Unit: III
Lecture: 9
Empirical Estimation Models

An estimation model for computer software **uses empirically derived formulas to predict effort as a function of LOC or FP.**

Estimation based on models allows us to estimate projects ignoring less significant parameters and concentrating on crucial parameters that drive the project estimate. Models are analytic and empirical in nature.

The estimation models are based on the following relationship:

$$E = f(v_i)$$

E → different project estimates like effort, cost, schedule etc

v_i → directly observable parameter like LOC, function points.

COCOMO Model:

COCOMO stands for Constructive Cost Model.

- It was introduced by Barry Boehm.

It is perhaps the best known and most documented of all software cost estimation models.

It is a regression model based on LOC i.e. number of Lines of Code.

It is a procedural cost estimate model for software projects and is often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality.

The key parameters which define the quality of any software products, which are also an outcome of the COCOMO are:

- **Effort:** Amount of labor that will be required to complete a task. It is measured in person-months units.

- **Schedule:** Simply means the amount of time required for the completion of the job, which is of course, proportional to the effort put in. It is measured in the units of time such as weeks, months.

It provided the following three levels of models:

1. **Basic COCOMO:** A single-value model that computes software development cost as a function of estimate of LOC.

2. **Intermediate COCOMO:** This model computes development cost and effort as a function of program size (LOC) and a set of cost drivers.

3. **Detailed COCOMO:** This model computes development effort and cost which incorporates all characteristics of intermediate level with assessment of cost implication on each step of development (analysis, design, testing etc.).

This model may be applied to three classes of software projects as given below:

- **Organic:** Small size project. A simple software project where the development team has good experience of the application.

- **Semi-detached:** An intermediate size project and project is based on rigid and semi-rigid requirements.

- **Embedded:** Such projects are developed under hardware, software and operational constraints. Examples are embedded software, flight control software.

In the COCOMO model, the development effort equation assumes the following form:

$$E = {}_aS^b\ m$$

where a and b are constraints that are determined for each model.

E = effort

S= value of source in LOC

m= multiplier that is determined from a set of 15 cost driver's attributes.

The following are few examples of the above cost drivers:

- Size of the application database.
- Complexity of the project.
- Reliability requirements for the software
- Performance constraints in run-time
- Capability of software engineer
- Schedule constraints.

Barry Boehm suggested that a detailed model would provide a cost estimate to the accuracy of +- 20 % actual value.