**SEN PRACTICAL 8**

COCOMO

The Constructive Cost Model (COCOMO) is an algorithmic software cost estimation model developed by Barry

W. Boehm. The model uses a basic regression formula with parameters that are derived from historical project data

and current project characteristics.

COCOMO was first published in Boehm's 1981 book Software Engineering Economics[1] as a model for estimating

effort, cost, and schedule for software projects. It drew on a study of 63 projects at TRW Aerospace where Boehm

was Director of Software Research and Technology. The study examined projects ranging in size from 2,000 to

100,000 lines of code, and programming languages ranging from assembly to PL/I. These projects were based on the

waterfall model of software development which was the prevalent software development process in 1981.

References to this model typically call it COCOMO 81. In 1995 COCOMO II was developed and finally published in

2000 in the book Software Cost Estimation with COCOMO II.

[2] COCOMO II is the successor of COCOMO 81 and

is better suited for estimating modern software development projects. It provides more support for modern software

development processes and an updated project database. The need for the new model came as software development

technology moved from mainframe and overnight batch processing to desktop development, code reusability and the

use of off-the-shelf software components. This article refers to COCOMO 81.

COCOMO consists of a hierarchy of three increasingly detailed and accurate forms. The first level, Basic COCOMO

is good for quick, early, rough order of magnitude estimates of software costs, but its accuracy is limited due to its

lack of factors to account for difference in project attributes (Cost Drivers). Intermediate COCOMO takes these Cost

Drivers into account and Detailed COCOMO additionally accounts for the influence of individual project phases.

Basic COCOMO

Basic COCOMO computes software development effort (and cost) as a function of program size. Program size is

expressed in estimated thousands of source lines of code (SLOC)

COCOMO applies to three classes of software projects:

• Organic projects - "small" teams with "good" experience working with "less than rigid" requirements

• Semi-detached projects - "medium" teams with mixed experience working with a mix of rigid and less than rigid

requirements

• Embedded projects - developed within a set of "tight" constraints. It is also combination of organic and

semi-detached projects.(hardware, software, operational, ...)

The basic COCOMO equations take the form

Effort Applied (E) = ab

(KLOC)b

b [ man-months ]

Development Time (D) = cb

(Effort Applied)d

b [months]

People required (P) = Effort Applied / Development Time [count]

where, KLOC is the estimated number of delivered lines (expressed in thousands ) of code for project.