.4 Save As Project

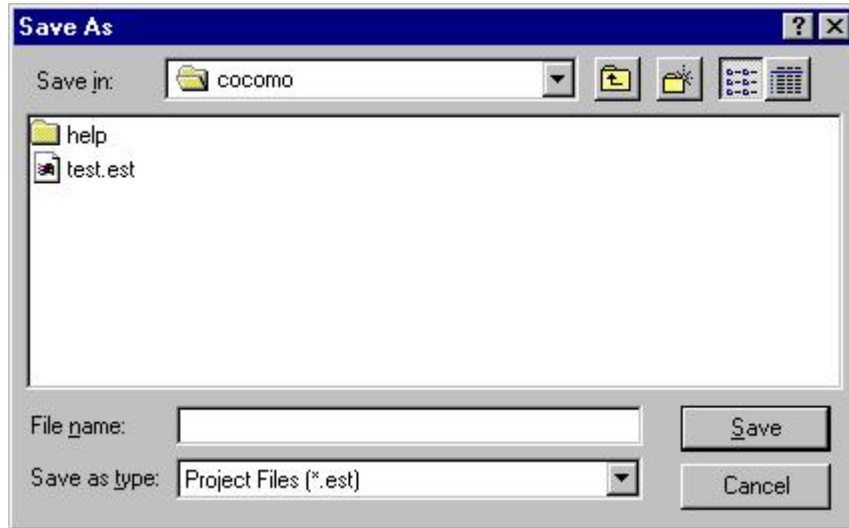
The Save As Project option is to store the current project as a COCOMO project file, which has a file name different from current file.

To Store Current Project With different File Name

1. Choose Save As Project from the File menu with the mouse.

2. The Save Project dialog box will appear, as seen in Figure 2-6.

FIGURE 2-6 Save Project Dialog Box



3. Look at the Files scroll window. If the file saving is to update (overwrite) a existing project file, the desired filename should be found in the Files scroll list. If the filename can not be found from current list, change the directory from the Directories scroll list until the desired filename is being shown. When the desired filename is on the list, click it.

4. If the file saving is to store a new project file, choose the desired directory, then type in a new filename in the SELECTION box.

5. After the desired filename is selected or inputted, click the OK button to initiate project saving. After a project file is saved, the project file name will be displayed on the PROJECT FILE field at the upper left corner of the working window.

2.5 Load Model

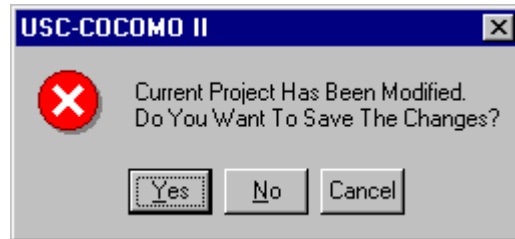
The Load Model command is used when a specific model, in which the values of multiplying factors and scale factors are different from the COCOMO default model, is to be applied to the current project.

The Load Model option is used to retrieve a model file as well as loading it for the current project.

To Retrieve or Load a Model File

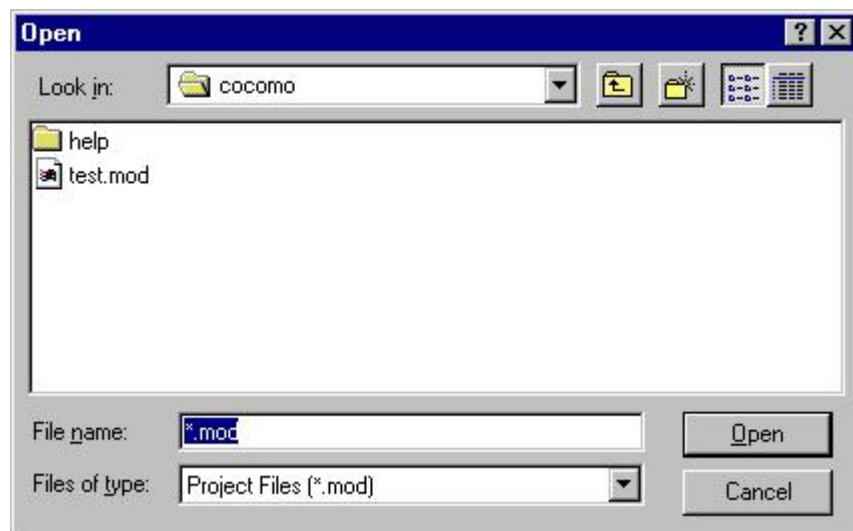
1. Choose Load Model from the File menu.
2. If a previous model has been modified in the current project, the following dialog box will appear.

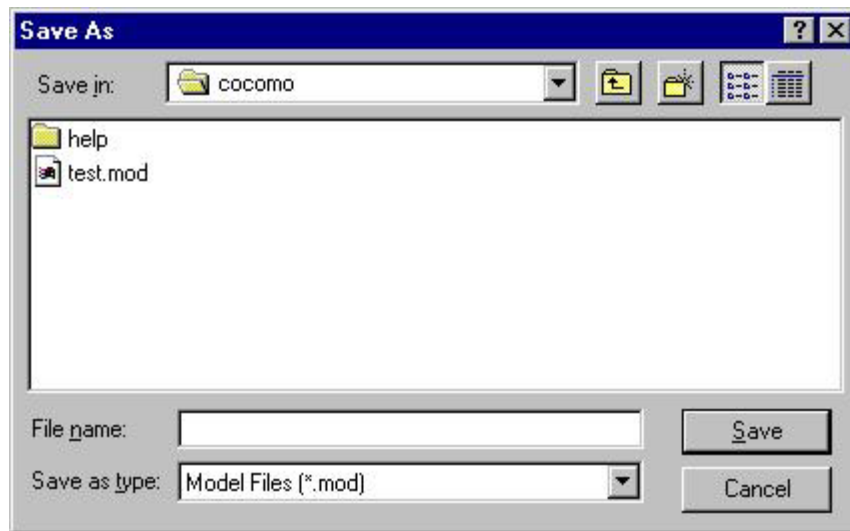
FIGURE 2-7 Warning Dialog Box



3. If the previous model file is to be saved, choose Yes, then a Save Model dialog box will appear. (See Save Model). If the modified model is not to be saved, choose No.
4. The Load Model dialog box will appear as seen in Figure 2-8.

FIGURE 2-8 Load Model Dialog Box





The file name of a COCOMO model has a default format with ".mod" as an extension. With this window, the desired model file can be selected from the Files scroll list for loading. If the desired model file does not exist in the scroll list, look for it in the other directories.

5. Choose desired directory for file loading
6. When the desired file is shown on the Files list, click it, and click the "OK" button to initiate model loading.
7. After a model file is loaded, its file name will be displayed on the MODEL FILE field at upper left corner on the working window, and the related costs of current project will be recalculated and shown on the working window.

2.6 Save Model

The Save Model option is used to store the results of the current COCOMO model as a file with ".mod" extension.

To Store the Results of Current Model

1. Choose Save Model from the File menu. If the current model is loaded from a previously stored model file, the Save Model will overwrite the same model file with the current model.
2. If the current model is a new one, the Save Model dialog box will appear, as seen in Figure 2-9.

FIGURE 2-9 Save Model Dialog Box

3. Look at the Files scroll window. If the file saving is to update (overwrite) a existing model file, the desired filename should be found in the Files scroll list. If the filename can not be found from current list, change the

directory from the Directories scroll list until the desired filename is being shown. When the desired filename is on the list, click it.

4. If the file saving is to store a new model file, choose the desired directory, then type in the filename.

5. After the desired filename is selected or inputted, click the OK button to initiate model saving.

2.7 *Save As Model*

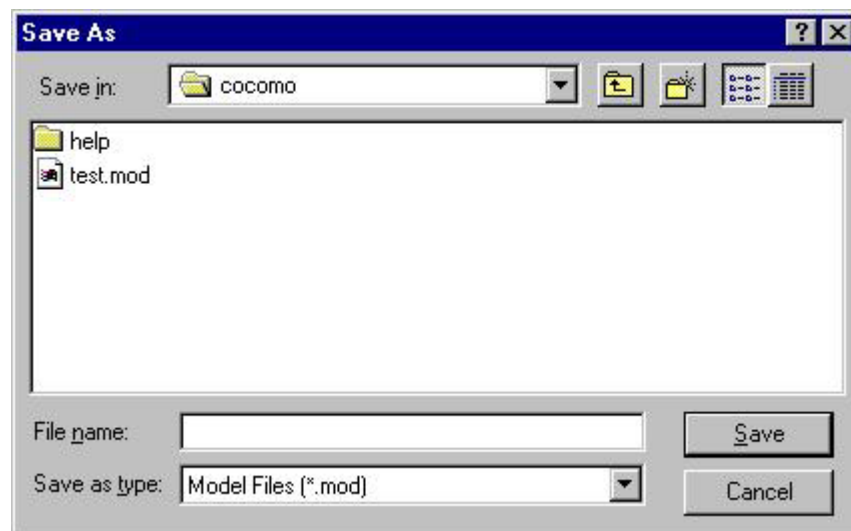
The Save As Model option is to store the current model as a COCOMO model file, which has a file name different from current model.

To Store Current Model With different File Name

1. Choose Save As Model from the File menu.

2. The Save Model dialog box will appear, as seen in Figure 2-10.

FIGURE 2-10 **Save As Model Dialog Box**



3. Look at the Files scroll window. If the file saving is to update (overwrite) a existing model file, the desired filename should be found in the Files scroll list. If the filename can not be found from current list, change the directory from the Directories scroll list until the desired filename is being shown. When the desired filename is on the list, click it.

4. If the file saving is to store a new model file, choose the desired directory, then type in the filename in the SELECTION box.

5. After the desired filename is selected or inputted, click the OK button to initiate model saving. After a model file is saved, the project file name will be displayed on the MODEL FILE field at the upper left corner of the working window.

2.8 Make Report

The Make Report option creates a COCOMO report in the form of a text file for printing.

To Create Project Report

1. Choose Make Report from the File menu.
2. The Make Report dialog box will appear, as seen in Figure 2-11.

FIGURE 2-11 Make Report Dialog Box



3. Look at the Files scroll window. If the file saving is to update (overwrite) a existing report file, the desired filename should be found in the Files scroll list. If the filename can not be found from current list, change the directory from the Directories scroll list until the desired filename is shown. When the desired filename is on the list, click it.

4. If the file saving is to store a new report file, choose the desired directory, then type in the filename.
5. Choose desired directory for file saving: Look at the filter input box. The path found in this box represents the directory where the report file is going to be saved. This path will be changed after each directory change. To change the directory, click the appropriate directory choice from the Directories scroll list, then click the "Filter" button.
6. After the desired filename is selected or inputted, click the OK button to initiate report file saving.
7. To print a COCOMO project report, execute the local commands for your system in order to send the file for printing.

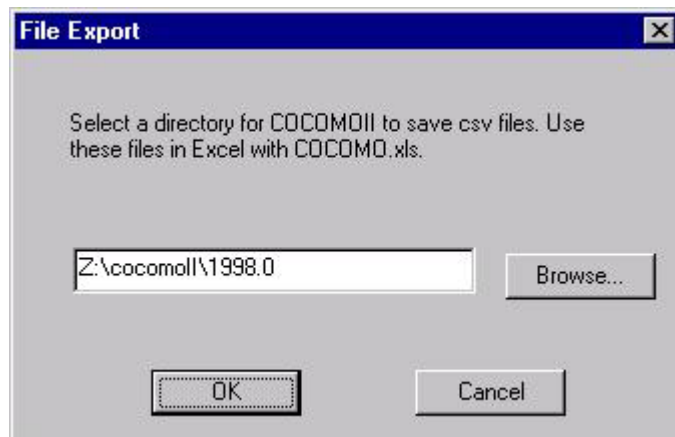
2.9 *Export*

The Export option lets you select a directory to write files that can be imported into Excel.

To Export

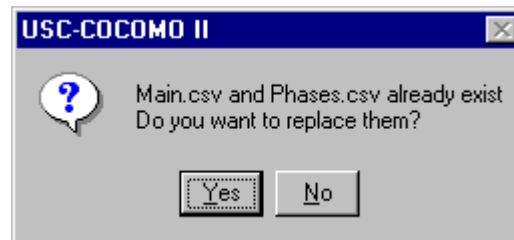
1. Choose Export from the File menu.
2. The Export dialog box will appear, as seen in Figure 2-12.

Figure 2-12 **File Export Dialog Box**



3. When you click on OK, a dialog box appears if Main.csv and Phases.csv already exist, as shown in Figure 2-13.
4. If Yes is selected, COCOMO saves two files(main.csv and phase.csv) in the chosen directory.
If No is selected, these files will not be replaced.

Figure 2-13 Subsequent File Export Dialog Box



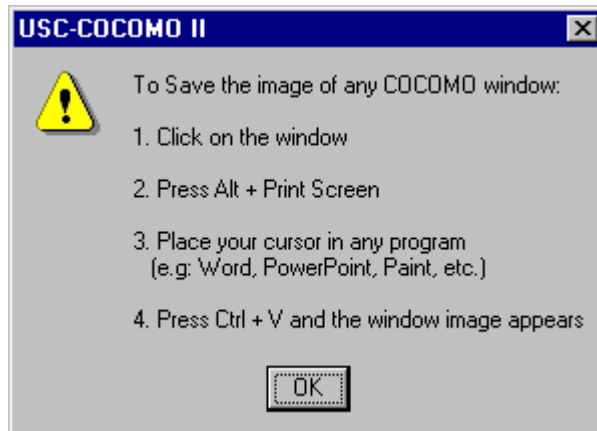
2.10 Save Screens

The Save Screens option allows the user to save the image of any Cocomo window.

To Save Screens

1. Choose Save Screens from the File menu.
2. The Save Screens dialog box will appear, as seen in Figure 2-14.
3. Follow the directions on the dialog box.

Figure 2-14 Save Screens Dialog Box



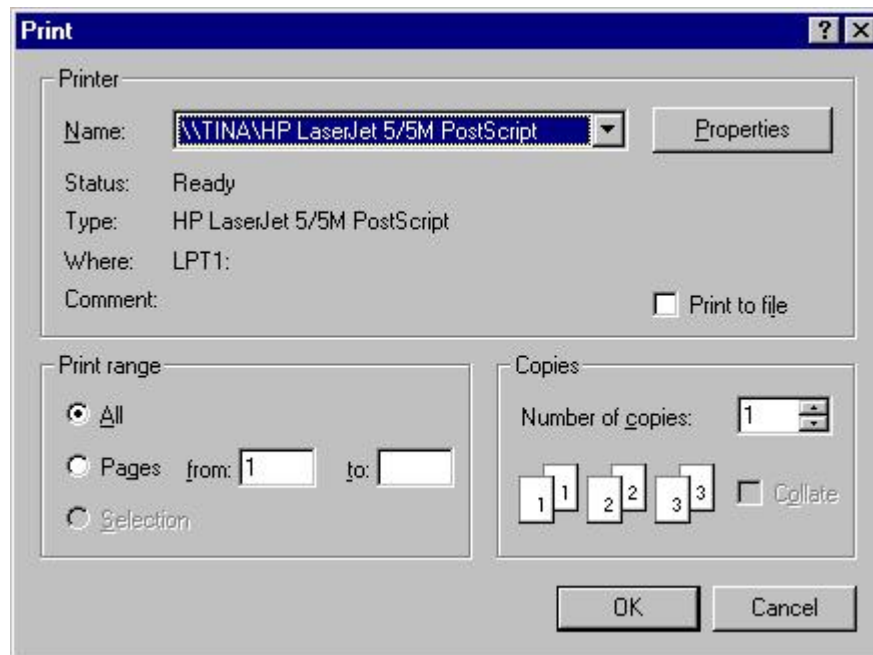
2.11 Print Screen

The Print Screen option prints the screen of the main Cocomo window.

To Print Screen

1. Choose Print Screen from the File menu.
2. The Print Screen dialog box will appear, as seen in Figure 2-15.

Figure 2-15 **Print Screen Dialog Box**



3. The Name of the printer can be selected from the dropdown list. Alternatively, you can print to a file by clicking on the Print to file checkbox.
4. Properties of the printer can be set by clicking on the Properties button.
5. The Print range can be All or Pages (e.g. from 1 to 3).
6. The Number of copies can be selected by clicking the up and down arrows, or by typing a number directly.
7. Select OK when finished to print or select Cancel to not print.

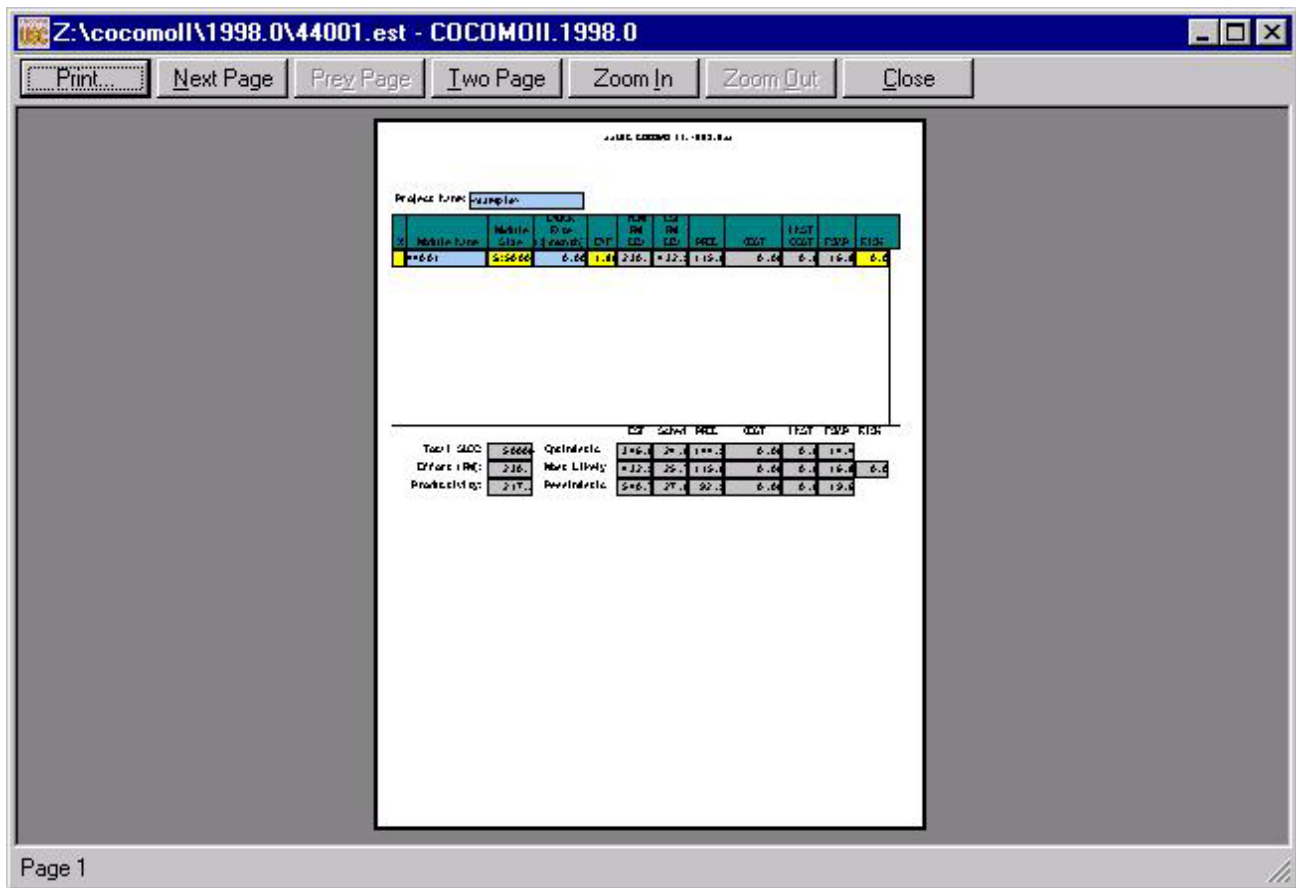
2.12 Print Preview

The Print Preview option displays that which will appear when printed, if Print Screen is selected from the File menu.

To Preview what is to be printed

1. Choose Print Preview from the File menu.
2. The Print Preview dialog box will appear, as seen in Figure 2-16.

Figure 2-16 Print Preview Dialog Box



3. Select the Print button to print.
4. The Next Page button to advance to the next Page.
5. The Zoom In button zooms in, and the Zoom Out button becomes enabled so that the user can zoom out.
6. The Close button closes the Print Preview dialog.

2.12 Print Setup

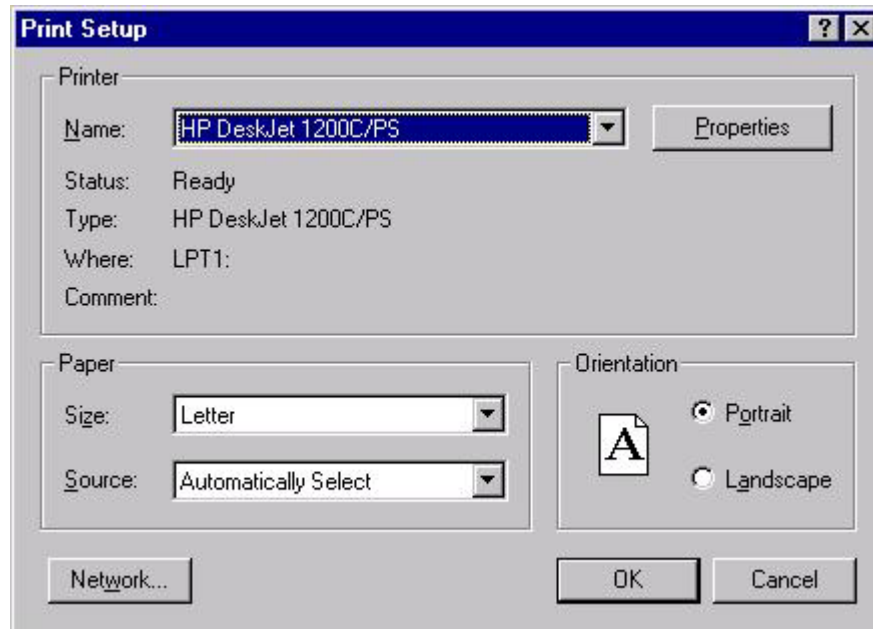
The Print Setup option allows the user to set up printing.

To set up Printing

1. Choose Print Setup from the File menu.

2. The Print Setup dialog box will appear, as seen in Figure 2-17.

Figure 2-17 Print Setup Dialog Box



3. The Name of the printer can be selected from the dropdown list.
4. Properties of the printer can be set by clicking on the Properties button.
5. The Size and Source of the paper can be selected from the dropdown lists.
6. The Network button can be selected to connect to a printer on a network.
7. Select OK when finished to print or select Cancel to not print.

2.13 Exit

The Exit option leaves the COCOMO system.

To Exit COCOMO

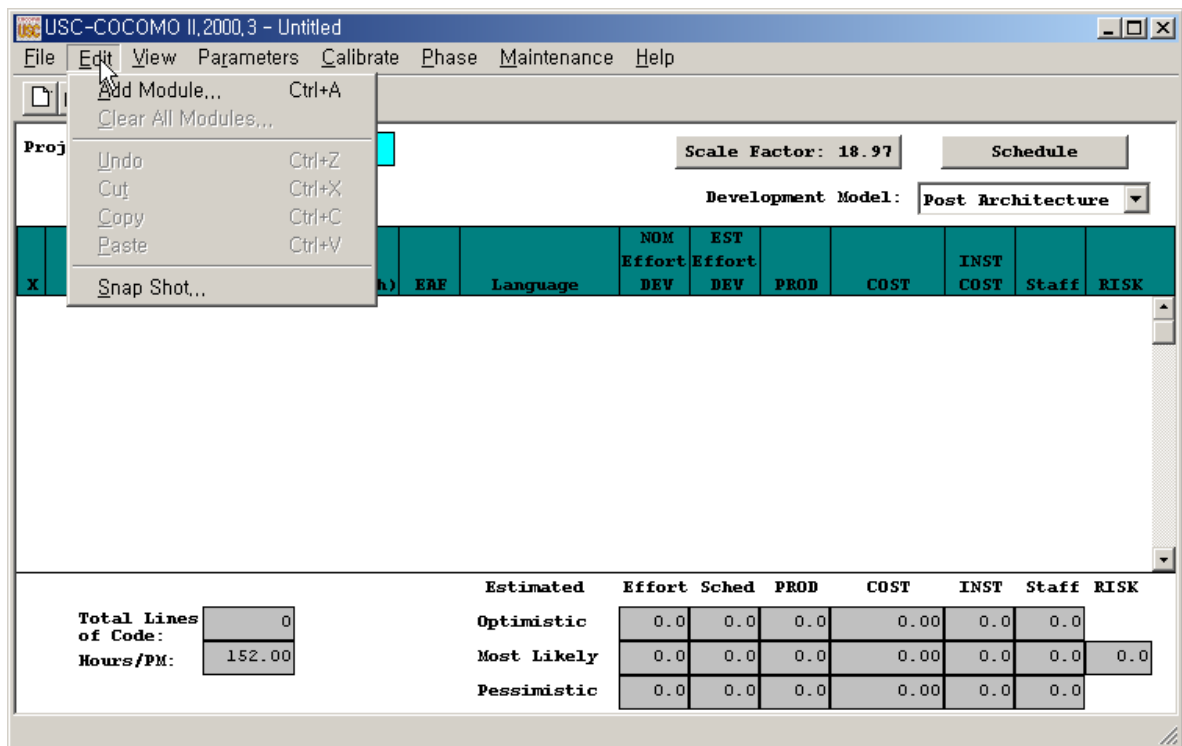
1. Choose Exit from the File menu with the mouse.
2. This causes your system to terminate the cocomo program.

Chapter 3: Edit Menu

The Edit Menu option supplies several useful commands, which will enable you to establish a project more conveniently.

To select the Edit menu and its options, click on Edit with the mouse, then the Edit menu will appear as Figure 3-1.

FIGURE 3-1 Edit Menu



3.1 Add Module

The Add Module option adds a new module to the project that is currently being worked upon by the user. This Add Module function can be done by pressing the Add Module button in the Tool bar area.

3.2 Clear All Module

The Clear All option erases all modules of the current project on the working window.

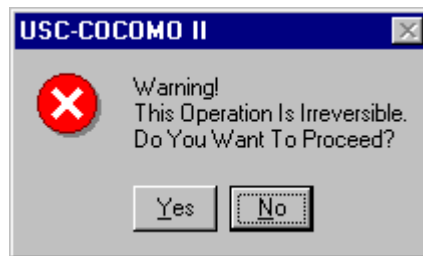
To Erase All Modules of Current Project

1. Choose Clear from the Edit menu.

During the execution of the Clear command, if some changes have occurred on the currently viewed project and have not been saved, the warning dialog box will appear as Figure 3.2.

2. If you really want to clear, click Yes. If not, click No.
3. After Clear, all modules of current project will disappear.

FIGURE 3-2 **Warning Dialog Box**



3.3 *Snapshot*

The Snapshot option enables users to compare the effort estimation change for a project so that he/she can decide to apply the change or not. This function makes COCOMO more convenient and powerful for software project decision analyses.

The Snapshot command stores the current set of modules, effort adjustment factors and all other data associated with a project. At a later time this data can be restored.

To Compare the Overall Change of a Project

1. Choose Snapshot from the Edit menu. The Snapshot dialog box will initially appear as Figure 3-3.

FIGURE 3-3 Snapshot Dialog Box-1

Current Project		EST	Sched	PROD	COST	INST	FSWP	RISK
Total EDSI:	37791	Optimistic	683.4	31.4	55.3	4090503.35	108.2	21.7
Effort (PM):	202.7	Most Likely	854.3	34.1	44.2	5113129.19	135.3	25.1
Productivity:	186.4	Pessimistic	1067.8	36.9	35.4	6391411.49	169.1	28.9

Buttons: Snap, Revert, Done, Help

In the dialog box, the lower section represents the current results for the project. The upper section is previously snapped results. The current project can be snapped by clicking upon the Snap button. After completing this action the upper and lower section of the Snapshot window will contain identical information. At this point changes can be made to the current project values after clicking upon the Done button.

2. Upon completing the modification of the project values, a comparison can be made between the previously snapped project and the modified project by clicking again upon the Snapshot option in the Edit menu.

3. Now the values in the upper part of the window will likely be different from the current values, in the lower part. To restore the upper values, click on Revert. The two sets of values are interchanged.

FIGURE 3-4 Snapshot Dialog Box-2

Snapped Project			EST	Sched	PROD	COST	INST	FSWP	RISK
Total EDSI:	37791	Optimistic	683.4	31.4	55.3	4090503.35	108.2	21.7	
Effort (PM):	202.7	Most Likely	854.3	34.1	44.2	5113129.19	135.3	25.1	28.9
Productivity:	186.4	Pessimistic	1067.8	36.9	35.4	6391411.49	169.1	28.9	
Current Project			EST	Sched	PROD	COST	INST	FSWP	RISK
Total EDSI:	40035	Optimistic	735.3	32.3	54.4	4354135.25	108.8	22.8	
Effort (PM):	216.7	Most Likely	919.1	35.0	43.6	5442669.06	135.9	26.3	28.9
Productivity:	184.7	Pessimistic	1148.9	37.9	34.8	6803336.32	169.9	30.3	

4. When finished, click the Done button.

3.4 Undo

The Undo option retracts the previous cut or paste done on a module.

To Retract Previous Cut/Paste for a Module

1. Choose Undo from the Edit menu with the mouse.
2. The changed module will go back to its previous status.

3.5 Cut

The Cut option copies a module into the cut buffer and removes it from the current project. The cut module can be used for Paste.

To Cut a Module and Remove It From the CLEF

1. Check the module which is to be cut. The Check boxes for modules are located in the leftmost column of the CLEF area. Place the mouse in the box just to the left of the module name, and click.

2. Choose Cut from the Edit menu with the mouse.
3. The cut module disappears.

3.6 Copy

The Copy option copies a module. The copied module can be used for Paste.

To Copy a Module

1. Check the module which is to be copied. The Check boxes for modules are located in the leftmost column of the CLEF area.
2. Choose Copy from the Edit menu with the mouse.
3. The cross sign in the check box disappears.

3.7 Paste

The Paste option pastes a previously copied or cut module in the CLEF.

To Paste a Previously Copied or Cut Module

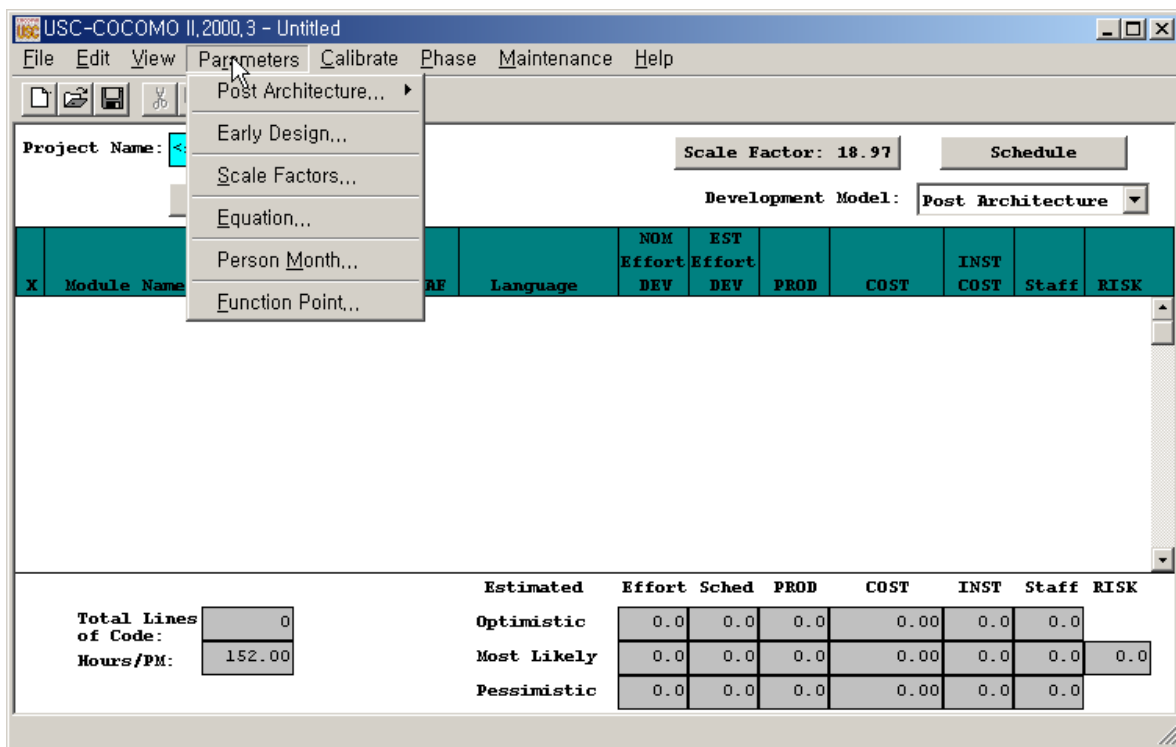
1. Check the module above which the previously copied or cut module is to be pasted. The Check boxes for modules are located in the leftmost column of CLEF area.
2. Choose Paste from the Edit menu with the mouse.
3. The pasted module appears at the checked position, and the modules lower than it were pushed one row down.
4. If there is no module checked, the Paste will attach the previously copied or cut module at the end.

Chapter 4: Parameters Menu

The Parameters menu option will enable you to look at, or change the values of effort adjustment factors, scale factors and effort/schedule estimating equations factors for the current project.

To choose the Parameters menu and its options, click on Parameters with the mouse. The Parameters menu will appear as Figure 4-1.

FIGURE 4-1 Parameters Menu



Post Architecture

There are five sub options in Post Architecture option as displayed in figure 4-1.

4.1 Product

The Product option displays five cost drivers: RELY, DATA, DOCU, CPLX, and RUSE and their corresponding ratings and multiplier values.

Select Product from the Parameters menu with the mouse. The Product Dialog Box will appear as Figure 4-2.

FIGURE 4-2 **Product Dialog Box**

The screenshot shows a dialog box titled "PRODUCT PARAMETERS" with a close button (X) in the top right corner. The dialog contains a table of input fields for five cost drivers (RELY, DATA, DOCU, CPLX, RUSE) across six rating categories (VLO, LO, NOM, HI, VHI, XHI). At the bottom are three buttons: OK, Cancel, and Help.

	VLO	LO	NOM	HI	VHI	XHI
RELY	0.75	0.88	1.00	1.15	1.40	XXXX
DATA	XXXX	0.94	1.00	1.08	1.16	XXXX
DOCU	0.85	0.93	1.00	1.08	1.17	XXXX
CPLX	0.75	0.88	1.00	1.15	1.30	1.65
RUSE	XXXX	0.89	1.00	1.16	1.34	1.56

Buttons: OK, Cancel, Help

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

4.2 Platform

The Platform option displays three cost drivers: TIME, STOR and PVOL, and their corresponding ratings and multiplier values.

Select Platform from the Parameters menu with the mouse. The Platform Dialog Box will appear as Figure 4-3.

FIGURE 4-3 **Platform Dialog Box**

	VLO	LO	NOM	HI	VHI	XHI
TIME	XXXX	XXXX	1.00	1.11	1.30	1.66
STOR	XXXX	XXXX	1.00	1.06	1.21	1.56
PVOL	XXXX	0.87	1.00	1.15	1.30	XXXX

OK Cancel Help

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

4.3 Personnel

The Personnel option displays six cost drivers: ACAP, AEXP, PCAP, PEXP, LEXP, and PCON and their corresponding ratings and multiplier values.

Select Personnel from the Parameters menu with the mouse. The Personnel Dialog Box will appear as Figure 4-4.

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

4.4 Project

The Project option displays three cost drivers: TOOL, SCED, and SITE and their corresponding ratings and multiplier values.

Select Project from the Parameters menu with the mouse. The Project Dialog Box will appear as Figure 4-5.

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

FIGURE 4-4 Personnel Dialog Box

	VLO	LO	NOM	HI	VHI	XHI
ACAP	1.50	1.22	1.00	0.83	0.67	XXXX
AEXP	1.23	1.10	1.00	0.88	0.80	XXXX
PCAP	1.37	1.16	1.00	0.87	0.74	XXXX
PEXP	1.26	1.12	1.00	0.88	0.80	XXXX
LTEX	1.24	1.11	1.00	0.90	0.82	XXXX
PCON	1.26	1.11	1.00	0.91	0.83	XXXX

OK Cancel Help

FIGURE 4-5 Project Dialog Box

	VLO	LO	NOM	HI	VHI	XHI
TOOL	1.20	1.10	1.00	0.88	0.75	XXXX
SCED	1.23	1.08	1.00	1.04	1.10	XXXX
SITE	1.24	1.10	1.00	0.92	0.85	0.79

OK Cancel Help

4.5 User Defined EAF

The User Defined EAF option displays two cost driver: USR1 and USR2, and their corresponding ratings and multipliers.

Select User EAF from the Parameters menu with the mouse. The User EAF Dialog Box will appear as Figure 4-6.

FIGURE 4-6 User Defined EAF Dialog Box

	VLO	LO	NOM	HI	VHI	XHI
USR1	1.00	1.00	1.00	1.00	1.00	1.00
USR2	1.00	1.00	1.00	1.00	1.00	1.00

OK Cancel Help

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

4.6 Early Design

The Early Design Factors option displays nine development attributes: RCPX, RUSE, PDIF, PERS, PREX, FCIL, SCED, USR1, USR2, and their corresponding ratings and values.

Select Early Design Factors from the Parameters menu with the mouse. The Early Design Factor Dialog Box will appear as Figure 4-7.

FIGURE 4-7 Early Design Factors Dialog Box

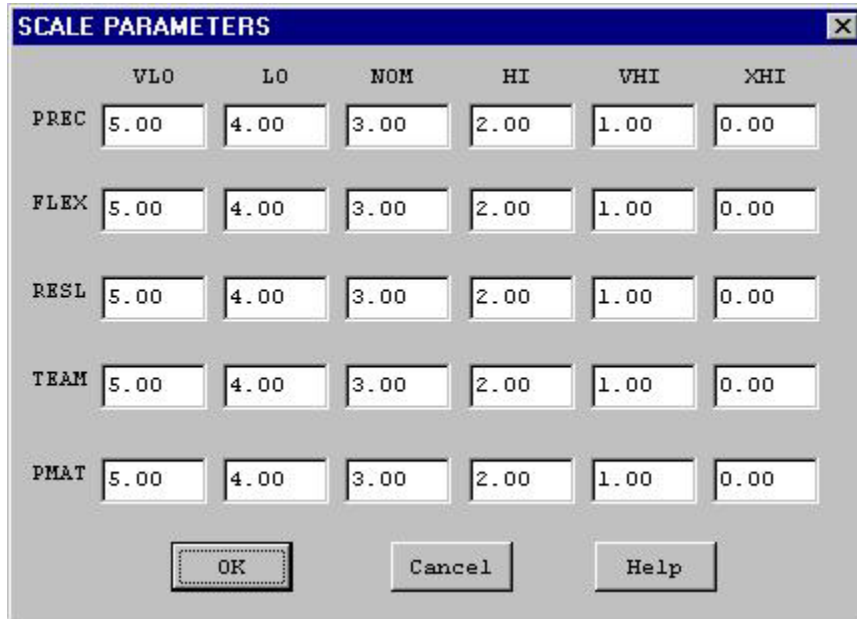
	XLO	VLO	LO	NOM	HI	VHI	XHI
RCPX	0.49	0.60	0.83	1.00	1.33	1.91	2.72
RUSE	XXXX	XXXX	0.95	1.00	1.07	1.15	1.24
PDIF	XXXX	XXXX	0.87	1.00	1.29	1.81	2.61
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.50
PREX	1.59	1.33	1.12	1.00	0.87	0.74	0.62
FCIL	1.43	1.30	1.10	1.00	0.87	0.73	0.62
SCED	XXXX	1.43	1.14	1.00	1.00	1.00	XXXX
USR1	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX
USR2	XXXX	1.00	1.00	1.00	1.00	1.00	XXXX

Buttons: OK, Reset, Cancel, Help

4.7 Scale Factors

The Scale Factors option displays five development attributes: PREC, FLEX, RESL, TEAM and PMAT, and their corresponding ratings and values.

Select Scale Factors from the Parameters menu with the mouse. The Scale Factor Dialog Box will appear as Figure 4-8.

FIGURE 4-8 Scale Factors Dialog Box

The dialog box titled "SCALE PARAMETERS" contains a table of scale factors. The columns are labeled VLO, LO, NOM, HI, VHI, and XHI. The rows are labeled PREC, FLEX, RESL, TEAM, and PMAT. Each cell contains a numerical value. At the bottom are three buttons: OK, Cancel, and Help.

	VLO	LO	NOM	HI	VHI	XHI
PREC	5.00	4.00	3.00	2.00	1.00	0.00
FLEX	5.00	4.00	3.00	2.00	1.00	0.00
RESL	5.00	4.00	3.00	2.00	1.00	0.00
TEAM	5.00	4.00	3.00	2.00	1.00	0.00
PMAT	5.00	4.00	3.00	2.00	1.00	0.00

OK Cancel Help

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click OK button.

4.8 Equation

The Equation option displays effort and schedule equations.

Select Equation from the Parameters menu with the mouse. The Equation Dialog Box will appear as Figure 4-9.

FIGURE 4-9 **Equation Dialog Box**

Equation Parameters

Exponent Equation

$$B = 1.01 + 0.01 (SF1 + \dots + SF5)$$

Effort Equation

$$PM = EM1 * \dots * EM17 * 3 * (Size)^B + (ASLOC * (AT/100) / ATPROD)$$

Schedule Equation

$$TDEV = (3 * PM^{(0.33 + 0.2 * (B - 1.01))} * (SCED\%/100)$$

OK Cancel Help

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

FIGURE 4-10 **Person Month Dialog Box**

Hours/Person Month - Default model value used

Hours/PM

152.00 Hours

OK Reset Cancel Help

4.9 Person Month

The Person Month option modifies the value of hours per person and month.

Type the value in field and click “OK” to apply new hour value to project. Figure 4-10 shows person month option box.

4.10 Function Point

The Function Point Factors option displays five function types: Internal Logical Files, External Interface Files, External Inputs, External Outputs, External Inquiries, and their corresponding ratings and weight values.

Select Function Point from the Parameters menu with the mouse. The Function Point Dialog Box will appear as Figure 4-11.

FIGURE 4-11 **Function Point Dialog Box**

The dialog box, titled "Function Point - Default model values used", contains a table with five rows of function types and three columns of rating values (Low, Average, High). Each value is in a text input field. At the bottom are four buttons: OK, Reset, Cancel, and Help.

Function Type	Low	Average	High
Internal Logical Files	7	10	15
External Interface Files	5	7	10
External Inputs	3	4	6
External Outputs	4	5	7
External Inquiries	3	4	6

To modify these values, go straight to those edit boxes and type new values. When finished with the modification, click the OK button.

Chapter 5: Calibrate Menu

COCOMOII now has the ability to archive your own software project data. Using this data, COCOMOII will compute various coefficients and exponents involved in the effort and schedule equations. This will make your COCOMOII estimates even more reliable.

Each software project to be archived is described as a complete COCOMOII project. It may include multiple modules, each with their own SLOC estimate and EAF factors. In addition, a software project consists of a name, date/time, actual effort and actual schedule. The actual effort and actual schedule must be supplied by the COCOMOII user. Entering revised values for effort and schedule are always possible. Effort is given in units of person/months. Schedule is given in units of months.

FIGURE 5-1 Calibrate Menu

USC-COCOMO II, 2000.3 - Untitled

File Edit View Parameters **Calibrate** Phase Maintenance Help

File Load...
File Save...
File Save As...
Project...
Compute...

Project Name: **Project A** Scale Factor: **18.97** Schedule
Project No. Development Model: **Post Architecture**

X	Module Name	Module Size	Rate (\$/month)	EAF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Module A	S:10000	0.00	1.00	Object-Orient	43.7	43.7	228.6	0.00	0.0	2.1	0.0
	Module B	F:25000	0.00	1.00	Shell Script	109.3	109.3	228.6	0.00	0.0	5.3	0.0
	Module C	A:18739	0.00	1.00	PASCAL	82.0	82.0	228.6	0.00	0.0	3.9	0.0

Total Lines of Code: **53739**
Hours/PM: **152.00**

Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
Optimistic	188.0	19.4	285.8	0.00	0.0	9.7	
Most Likely	235.0	20.8	228.6	0.00	0.0	11.3	0.0
Pessimistic	293.8	22.4	182.9	0.00	0.0	13.1	

5.1 File Load

The Calibrate File Load option is used to retrieve a calibration project file as well as loading all project data on the working project window(Figure5-2).

FIGURE 5-2 Load Calibration Dialog Box



5.2 File Save

The Calibrate File Save command saves the current calibration data in the file whose name was previously identified using File Save As.

5.3 File Save As

The Calibrate File Save As command stores the current calibration data as a *.cal file, which has a different file name from the current file. This command works precisely the same as the File Save As for *.est and *.mod files (see Figure 5-4).

FIGURE 5-3 Save Calibration Dialog Box

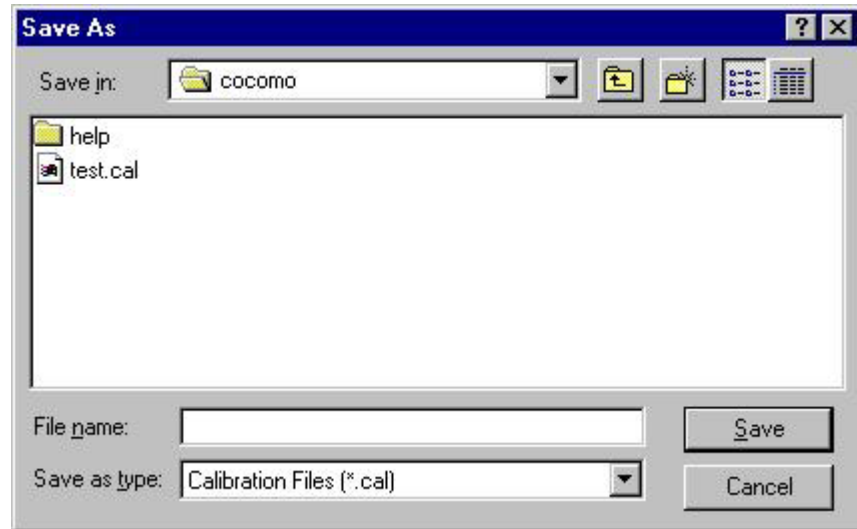


FIGURE 5-4 Save As Calibration Dialog Box



5.4 Project

A window appears (shown in Figure 5-5) which displays the archived project data.

- To remove the window, click on Cancel.
- To delete an existing entry, first place an x at the leftmost end of the row and click on Delete. A warning box appears as shown in Figure 5-6.
- To display the entire set of values for an archived project, click on Display. Since the display of an archived project eliminates the display of any existing CLEF data, a warning message appears as shown in Figure 5-7.
- To insert a new archived project from the CLEF, click on Insert.

FIGURE 5-5 Projects Dialog Box

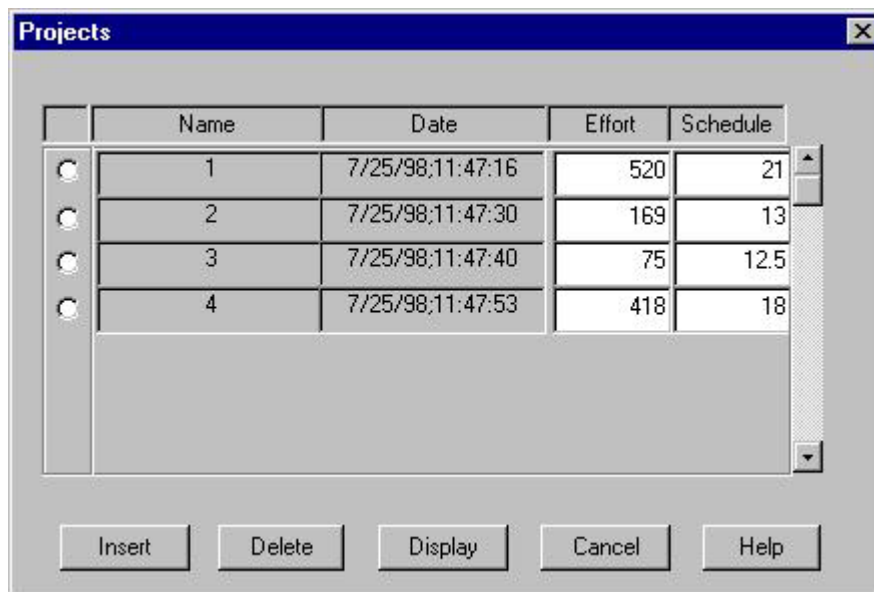


FIGURE 5-6 Delete Warning Dialog

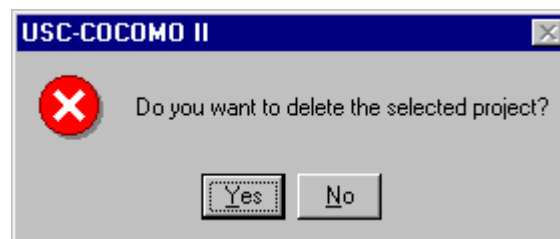
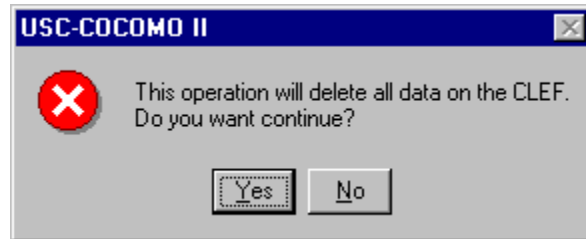


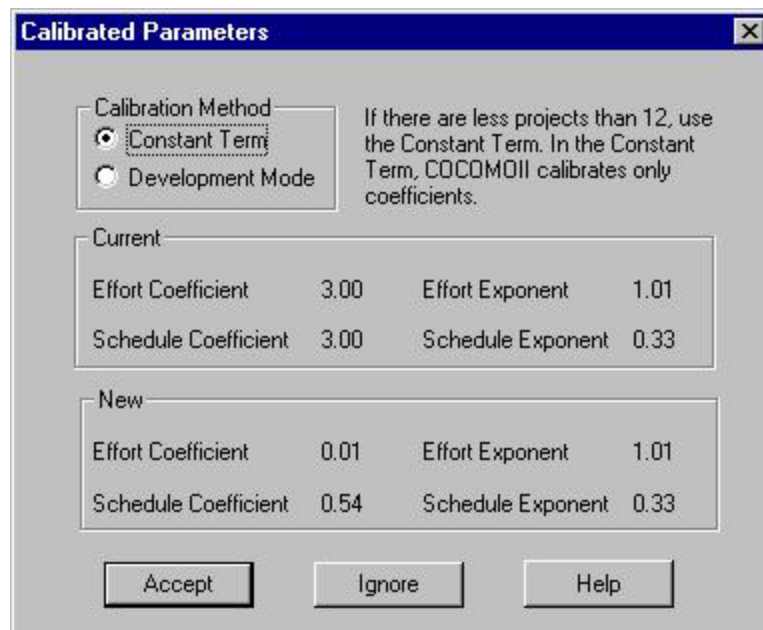
FIGURE 5-7 **Display Warning Dialog**



5.5 Compute

This command takes all of the data that has been archived and uses it to compute new constant and exponent values for the effort equation and similarly for the schedule equation. There are two options to calibrate equation parameters. One is the Constant Term and the other one is Development Mode. Those two options are explained below in detail. They are displayed in this window and compared to the values currently used by COCOMOII. To get COCOMOII to use these values, click on Accept, as shown Figure 5-8.

FIGURE 5-8 **Compute Dialog Box**



Chapter 6: Phase Distribution

The Phase Distribution is one of the menu selections in the menu bar that can be accessed by either clicking upon Phase Distribution in the main menu. Its function is to display a breakdown of the software effort and schedule into the phases of the development cycle. There are two models for the phases; Waterfall Model and MBASE Model. The phases of Waterfall model are plans & requirements, design, programming and integration & test. And the phases of MBASE model are Inception, Elaboration, Construction, and Transition. These phases are described as follows:

Waterfall Model

Plans & Requirements - In this phase, a statement for the required functions, interfaces and performance is created. These expectations are used to define the capabilities of the software product as expressed by representatives of all interested parties.

Product Design - In this phase, a hardware/software architecture, control structure and data structure for the product are defined. A draft of the user's manual and test plans are also created during this phase.

Programming - In this phase, the design of the previous phase is implemented in the creation of complete sets of software components.

Integration & Test - In this phase, the various software components are brought together in order to achieve a properly functioning software product composed of loosely coupled modules. The requirements as defined in the first phase are used to determine the fitness of the delivered product.

MBASE Model

Inception phase - The first phase of the software life cycle, where the seed idea for the development is brought to the point of being sufficiently well founded to warrant entering into the elaboration. Activities are

- Understanding the organization background and analyzing the problem domain
- Identifying feasible solutions
- Understanding the requirements of the life cycle and obtaining concurrence from the system stakeholders

Elaboration - most of the system requirements are determined and the system architecture is designed.

- Identifying and resolving the critical risks
- Analyzing the problem domain
- Identifying and enunciating a feasible architecture as the proposed solution
- Determining the feasibility of proposed solution
- Estimating the requirements of the life cycle and planning for their use
- Identifying responsibilities of the system stakeholders

Construction - In this phase, the design of the previous phase is implemented in the creation of complete sets of software components.

- Requirements Management
- Detailed Design
- Coding
- Unit and Integration Testing
- Inspections
- Configuration Management
- Quality Assurance

Transition - In this phase, the transition plan, user manual, support plan will be done.

The phase distribution menu has two selections: project phase distribution and module phase distribution. The project phase distribution allows the user to view the development phases for the entire project all together or individually. The module phase distribution allows the user to view the development phases for a particular module either all together or individually. These two variations of phase distribution are discussed further in this chapter under sections 6.1 and 6.2 in this chapter. (figure 6-1)

NOTE: These phase distribution estimates are retained from the COCOMO81 model, which assumed a waterfall (sequentially phased) process model. If your project's process model is not a waterfall, these phase distributions are either inapplicable or need to be reinterpreted.

6.1 Waterfall

6.1.1 Project Phase Distribution

In order to view the phase distribution of an entire project, the user can click on the Project Phase Distribution button under the Phase Distribution menu (see FIGURE 6-1). Four formats for viewing will appear in another menu: overall phase, plan & requirements, programming, and integration & test. Each of these menu selections will be discussed in sections 6.1.1 - 6.1.4, respectively. The phase distribution of plan & requirements, programming and integration & test are broken down into sub-phases. These phases include: requirements analysis, product design, programming, test planning, verification & validation, project office, CM/QA, and manuals. For each of these sub-phases the percentage of the phase, the estimated effort, the estimated schedule, and the estimated FSWP is displayed. A description of each of these sub-phases follows:

Requirements analysis: Determination, specification review and update of software functional, performance, interface, and verification requirements.

FIGURE 6-1 **Phase Distribution Sub-menu**

USC-COCOMO II, 2000.3 - Untitled

File Edit View Parameters Calibrate Phase Maintenance Help

Waterfall Project... MBASE

Project Name: Project A

Project Notes

X	Module Name	Module Size	LABOR Rate (\$/month)	ERF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Module A	S:10000	0.00	1.00	Object-Orient	43.7	43.7	228.6	0.00	0.0	2.1	0.0
	Module B	F:25000	0.00	1.00	Shell Script	109.3	109.3	228.6	0.00	0.0	5.3	0.0
	Module C	A:18739	0.00	1.00	PASCAL	82.0	82.0	228.6	0.00	0.0	3.9	0.0

Total Lines of Code: 53739
Hours/PM: 152.00

Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
Optimistic	188.0	19.4	285.8	0.00	0.0	9.7	
Most Likely	235.0	20.8	228.6	0.00	0.0	11.3	0.0
Pessimistic	293.8	22.4	182.9	0.00	0.0	13.1	

Product Design: Determination, specification, review and update of hardware-software architecture, program design, and database design.

Programming: Detailed design, code, unit test, and integration of individual computer program components. Includes programming personnel planning, tool acquisitions, database development, component level documentation, and intermediate level programming management.

Test Planning: Specification, review, and update of product test and acceptance test plans. Acquisition of associated test drivers, test tools, and test data.

Verification & Validation(V&V): Performance of independent requirements validation, design V&V, product test, and acceptance test. Acquisition of requirements and design V&V tools. "Are we building the product right?" and "are we building the right product?"

Project Office Functions: Project level management functions. Includes project level planning and control, contract and subcontract management, and customer interface.

Configuration Management and Quality Assurance (CM/QA): Configuration management includes product identification, change control, status accounting, operation of program support library, development and monitoring of end item acceptance plan. Quality assurance includes development and monitoring of project standards, and technical audits of software products and processes.

Manuals: Development and update of users' manuals, operators' manuals and maintenance manuals.

6.1.1.1 Overall Project Phase Distribution

The overall phase distribution allows the user to view an entire project's estimated effort, schedule and number of personnel needed for phase completion. Upon clicking on "Overall Phase," a window will be displayed showing the phase breakdown of the current project in COCOMO (see FIGURE 6-2). This window displays the project name, project SLOC, and the total estimated effort for the project. Looking at FIGURE 6-1, this information can be seen in the upper left corner of the window.

FIGURE 6-2 Phase Distribution window displaying a sample project's overall phase distribution

Overall Phase Distribution					
=====			=====		
PROJECT			Test		
SLOC			37791		
TOTAL EFFORT			853.045 Person Months		
=====					
	PCNT	EFFORT (PM)	PCNT	SCHEDULE	Staff
Plans And Requirements	7.000	59.713	20.121	6.856	8.710
Product Design	17.000	145.018	26.060	8.880	16.331
Programming	57.819	493.222	47.759	16.274	30.308
- Detailed Design	24.940	212.747	----	----	----
- Code and Unit Test	32.879	280.476	----	----	----
Integration and Test	25.181	214.805	26.181	8.921	24.078

In addition, each phase of the project's development cycle is represented by the estimated effort, the estimated schedule and the estimated number of personnel needed for phase completion. Again looking at FIGURE 6-2, the information has been separated into columns. The first column displays the phase name. The second column displays the percentage that the corresponding phase takes in the estimated effort. The third column displays the estimated effort for each phase. The fourth column displays the percentage of the estimated schedule that is dedicated to the corresponding phase's

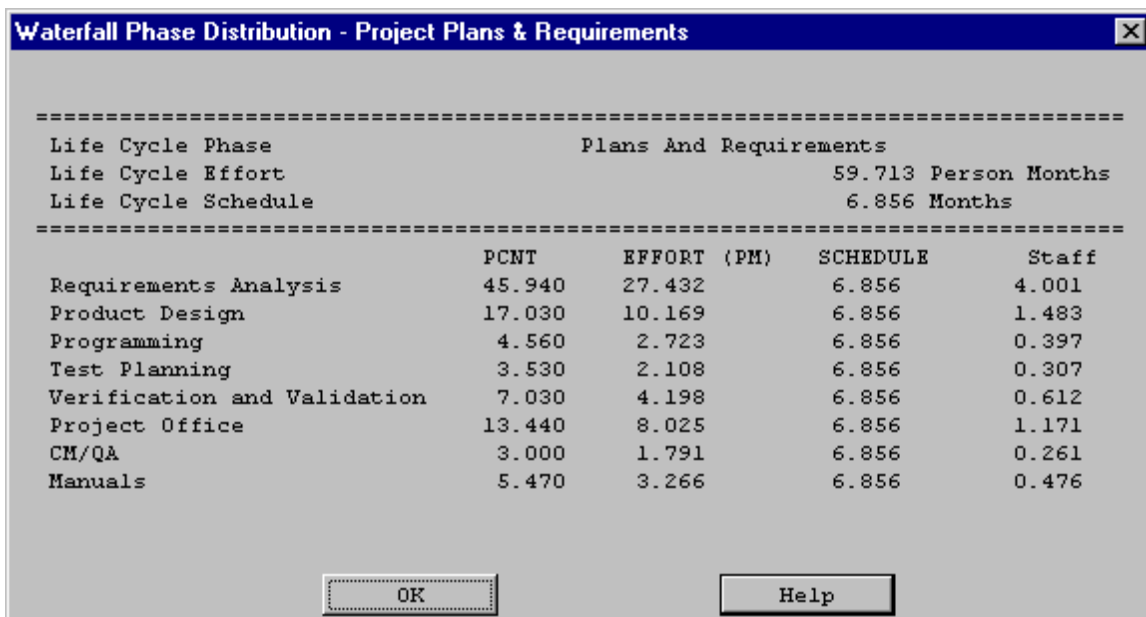
completion. The fifth column displays the estimated schedule for phase completion. And the last column displays the estimated number of personnel needed for phase completion (Staff).

Note: The programming phase has been broken down into two additional phases: "Detailed Design" and "Code and Unit Test." The detailed design is a follow-up to the product design phase. In this sub phase, those points developed in the product design are elaborated to a point necessary to breakdown agreed functions into units necessary for coding. The code and unit test sub-phases house the actual coding effort of the individual units of code. The testing of these units (upon completion) is also encompassed within this sub phase.

6.1.1.2 Plans and Requirements Project Phase Distribution

The plans and requirements phase distribution allows the user to view the components of this particular phase. When the Plans and Requirements distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-3 is displayed. This window displays the following information: project phase, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements analysis, product design, programming, test planning, verification & validation, project office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-3. To exit from this window click the OK button.

FIGURE 6-3 Plans and Requirements Phase window for the overall project



6.1.1.3 Programming Project Phase

The programming phase distribution allows the user to view the components of this particular phase. When the Programming distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-4 is displayed. This window displays the following information: project name, the total project SLOC, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements analysis, product design, programming, test planning, verification & validation, project office, CM/QA, and manuals. These activities are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-4. To exit from this window click the OK button.

FIGURE 6-4 Programming Phase window for the overall project

Waterfall Phase Distribution - Project Programming				
=====				
Life Cycle Phase	Programming			
Life Cycle Effort	493.222 Person Months			
Life Cycle Schedule	16.274 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Analysis	4.000	19.729	16.274	1.212
Product Design	8.000	39.458	16.274	2.425
Programming	56.500	278.671	16.274	17.124
Test Planning	5.030	24.810	16.274	1.525
Verification and Validation	8.030	39.607	16.274	2.434
Project Office	6.470	31.911	16.274	1.961
CM/QA	6.500	32.059	16.274	1.970
Manuals	5.470	26.978	16.274	1.658

OK Help

6.1.1.4 Product Design Phase

The product design phase distribution allows the user to view the components of this particular phase. When the Product Design distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-5 is displayed. This window displays the following information: project name, the total project SLOC, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements analysis, product design, programming, test planning, verification & validation, project office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-5. To exit from this window click the OK button.

FIGURE 6-5 Product Design window for the overall project

Waterfall Phase Distribution - Project Product Design

=====

Life Cycle Phase

Life Cycle Effort

Life Cycle Schedule

Product Design

145.018 Person Months

8.880 Months

=====

	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Analysis	12.500	18.127	8.880	2.041
Product Design	41.000	59.457	8.880	6.696
Programming	13.030	18.896	8.880	2.128
Test Planning	5.530	8.020	8.880	0.903
Verification and Validation	7.030	10.195	8.880	1.148
Project Office	10.940	15.864	8.880	1.787
CM/QA	2.500	3.625	8.880	0.408
Manuals	7.470	10.833	8.880	1.220

=====

OK

Help

6.1.1.5 Integration and Test Project Phase

The integration & test phase distribution allows the user to view the components of this particular phase. When the Integration and Test distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-6 is displayed. This window displays the following information: project name, the total project SLOC, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements analysis, product design, programming, test planning, verification & validation, project office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-6. To exit from this window click the OK button.

FIGURE 6-6 Integration and Test window for the overall project

Waterfall Phase Distribution - Project Integration & Test

=====

Life Cycle Phase

Life Cycle Effort

Life Cycle Schedule

Integration and Test

214.805 Person Months

8.921 Months

=====

PCNT

EFFORT (PM)

SCHEDULE

Staff

Requirements Analysis

2.500

5.370

8.921

0.602

Product Design

5.000

10.740

8.921

1.204

Programming

37.121

79.737

8.921

8.938

Test Planning

3.000

6.444

8.921

0.722

Verification and Validation

29.440

63.238

8.921

7.089

Project Office

7.470

16.046

8.921

1.799

CM/QA

8.000

17.184

8.921

1.926

Manuals

7.470

16.046

8.921

1.799

=====

OK

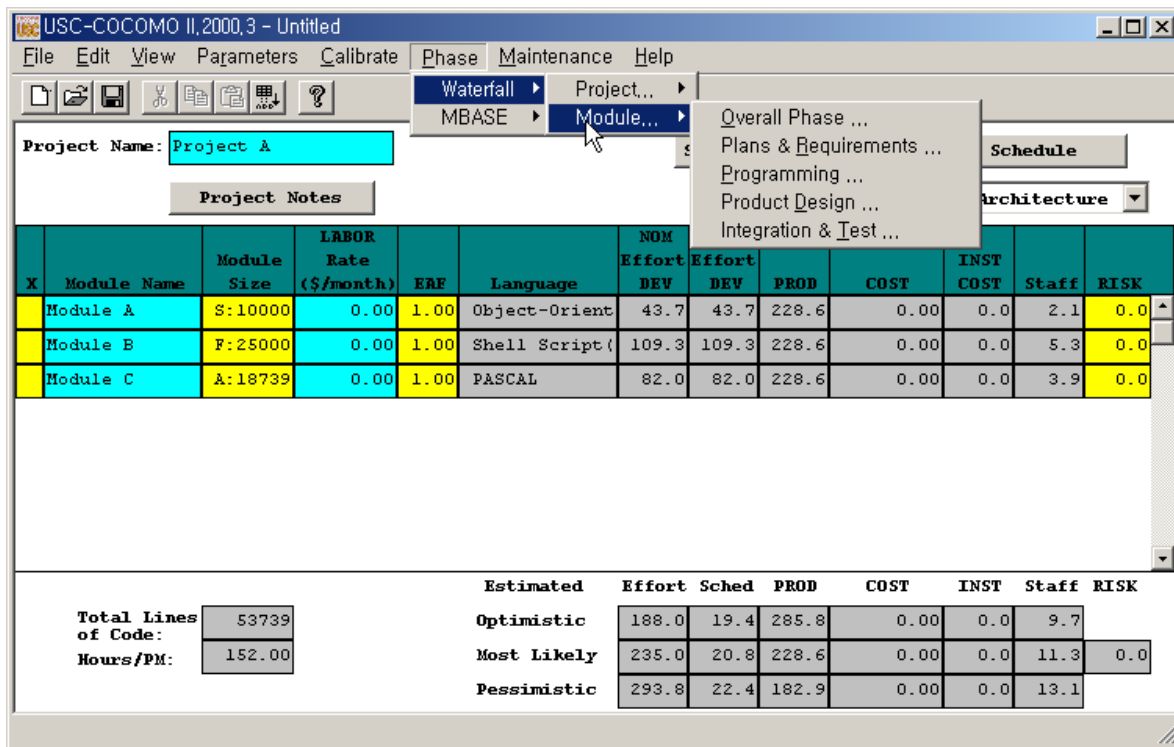
Help

6.1.2 Module Phase Distribution

Four formats for viewing will appear in another menu: overall phase, plan & requirements, programming, and integration & test (see FIGURE 6-7). Each of these menu selections will be discussed in sections 6.2.1 - 6.2.4, respectively. The phase distribution of plan & requirements, programming and integration & test are broken down into activities. These activities include: requirements analysis, product design, programming, test planning, verification & validation, Module office, CM/QA, and manuals. For each of these activities, the percentage of the phase, the estimated effort, the estimated schedule, and the estimated FSWP is displayed. A description of each of these activities follows:

Requirements analysis: Determination, specification review and update of software functional, performance, interface, and verification requirements.

Product Design: Determination, specification, review and update of hardware-software architecture, program design, and database design.

FIGURE 6-7 Phase Distribution Module Sub-menu

Programming: Detailed design, code, unit test, and integration of individual computer program components. Includes programming personnel planning, tool acquisitions, database development, component level documentation, and intermediate level programming management.

Test Planning: Specification, review, and update of product test and acceptance test plans. Acquisition of associated test drivers, test tools, and test data.

Verification & Validation(V&V): Performance of independent requirements validation, design V&V, product test, and acceptance test. Acquisition of requirements and design V&V tools. "Are we building the product right?" and "are we building the right product?"

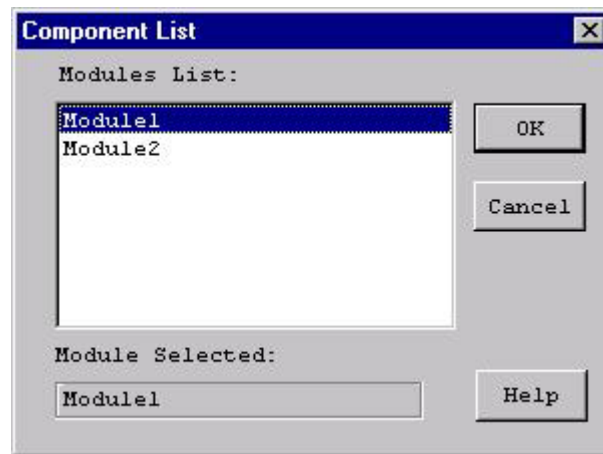
Module Office Functions: Module level management functions. Includes Module level planning and control, contract and subcontract management, and customer interface.

Configuration Management and Quality Assurance (CM/QA): Configuration management includes product identification, change control, status accounting, operation of program support library, development and monitoring of end item acceptance plan. Quality assurance includes development and monitoring of Module standards, and technical audits of software products and processes.

Manuals: Development and update of users' manuals, operators' manuals and maintenance manuals.

In order to view the phase distribution of an entire Module, the user can click on the Module Phase Distribution button under the Phase Distribution menu. When choosing any of the views of phase distribution, you will be confronted with a module selection window (see FIGURE 6-8). At this point, you may choose which module is to be viewed by clicking on the desired module name, which will be highlighted after the click. Click the OK button in order to initiate phase distribution of the chosen module.

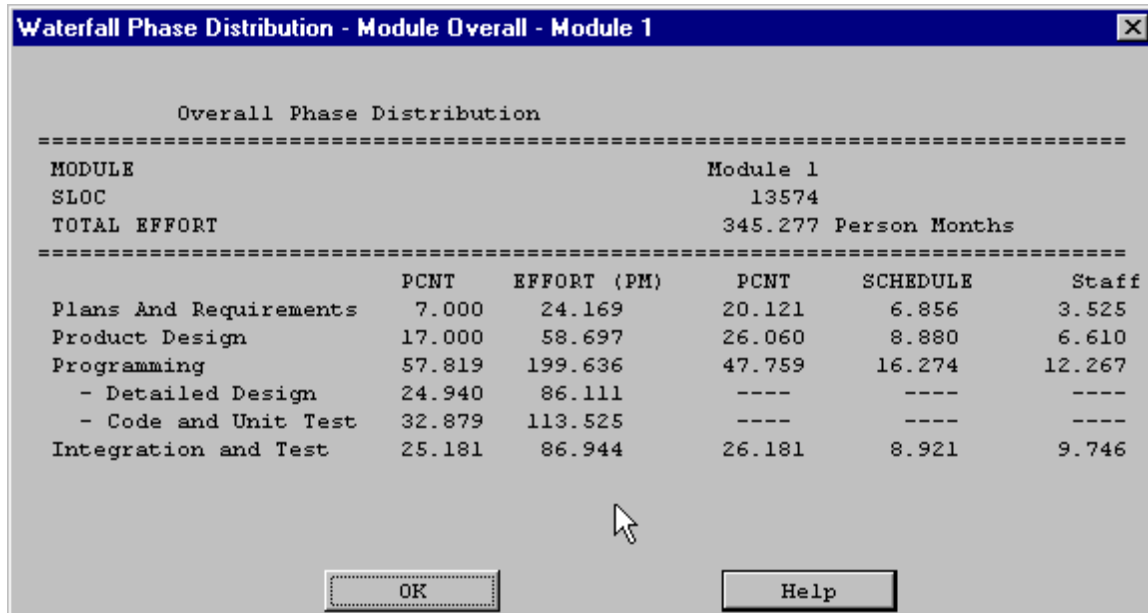
FIGURE 6-8 **Module selection window**



6.1.2.1 Overall Module Phase Distribution

The overall phase distribution allows the user to view an entire Module's estimated effort, schedule and number of personnel needed for phase completion. Upon clicking on "Overall Phase," a window will be displayed showing the phase breakdown four formats for viewing will appear in another menu: overall phase, plan & requirements, programming, and integration & test (see FIGURE 6-9). To exit from this window click the OK button.

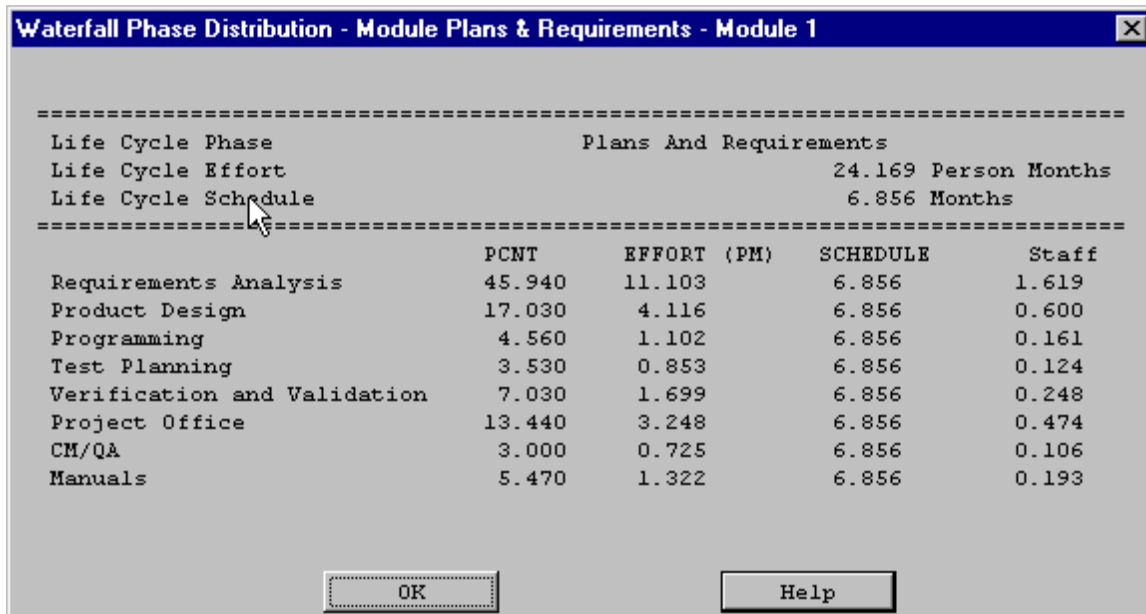
In addition, each phase of the Module's development cycle is represented by the estimated effort, the estimated schedule and the estimated number of personnel needed for phase completion. Again looking at FIGURE 6-9, the information has been separated into columns. The first column displays the phase name. The second column displays the percentage that the corresponding phase takes in the estimated effort. The third column displays the estimated effort for each phase. The fourth column displays the percentage of the estimated schedule that is dedicated to the corresponding phase's completion. The fifth column displays the estimated schedule for phase completion. And the last column displays the estimated number of personnel needed for phase completion (FSWP).

FIGURE 6-9 Phase Distribution window displaying a sample Module's overall phase distribution

Note: The programming phase has been broken down into two additional phases: "Detailed Design" and "Code and Unit Test." The detailed design is a follow-up to the product design phase. In this sub phase, those points developed in the product design are elaborated to a point necessary to breakdown agreed functions into units necessary for coding. The code and unit test sub phase houses the actually coding effort of the individual units of code. The testing of these units (upon completion) is also encompassed within this sub phase.

6.1.2.2 Plans and Requirements Module Phase Distribution

The plans and requirements phase distribution allows the user to view the components of this particular phase. When the Plans and Requirements distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-10 is displayed. This window displays the following information: Module name, the total Module SLOC, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements analysis, product design, programming, test planning, verification & validation, Module office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-10. To exit from this window click the OK button.

FIGURE 6-10 Plans and Requirements Phase window for the overall Module

6.1.2.3 Programming Module Phase Distribution

The programming phase distribution allows the user to view the components of this particular phase. When the Programming distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-11 is displayed. This window displays the following information: Module name, the total Module SLOC, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activity's requirements analysis, product design, programming, test planning, verification & validation, Module office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-11. To exit from this window click the OK button.

6.1.2.4 Product Design Phase Distribution

The product design phase distribution allows the user to view the components of this particular phase. When the Product Design distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-12 is displayed. This window displays the following information: Module name, the total Module SLOC, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements analysis, product design, programming, test planning, verification & validation, Module office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-12. To exit from this window click the OK button.

FIGURE 6-11 Programming Phase window for the overall Module

Waterfall Phase Distribution - Module Programming - Module 1

Life Cycle Phase

Life Cycle Effort

Life Cycle Schedule

Programming

199.636 Person Months

16.274 Months

	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Analysis	4.000	7.985	16.274	0.491
Product Design	8.000	15.971	16.274	0.981
Programming	56.500	112.794	16.274	6.931
Test Planning	5.030	10.042	16.274	0.617
Verification and Validation	8.030	16.031	16.274	0.985
Project Office	6.470	12.916	16.274	0.794
CM/QA	6.500	12.976	16.274	0.797
Manuals	5.470	10.920	16.274	0.671

OK

Help

FIGURE 6-12 Product Design window for the overall Module

Waterfall Phase Distribution - Module Product Design - Module 1

=====				
Life Cycle Phase	Product Design			
Life Cycle Effort	58.697 Person Months			
Life Cycle Schedule	8.880 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Analysis	12.500	7.337	8.880	0.826
Product Design	41.000	24.066	8.880	2.710
Programming	13.030	7.648	8.880	0.861
Test Planning	5.530	3.246	8.880	0.366
Verification and Validation	7.030	4.126	8.880	0.465
Project Office	10.940	6.421	8.880	0.723
CM/QA	2.500	1.467	8.880	0.165
Manuals	7.470	4.385	8.880	0.494
=====				

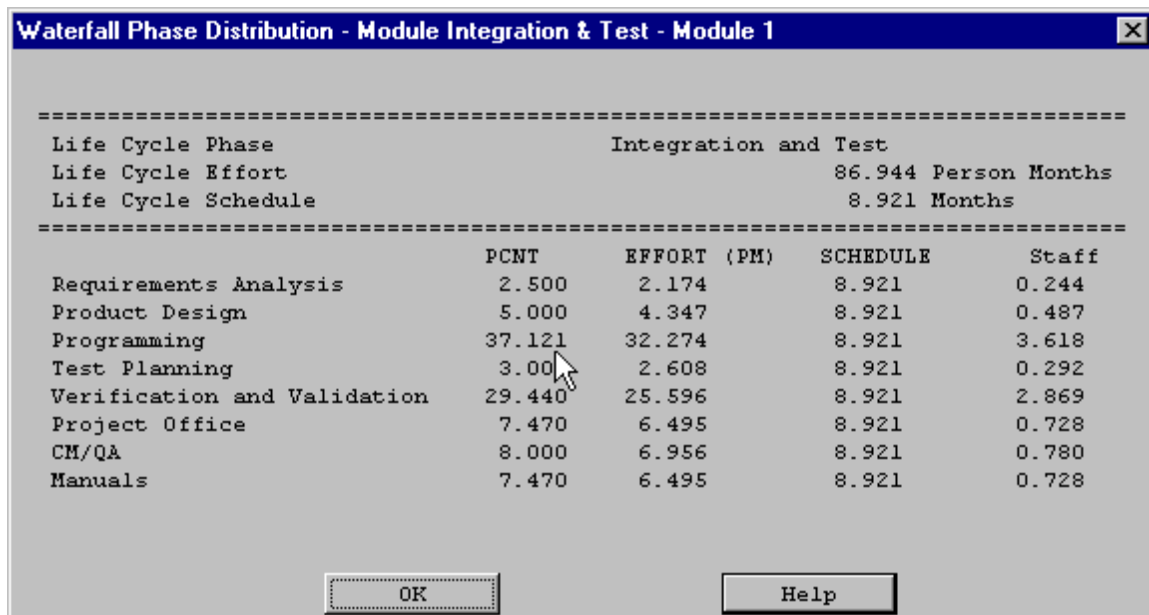
OK

Help

6.1.2.5 Integration and Test Module Phase Distribution

The integration & test phase distribution allows the user to view the components of this particular phase. When the Integration and Test distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-13 is displayed. This window displays the following information: Module name, the total Module SLOC, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements analysis, product design, programming, test planning, verification & validation, Module office, CM/QA, and manuals. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-13. To exit from this window click the OK button.

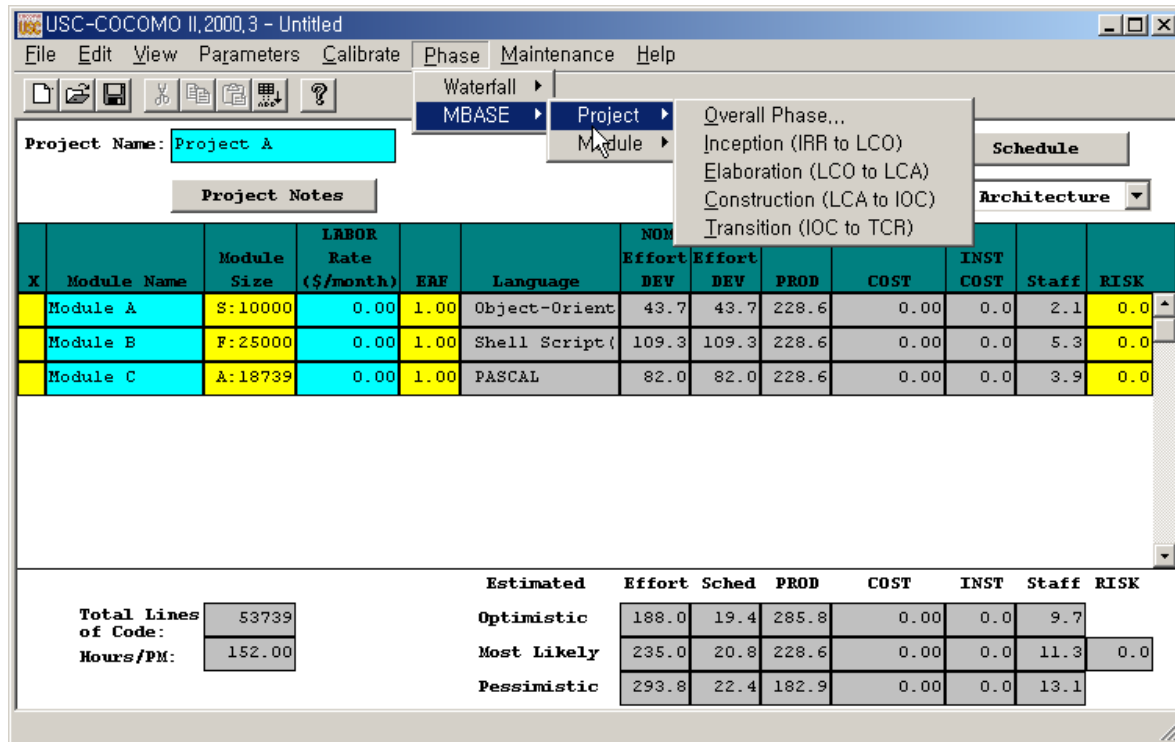
FIGURE 6-13 Integration & Test window for the overall Module



6.2 MBASE

6.2.1 Project Phase Distribution

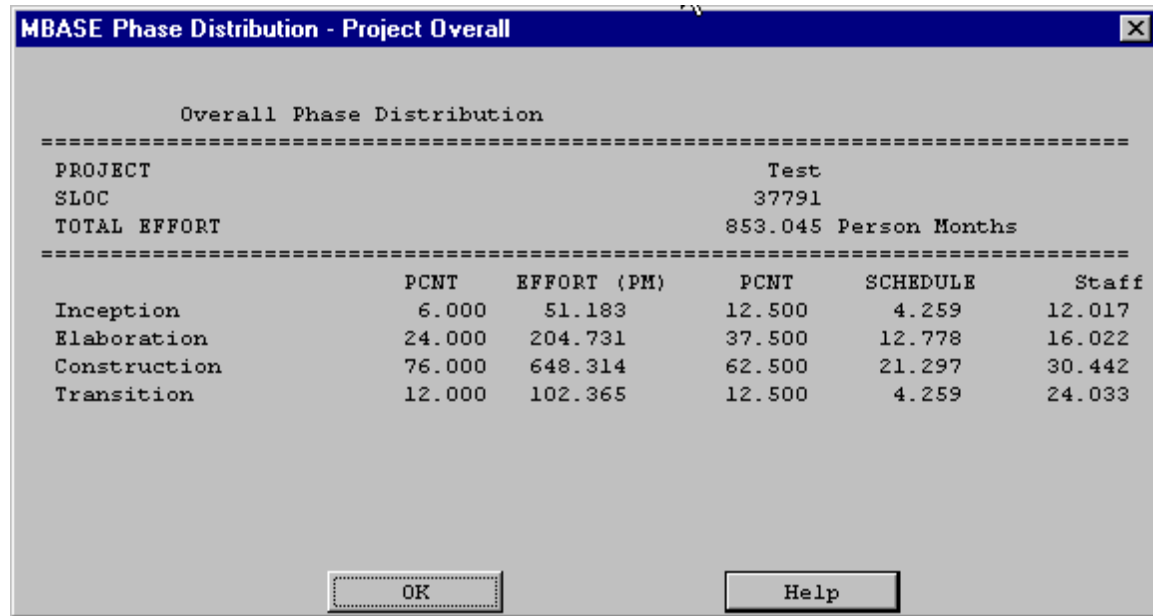
In order to view the phase distribution of an entire project, the user can click on the Project Phase Distribution button under the Phase Distribution menu (see FIGURE 6-14). Five formats for viewing will appear in another menu: overall phase, Inception, Elaboration, Construction, and Transition. Each of these menu selections will be discussed in sections 6.2.1.1 - 6.2.1.5, respectively.

FIGURE 6-14 Phase Distribution Sub-menu

6.2.1.1 Overall Project Phase Distribution

The overall phase distribution allows the user to view an entire project's estimated effort, schedule and number of personnel needed for phase completion. Upon clicking on "Overall Phase," a window will be displayed showing the phase breakdown of the current project in COCOMO (see FIGURE 6-15). This window displays the project name, project SLOC, and the total estimated effort for the project. Looking at FIGURE 6-14, this information can be seen in the upper left corner of the window.

In addition, each phase of the project's development cycle is represented by the estimated effort, the estimated schedule and the estimated number of personnel needed for phase completion. Again looking at FIGURE 6-15, the information has been separated into columns. The first column displays the phase name. The second column displays the percentage that the corresponding phase takes in the estimated effort. The third column displays the estimated effort for each phase. The fourth column displays the percentage of the estimated schedule that is dedicated to the corresponding phase's completion. The fifth column displays the estimated schedule for phase completion. And the last column displays the estimated number of personnel needed for phase completion (Staff).

FIGURE 6-15 Phase Distribution window displaying a sample project's overall phase distribution

Note: The programming phase has been broken down into two additional phases: "Detailed Design" and "Code and Unit Test." The detailed design is a follow-up to the product design phase. In this sub phase, those points developed in the product design are elaborated to a point necessary to breakdown agreed functions into units necessary for coding. The code and unit test sub-phases house the actual coding effort of the individual units of code. The testing of these units (upon completion) is also encompassed within this sub phase.

6.2.1.2 Inception Project Phase Distribution

The inception phase distribution allows the user to view the components of this particular phase. When the inception distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-16 is displayed. This window displays the following information: project phase, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-16. To exit from this window click the OK button.

FIGURE 6-16 Inception Phase window for the overall project

MBASE Phase Distribution - Project Inception (IRR to CO)				
=====				
Life Cycle Phase	Inception			
Life Cycle Effort	51.183 Person Months			
Life Cycle Schedule	4.259 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	38.000	19.449	4.259	4.566
Analysis and Design	19.000	9.725	4.259	2.283
Implementation	8.000	4.095	4.259	0.961
Test	8.000	4.095	4.259	0.961
=====				
Management	14.000	7.166	4.259	1.682
Environment	10.000	5.118	4.259	1.202
Deployment	3.000	1.535	4.259	0.360
=====				

OK Help

6.2.1.3 Elaboration Project Phase

The elaboration phase distribution allows the user to view the components of this particular phase. When the elaboration distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-17 is displayed. This window displays the following information: project phase, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements capture, analysis and design, implementation, test, management, environment and deployment. These activities are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-17. To exit from this window click the OK button.

6.2.1.4 Construction Design Phase

The construction design phase distribution allows the user to view the components of this particular phase. When the construction distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-18 is displayed. This window displays the following information: project phase, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-18. To exit from this window click the OK button.

FIGURE 6-17 Elaboration Phase window for the overall project

MBASE Phase Distribution - Project Elaboration (LCO to LCA)

=====				
Life Cycle Phase	Elaboration			
Life Cycle Effort	204.731 Person Months			
Life Cycle Schedule	12.778 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	18.000	36.852	12.778	2.884
Analysis and Design	36.000	73.703	12.778	5.768
Implementation	13.000	26.615	12.778	2.083
Test	10.000	20.473	12.778	1.602
=====				
Management	12.000	24.568	12.778	1.923
Environment	8.000	16.378	12.778	1.282
Deployment	3.000	6.142	12.778	0.481
=====				

OK

Help

FIGURE 6-18 Construction window for the overall project

MBASE Phase Distribution - Project Construction (LCA to IOC)

=====				
Life Cycle Phase	Construction			
Life Cycle Effort	648.314 Person Months			
Life Cycle Schedule	21.297 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	8.000	51.865	21.297	2.435
Analysis and Design	16.000	103.730	21.297	4.871
Implementation	34.000	220.427	21.297	10.350
Test	24.000	155.595	21.297	7.306
=====				
Management	10.000	64.831	21.297	3.044
Environment	5.000	32.416	21.297	1.522
Deployment	3.000	19.449	21.297	0.913
=====				

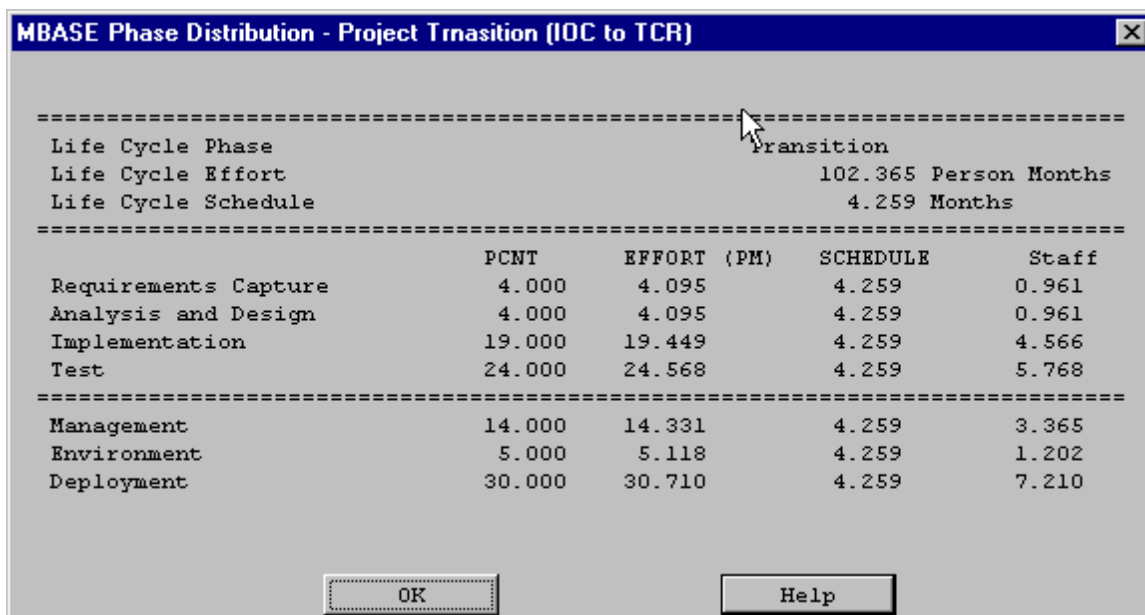
OK

Help

6.2.1.5 Transition Project Phase

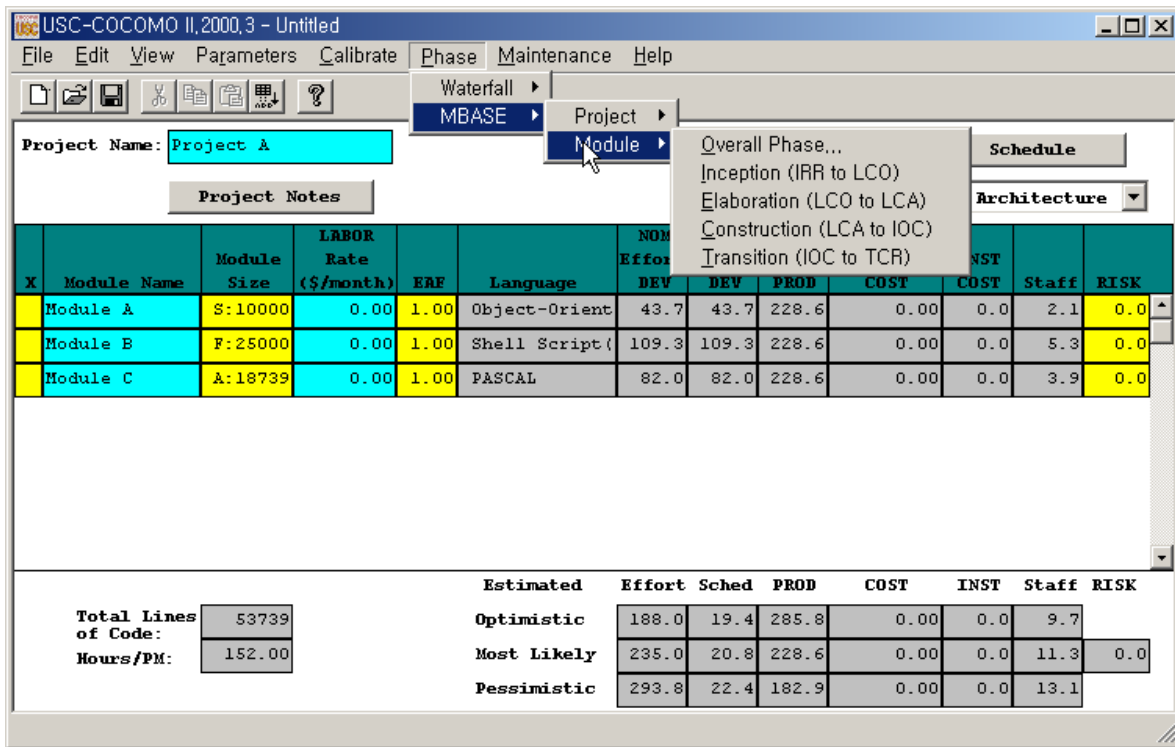
The transition phase distribution allows the user to view the components of this particular phase. When the transition distribution is chosen from the Project Phase distribution menu, the window shown in FIGURE 6-19 is displayed. This window displays the following information: project phase, the total estimated project effort, and the total estimated project schedule. In addition the window displays the estimated effort for the activities of requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-19. To exit from this window click the OK button.

FIGURE 6-19 Transition window for the overall project

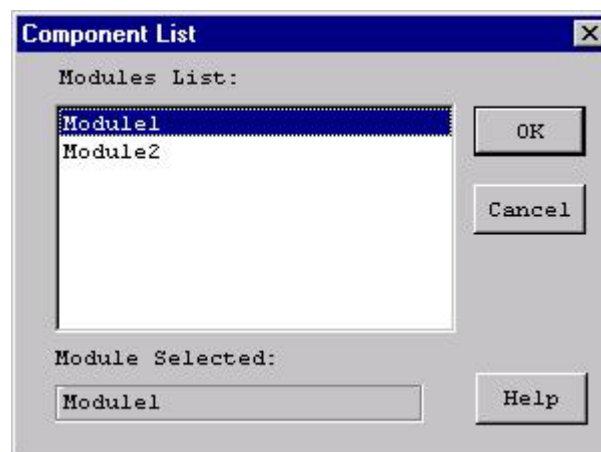


6.2.2 Module Phase Distribution

Five formats for viewing will appear in another menu: overall phase, inception, elaboration, construction, and transition (see FIGURE 6-20). Each of these menu selections will be discussed in sections 6.2.2.1 - 6.2.2.5, respectively. The phase distribution of inception, elaboration, construction and transition are broken down into activities. These activities include: requirements capture, analysis and design, implementation, test, management, environment and deployment. For each of these activities, the percentage of the phase, the estimated effort, the estimated schedule, and the estimated Staff is displayed.

FIGURE 6-20 Phase Distribution Module Sub-menu

In order to view the phase distribution of an entire Module, the user can click on the Module Phase Distribution button under the Phase Distribution menu. When choosing any of the views of phase distribution, you will be confronted with a module selection window (see FIGURE 6-21). At this point, you may choose which module is to be viewed by clicking on the desired module name, which will be highlighted after the click. Click the OK button in order to initiate phase distribution of the chosen module.

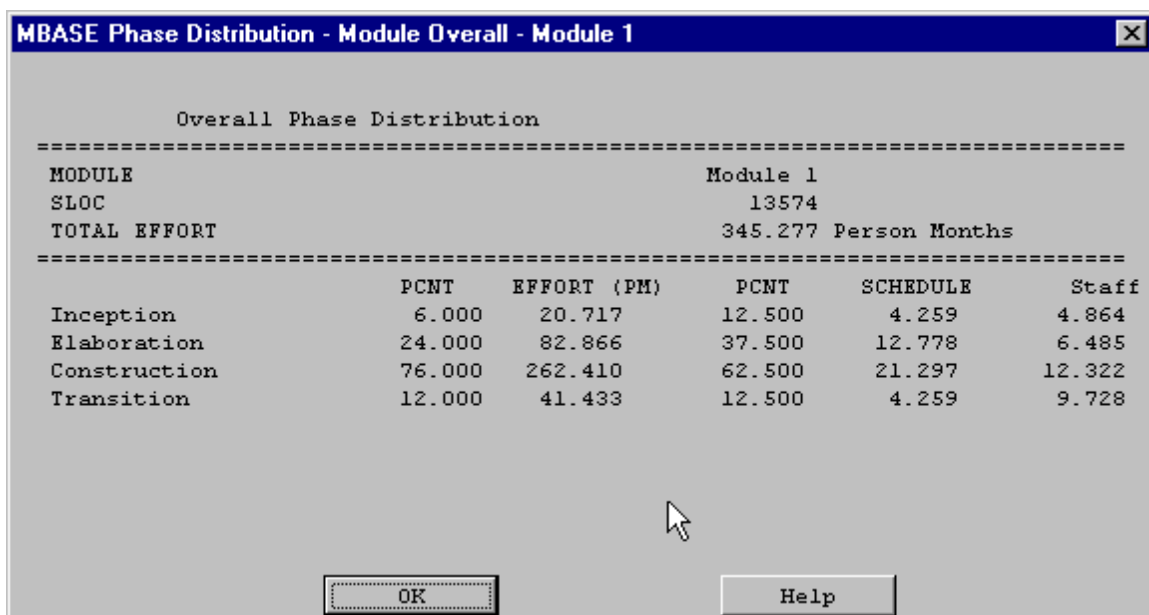
FIGURE 6-21 Module selection window

6.2.2.1 Overall Module Phase Distribution

The overall phase distribution allows the user to view an entire Module's estimated effort, schedule and number of personnel needed for phase completion. Upon clicking on "Overall Phase," a window will be displayed showing the phase breakdown four formats for viewing will appear in another menu: overall phase, plan & requirements, programming, and integration & test (see FIGURE 6-22). To exit from this window click the OK button.

In addition, each phase of the Module's development cycle is represented by the estimated effort, the estimated schedule and the estimated number of personnel needed for phase completion. Again looking at FIGURE 6-22, the information has been separated into columns. The first column displays the phase name. The second column displays the percentage that the corresponding phase takes in the estimated effort. The third column displays the estimated effort for each phase. The fourth column displays the percentage of the estimated schedule that is dedicated to the corresponding phase's completion. The fifth column displays the estimated schedule for phase completion. And the last column displays the estimated number of personnel needed for phase completion (Staff).

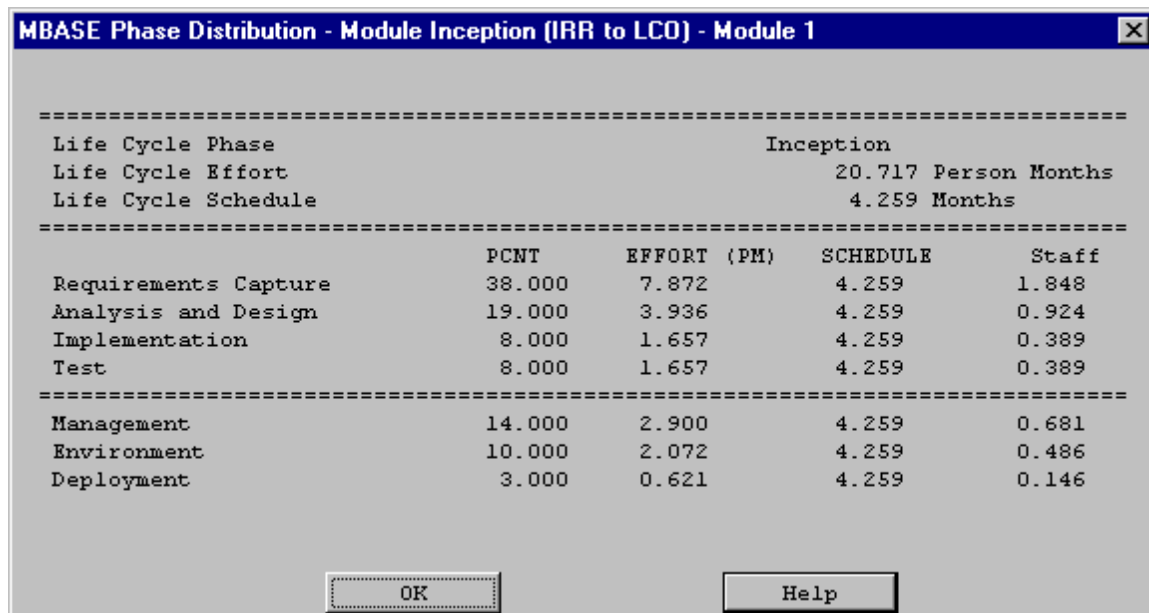
FIGURE 6-9 Phase Distribution window displaying a sample Module's overall phase distribution



6.2.2.2 Inception Module Phase Distribution

The inception phase distribution allows the user to view the components of this particular phase. When the inception distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-23 is displayed. This window displays the following information: Module name, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and FSWP for the activity's completion as shown in FIGURE 6-23. To exit from this window click the OK button.

FIGURE 6-23 Inception Phase window for the overall Module



MBASE Phase Distribution - Module Inception (IRR to LCD) - Module 1				
=====				
Life Cycle Phase	Inception			
Life Cycle Effort	20.717 Person Months			
Life Cycle Schedule	4.259 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	38.000	7.872	4.259	1.848
Analysis and Design	19.000	3.936	4.259	0.924
Implementation	8.000	1.657	4.259	0.389
Test	8.000	1.657	4.259	0.389
=====				
Management	14.000	2.900	4.259	0.681
Environment	10.000	2.072	4.259	0.486
Deployment	3.000	0.621	4.259	0.146
=====				

OK Help

6.2.2.3 Elaboration Module Phase Distribution

The elaboration phase distribution allows the user to view the components of this particular phase. When the elaboration distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-24 is displayed. This window displays the following information: Module name, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activity's requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-24. To exit from this window click the OK button.

FIGURE 6-24 Elaboration Phase window for the overall Module

MBASE Phase Distribution - Module Elaboration (LCO to LCA) - Module 1

=====				
Life Cycle Phase	Elaboration			
Life Cycle Effort	82.866 Person Months			
Life Cycle Schedule	12.778 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	18.000	14.916	12.778	1.167
Analysis and Design	36.000	29.832	12.778	2.335
Implementation	13.000	10.773	12.778	0.843
Test	10.000	8.287	12.778	0.649
=====				
Management	12.000	9.944	12.778	0.778
Environment	8.000	6.629	12.778	0.519
Deployment	3.000	2.486	12.778	0.195
=====				

OK

Help

FIGURE 6-25 Construction window for the overall Module

MBASE Phase Distribution - Module Construction (LCA to IOC) - Module 1

=====				
Life Cycle Phase	Construction			
Life Cycle Effort	262.410 Person Months			
Life Cycle Schedule	21.297 Months			
=====				
	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	8.000	20.993	21.297	0.986
Analysis and Design	16.000	41.986	21.297	1.971
Implementation	34.000	89.219	21.297	4.189
Test	24.000	62.978	21.297	2.957
=====				
Management	10.000	26.241	21.297	1.232
Environment	5.000	13.121	21.297	0.616
Deployment	3.000	7.872	21.297	0.370
=====				

OK

Help

6.2.2.4 Construction Phase Distribution

The construction design phase distribution allows the user to view the components of this particular phase. When the Product Design distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-25 is displayed. This window displays the following information: Module name, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-25. To exit from this window click the OK button.

6.2.2.5 Transition Module Phase Distribution

The transition phase distribution allows the user to view the components of this particular phase. When the transition distribution is chosen from the Module Phase distribution menu, the window shown in FIGURE 6-26 is displayed. This window displays the following information: Module name, the total estimated Module effort, and the total estimated Module schedule. In addition the window displays the activities requirements capture, analysis and design, implementation, test, management, environment and deployment. These activity estimates are accompanied with a percentage of the phase effort that they encompass, the estimated effort, schedule and Staff for the activity's completion as shown in FIGURE 6-26. To exit from this window click the OK button.

FIGURE 6-26 Transition window for the overall Module

MBASE Phase Distribution - Module Transition (IOC to TCR) - Module 1

Life Cycle Phase

Transition

Life Cycle Effort

41.433 Person Months

Life Cycle Schedule

4.259 Months

	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Capture	4.000	1.657	4.259	0.389
Analysis and Design	4.000	1.657	4.259	0.389
Implementation	19.000	7.872	4.259	1.848
Test	24.000	9.944	4.259	2.335
Management	14.000	5.801	4.259	1.362
Environment	5.000	2.072	4.259	0.486
Deployment	30.000	12.430	4.259	2.918

OK

Help

Chapter 7: Maintenance

Maintenance is one of the menu selections in the menu bar that can be accessed by either clicking upon "Maintenance" in the menu bar or pressing Meta+M. Its function is to calculate and display an estimate of the effort and cost necessary to maintain a post development software product for a user-defined number of years (maximum five years). Maintenance encompasses the process of modifying existing operational software while leaving its primary functions intact. This process excludes the following types of activities:

- Major re-design and re-development (more than 50% new code) of a new software product performing substantially the same functions
- Design and development of a sizeable (more than 20% of the source instructions comprising the existing product) interfacing software package which requires relatively little redesigning of the existing product
- Data processing system operations, data entry, and modification of values in the database

Maintenance does include the following types of activities:

- Re-design and re-development of small portions of an existing software product
- Design and development of small interfacing software packages, which require some redesign of the existing software product
- Modification of the software product's code, documentation, or database structure

Maintenance effort and costs are determined by essentially the same cost driver attributes used to determine the software development costs and effort (exceptions are the RELY, SCED and MODP factors which will be discussed in greater detail later in this chapter). The maintenance calculations are heavily based upon the Maintenance change Factor (MCF) and the Maintenance Adjustment Factor (MAF). The MCF is similar to the Annual change Traffic in COCOMO81, except that maintenance periods other than a year can be used (see EQ 7-1).

Maintenance Change Factor (EQ 7-1)

$$MCF = \frac{SizeAdded + SizeModified}{BaseCodeSize}$$

The initial maintenance size is obtained in one to two ways. The first equation in EQ 7-2 is used when the base code size is known and percentage of change to the base code is known. The second equation in EQ 7-2 is used when the fraction of code added or modified to the existing base code during the maintenance period is known.

Initial Maintenance Size(EQ 7-2)

$$\begin{aligned} \{Size\}_M &= \{BaseCodeSize\} \times MCF \times MAF \\ \{Size\}_M &= \{SizeAdded + SizeModified\} \times MAF \end{aligned}$$

As shown in EQ 7-2, the initial maintenance size estimate is adjusted with a Maintenance Adjustment Factor (see EQ 7-3).

Maintenance Adjustment Factor(EQ 7-3)

$$MAF = 1 + \left(\frac{SU}{100} \times UNFM \right)$$

The resulting maintenance effort estimation formula is the same as the COCOMOII Post Architecture development model (see EQ 7-4).

Maintenance Effort(EQ 7-4)

$$PM_M = A \times (Size_M)^B \times \prod_{i=1}^{17} EM_i$$

FIGURE 7-1 Maintenance sub-menu

X	Module Name	Module Size	LABOR Rate (\$/month)	ERF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Module A	S:10000	0.00	1.00	Object-Orient	43.7	43.7	228.6	0.00	0.0	2.1	0.0
	Module B	F:25000	0.00	1.00	Shell Script	109.3	109.3	228.6	0.00	0.0	5.3	0.0
	Module C	A:18739	0.00	1.00	PASCAL	82.0	82.0	228.6	0.00	0.0	3.9	0.0

	Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
Optimistic	188.0	19.4	285.8	0.00	0.0	9.7		
Most Likely	235.0	20.8	228.6	0.00	0.0	11.3	0.0	
Pessimistic	293.8	22.4	182.9	0.00	0.0	13.1		

As stated previously, three cost drivers for maintenance differ from development. Those cost drivers are software reliability (RELY), modern programming practices (MODP) and schedule (SCED). The reason for the change in MODP, RELY is that increased investment in software reliability and use of modern programming practices during software development have a strong positive effect upon the maintenance stage. The SCED attribute is controlled by the number

of years value entered by the user. As a result the SCED driver is no longer editable in the EAF window, but is calculated from the user inputted value for number of years when the maintenance function is engaged. For more information on these cost drivers please refer to the introduction of this manual.

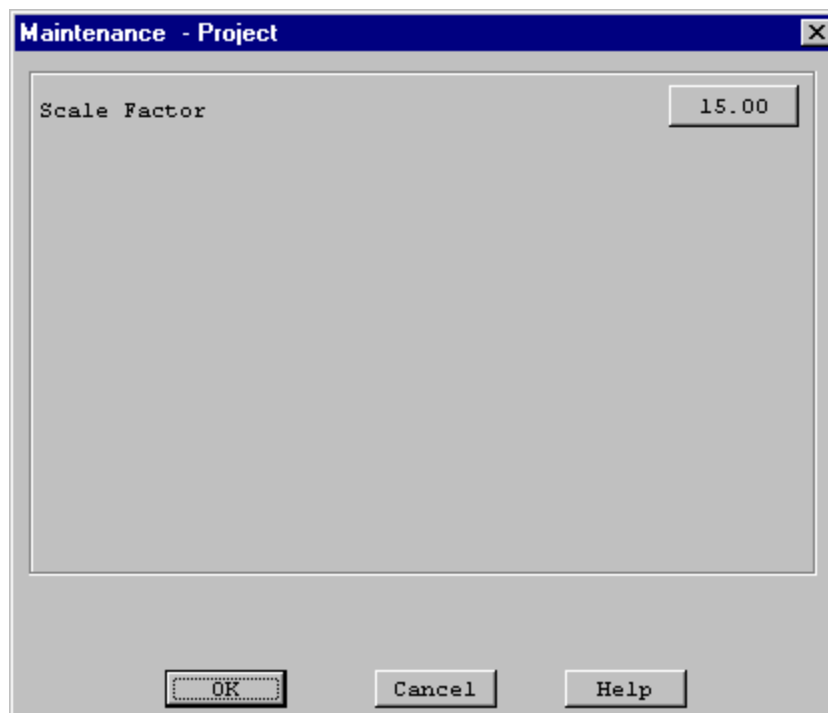
The Maintenance menu option offers sub-menu for either a maintenance effort estimation upon either an entire project or an individual module (see Figure 7-1). These separate options are discussed in section 7.1 and 7.2.

7.1 *Project Maintenance*

In order to view the maintenance estimation calculations for an entire project, the user can click on Project under the Maintenance menu (see Figure 7-1). Upon clicking upon this selection a window will appear displaying the Scale factor button (see Figure 7-2).

The Scale rate can be changed by clicking upon the corresponding button. This action

FIGURE 7-2 **Project Maintenance Dialog Box**



will result in the appearance of an Scale dialog box where the cost driver ratings can be changed as described in the introduction (see Figure 7-3).

Upon completing the adjustment of the cost drivers click the OK button or click the Cancel button to return to the CLEF without viewing maintenance estimations.

After exiting the Scale dialog box, you will be returned to the Project Maintenance Dialog box to continue inputting the editable values.

Click upon the OK button upon completion of editing the displayed fields or click upon the Cancel button if no changes are desired to the default values (if more assistance, the Help button is available to receive on-line assistance).

FIGURE 7-3 Project Maintenance Scale Dialog Box

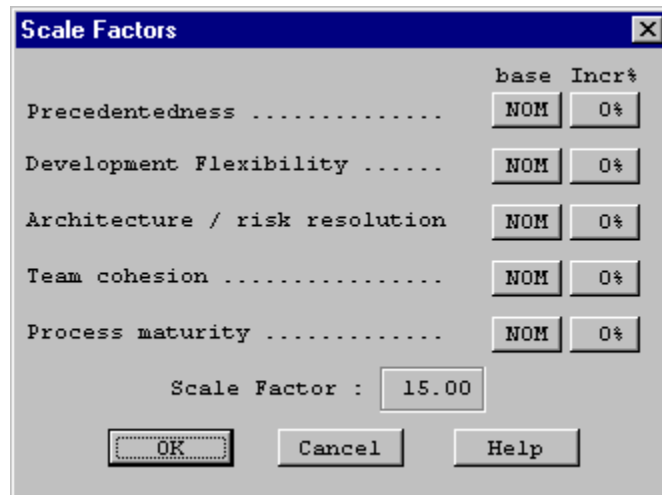
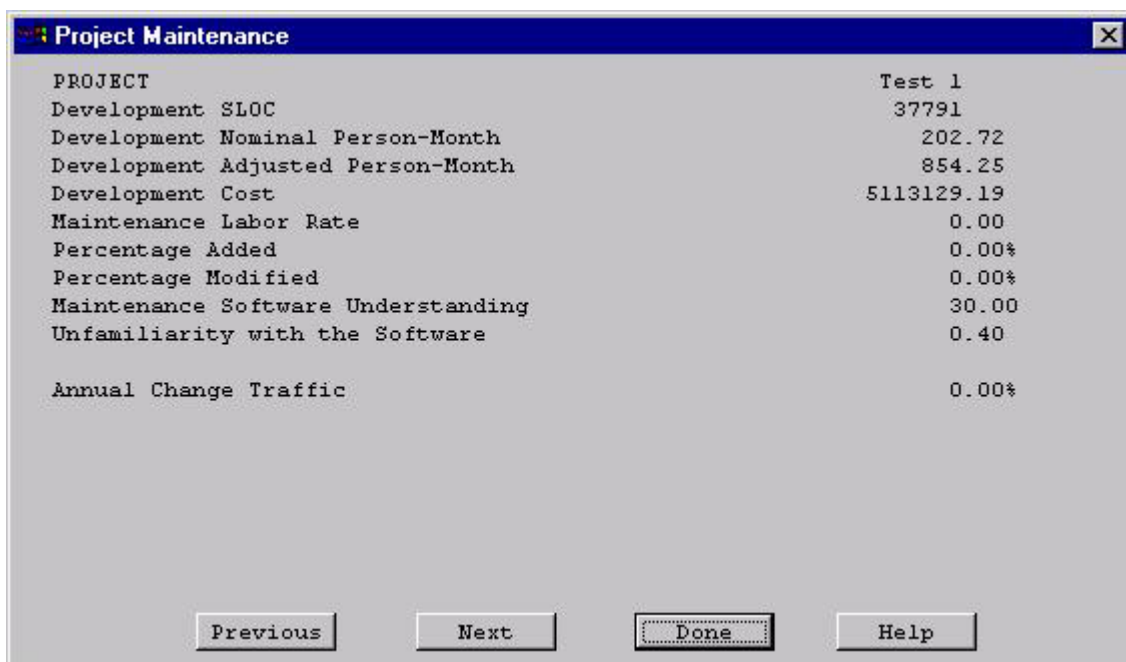


FIGURE 7-4 Project Maintenance window (page 1)



When the OK button is clicked in the Project Maintenance Dialog Box, a window displaying the first of four pages that contains the project name, the current development mode, the total number of source instructions for development of the project (EDSI) that is loaded in the CLEF, the nominal effort of the project, the actual effort of the project, the development cost, the inputted maintenance labor rate, the inputted percent of code added during maintenance per year, the inputted percent of code modified during maintenance per year (see Figure 7-4) and the calculated annual change traffic. The second page of the maintenance window can be seen by clicking upon the Next button. It contains the settings for the 16 cost drivers, SCED is not applicable (see Figure 7-5).

FIGURE 7-5 Project Maintenance window (page 2)

The screenshot shows a window titled "Project Maintenance" with a close button (X) in the top right corner. The window displays a form with several sections of cost drivers, each with a "Nominal" value and a "0%" input field.

Effort Adjustment Factor								1.00		
Product										
+	RELY	DATA	DOCU	CPLX	RUSE	TIME	STOR	++	Platform	+
	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal		Nominal	
	0%	0%	0%	0%	0%	0%	0%		0%	
Personnel										
+	ACAP	AEXP	PCAP	PEXP	LTEX	PCON		+		
	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal			Nominal	
	0%	0%	0%	0%	0%	0%			0%	
Project				User						
+	TOOL	SITE	USR1	USR2				+		
	Nominal	Nominal	Nominal	Nominal					Nominal	
	0%	0%	0%	0%					0%	

At the bottom of the window are four buttons: "Previous", "Next" (highlighted with a dashed border), "Done", and "Help".

The third page of the maintenance window contains the effort and cost estimation for the next N number of years (as defined by the user). With each year is listed the KDSI ($EDSI * 10^3$), the nominal effort for development (PM nom), the actual effort for maintenance (PM maint), the number of full time software personnel necessary to maintain the project for the year (FSWP), the number of instructions that are to be maintained be per personnel($KDSI/FSWP$) and the total cost for maintenance for the year (see Figure 7-6).

The fourth window of the maintenance window contains the cumulative figures for effort and cost for maintenance for the total number of years (see figure 7-7). This first displays the total number of effort estimated for maintenance, then sums the effort of development and maintenance together. It also displays the total cost of maintenance of the project and then displays the summed total cost of development and maintenance for the entire project.

FIGURE 7-6 Project Maintenance window (page 3)

Project Maintenance

WORK LOAD PROJECTION FOR THE NEXT 1 YEARS

Year	KDSI	PM_Nom	PM_Maint	FSWP	KDSI/FSWP	Cost
1	37.79	0.00	0.00	0.00	0.00	0.00

Previous Next Done Help

FIGURE 7-7 Project Maintenance window (page 4)

Project Maintenance

PROJECT Test 1

Development SLOC	37791
Development Nominal Person-Month	202.72
Development Adjusted Person-Month	854.25
Development Cost	5113129.19
Maintenance Labor Rate	0.00
Percentage Added	0.00%
Percentage Modified	0.00%
Maintenance Software Understanding	30.00
Unfamiliarity with the Software	0.40
Annual Change Traffic	0.00%

Previous Next Done Help

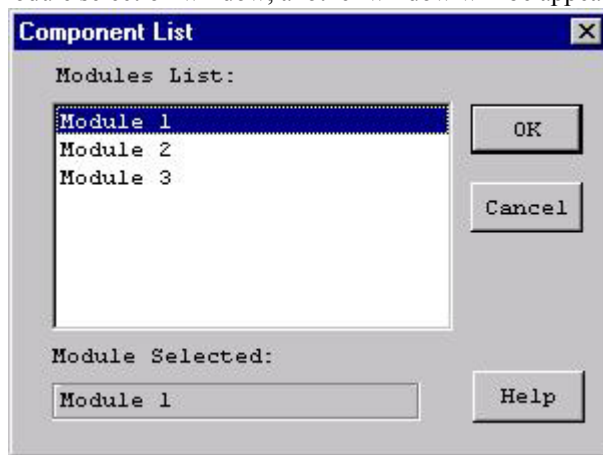
Note - Each individual page can be seen by cycling through the pages pressing either the Previous or Next buttons as needed.

7.2 *Module Maintenance*

In order to view the maintenance estimation calculations for an entire module, the user can click on Module under the Maintenance menu (see Figure 7-1). Upon clicking upon this selection a window will appear displaying the current module names. Choose only one of the modules by highlighting the appropriate module name and then clicking upon OK (see Figure 7-8).

FIGURE 7-8 **Module Selection window**

Upon exiting the module selection window, another window will be appear that displays, the selected module name, an



EAF button, an editable labor rate field, editable number of years of maintenance field, an editable percent of added source instructions field per year of maintenance and an editable percent of modified source instructions field per year of maintenance (see Figure 7-9).

FIGURE 7-9 **Module Maintenance Dialog Box**

The EAF rate can be changed by clicking upon the corresponding button. This action will result in the appearance of an EAF dialog box where the cost driver ratings can be changed as described in the introduction (see Figure 7-10).

FIGURE 7-10 **Module Maintenance EAF Dialog Box**

base + Incr % = rating

Product:	RELY	DATA	DOCU	CPLX	RUSE
base	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%

Platform:	TIME	STOR	PVOL
base	NOM	NOM	NOM
Incr%	0%	0%	0%

Personnel:	ACAP	AEXP	PCAP	PEXP	LTEX	PCON
base	NOM	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%

Project:	TOOL	SITE
base	NOM	NOM
Incr%	0%	0%

User:	USR1	USR2
base	NOM	NOM
Incr%	0%	0%

EAF: 1.00

Upon completing the adjustment of the cost drivers click the OK button or click the Cancel button to return to the CLEF without viewing maintenance estimations.

After exiting the EAF dialog box, you will be returned to the Module Maintenance Dialog box to continue inputting the editable values.

Click upon the OK button upon completion of editing the displayed fields or click upon the Cancel button if no changes are desired to the default values (if more assistance, the Help button is available to receive on-line assistance).

When the OK button is clicked in the Module Maintenance Dialog Box, a window displaying the first of four pages that contains the module name, the current development mode, the total number of source instructions for development of the module (EDSI) that is loaded in the CLEF, the nominal effort of the module, the actual effort of the module, the development cost, the inputted maintenance labor rate, the inputted percent of code added during maintenance per year,

the inputted percent of code modified during maintenance per year (see figure 7-11) and the calculated annual change traffic.

FIGURE 7-11 Module Maintenance window (page 1)

The second page of the maintenance window can be seen by clicking upon the Next button. It contains the settings for

MODULE	Module 1
Development SLOC	13574
Development Nominal Person-Month	72.81
Development Adjusted Person-Month	345.28
Development Cost	1957717.90
Maintenance Labor Rate	0.00
Percentage Added	0.00%
Percentage Modified	0.00%
Maintenance Software Understanding	30.00
Unfamiliarity with the Software	0.40
Annual Change Traffic	0.00%

Buttons: Previous, Next, Done, Help

the 16 cost drivers, SCED is not applicable (see Figure 7-12).

FIGURE 7-12 Module Maintenance window (page 2)

Effort Adjustment Factor: 1.00

Product				Platform			
RELY	DATA	DOCU	CPLX	RUSE	TIME	STOR	PVOL
Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal
0%	0%	0%	0%	0%	0%	0%	0%

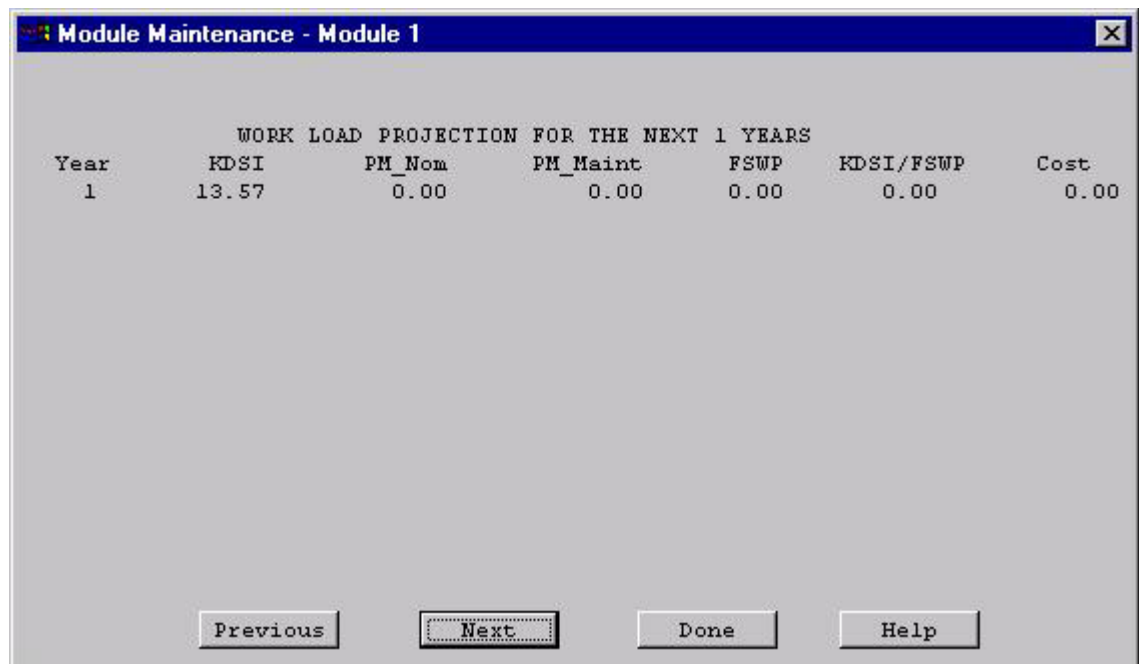
Personnel			
ACAP	AEXP	PCAP	PEXP
Nominal	Nominal	Nominal	Nominal
0%	0%	0%	0%

Project		User	
TOOL	SITE	USR1	USR2
Nominal	Nominal	Nominal	Nominal
0%	0%	0%	0%

Buttons: Previous, Next, Done, Help

The third page of the maintenance window contains the effort and cost estimation for the next N number of years (as defined by the user). With each year is a listed the KDSI ($EDSI * 10^3$), the nominal effort for development (PM nom), the actual effort for maintenance (PM maint), the number of full time software personnel necessary to maintain the module for the year (FSWP), the number of instructions that are to be maintained be per personnel(KDSI/FSWP) and the total cost for maintenance for the year (see Figure 7-13).

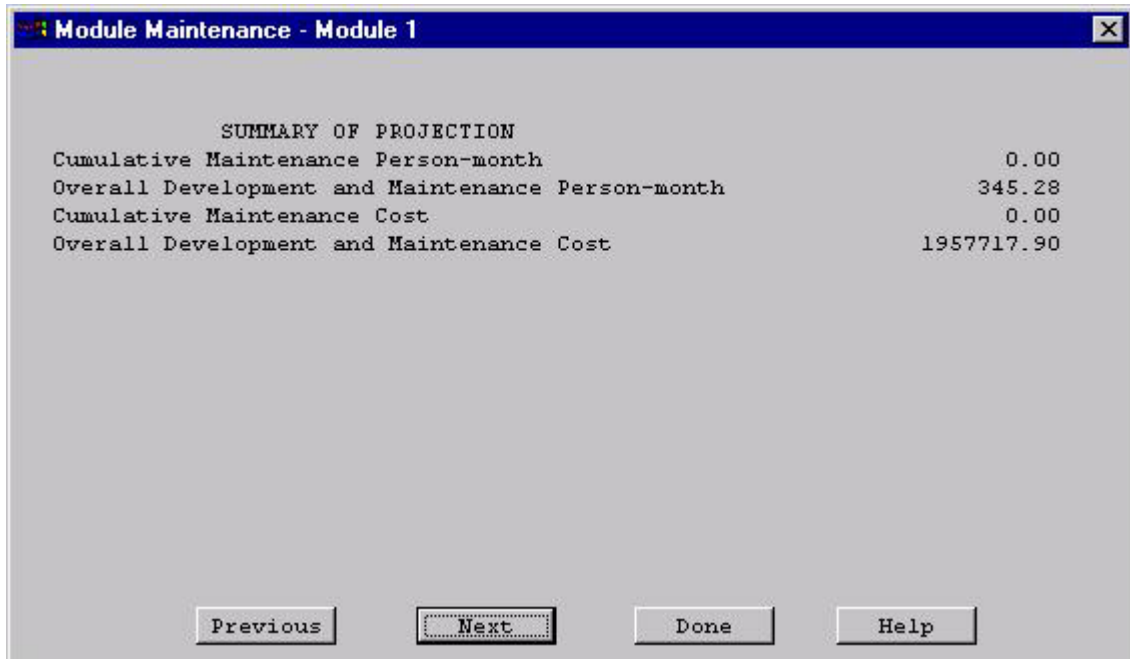
FIGURE 7-13 **Module Maintenance window (page 3)**



The screenshot shows a window titled "Module Maintenance - Module 1". Inside, there is a table titled "WORK LOAD PROJECTION FOR THE NEXT 1 YEARS". The table has 7 columns: Year, KDSI, PM_Nom, PM_Maint, FSWP, KDSI/FSWP, and Cost. The first row of data shows Year 1 with KDSI 13.57, PM_Nom 0.00, PM_Maint 0.00, FSWP 0.00, KDSI/FSWP 0.00, and Cost 0.00. At the bottom of the window are four buttons: "Previous", "Next" (which is highlighted with a dashed border), "Done", and "Help".

Year	KDSI	PM_Nom	PM_Maint	FSWP	KDSI/FSWP	Cost
1	13.57	0.00	0.00	0.00	0.00	0.00

The fourth window of the maintenance window contains the cumulative figures for effort and cost for maintenance for the total number of years (see Figure 7-14). This first displays the total number of effort estimated for maintenance, then sums the effort of development and maintenance together. It also displays the total cost of maintenance of the module and then displays the summed total cost of development and maintenance for the entire module.

FIGURE 7-14 **Module Maintenance window (page 4)**

Note - Each individual page can be seen by cycling through the pages pressing either the Previous or Next buttons as needed.