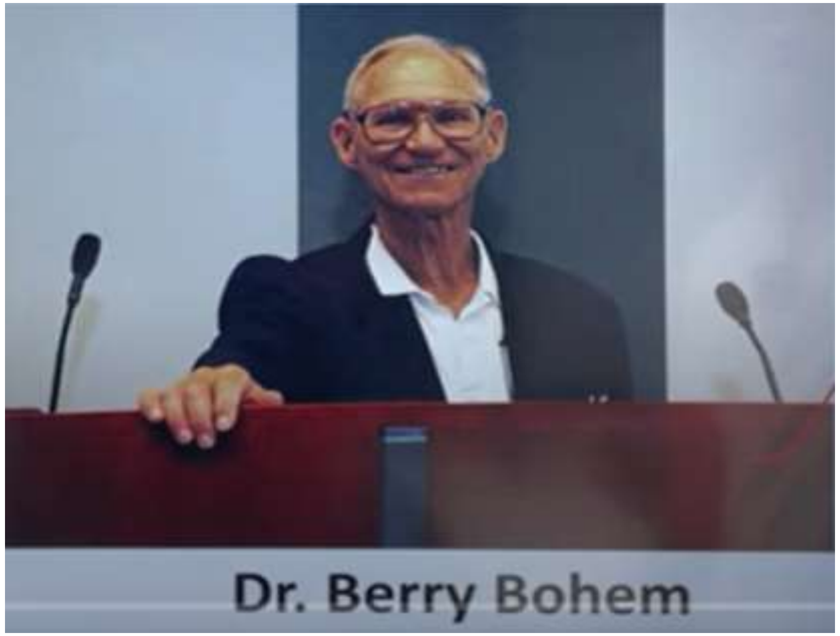
COCOMO

Constructive Cost Model

COCOMO (Constructive Cost Model)

COCOMO is the most widely used estimating technique. It is regression based model introduced by Barry Boehm, he postulated that there are essentially three important classes of software.

- Organic (application)
- Semidetached (utility)
- Embedded (system)



COCOMO (Constructive Cost Model)

- In order to classify a product into any of the three proposed classes, we take into consideration the characteristics of the product as well as those of the development team.

- Data processing and scientific program are considered to be application programs. Compiler, linkers etc are utility programs. Operating system and real time system programs are system programs.

- According to the Brooks the relative level of the product complexity for the three categories of the software product are in the ratio 1:3:9 for application, utility and system programs relatively.

COCOMO (Constructive Cost Model)

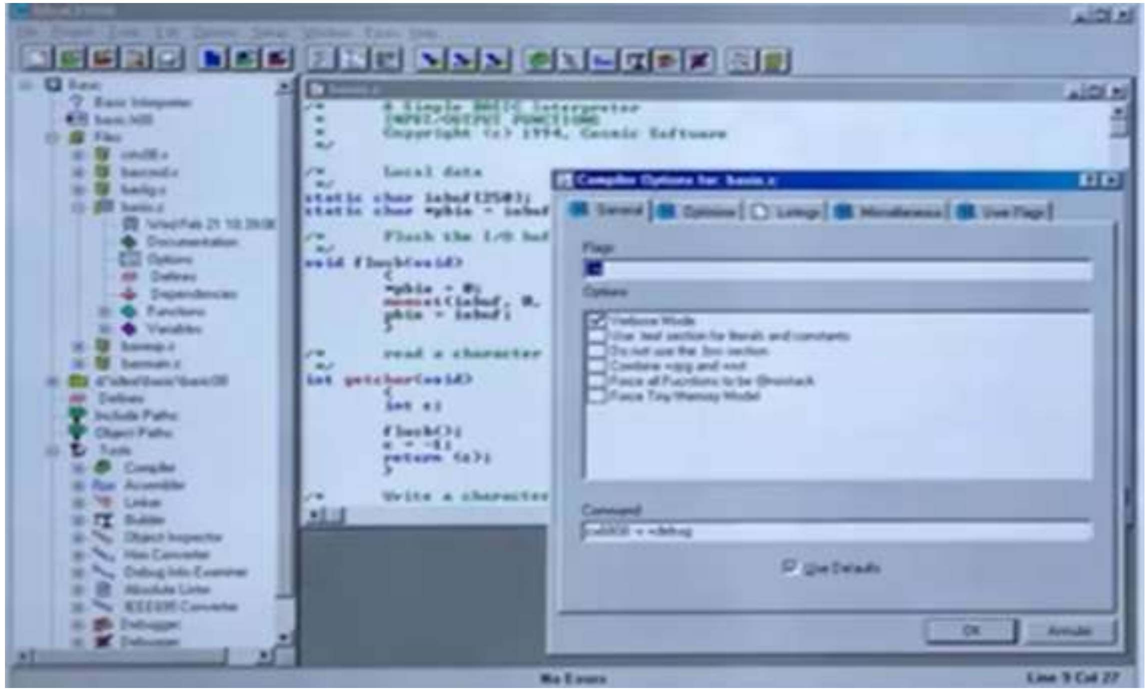
- **Organic:** Relatively small group work in a familiar environment to develop well understood application program, here little innovation is required, constraints and deadlines are few and the development environment is stable.

Example:

College Management System, Library Management System etc.

COCOMO (Constructive Cost Model)

- **Semidetached:** Project teams consist of a mixture of experienced and inexperienced staff. It is of medium size, some innovation is required, constraints and deadlines are moderate and the development environment somewhat fluid.



```
/*      A Simple BASIC Interpreter
 *      INPUT/OUTPUT FUNCTIONS
 *      Copyright (c) 1994, Cosmic Software
 */

/*      Local data
 */
static char iobuf[250];
static char *pbio = iobuf

/*      Flush the I/O buf
 */
void flush(void)
        {
        *pbio = 0;
        memset(iobuf, 0,
        pbio = iobuf;
        }

/*      read a character
 */
int getchar(void)
        {
        int c;

        flush();
        c = -1;
        return (c);
        }

/*      Write a character
```

Compiler Options for: basic.c

General | Options | Listings | Miscellaneous | User Flags

Flags:

Options:

☑ Verbose Mode
□ Use .text section for literals and constants
□ Do not use the .bss section
□ Combine .cpsg and .text
□ Force all Functions to be @nostack
□ Force Tiny Memory Model

Command: cx6808 -v +debug

☑ Use Defaults

No Errors   Line 3 Col 27

COCOMO (Constructive Cost Model)

- **Embedded:** The s/w is strongly coupled to complex h/w, such as air traffic control, ATM's or weapon systems. The project team is large, a great deal of innovation is required, constraints and deadlines are tight and the development environment consist of many complex interfaces, including those with h/w and with customer.

Types of COCOMO (Level of estimation)

- Basic COCOMO
- Intermediate COCOMO
- Detailed COCOMO

Basic COCOMO

- A simple and static cost estimation model that calculates project effort and duration based on the size of the software, measured in thousands of lines of code (KLOC)

- Provide a rough order-of-magnitude estimate of project effort and duration.

- Uses three modes: Organic, Semi-Detached, and Embedded.

Basic COCOMO

Software Project	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semidetached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

$$D_E = a_b\ (\text{KLOC})^{\ bb}$$

$$D_D = c_b\ (D_E)^{\ db}$$

$$\text{Team Size} = D_E\ /\ D_D$$

Software Project	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semidetached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

20KLOC	Organic	Semidetached	Embedded
Effort	55.75 PM	85.95 PM	131 PM
Duration	11.52 M	11.8 M	11.89 M
Team Size	5	7	11

$$D_E = a_b \, (KLOC)^{bb}$$

$$D_D = c_b \, (D_E)^{db}$$

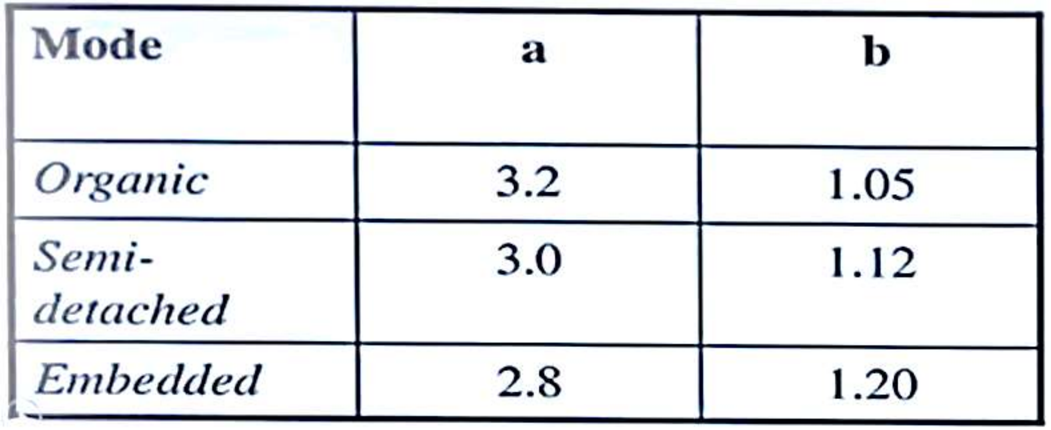
$$\text{Team Size} = D_E / D_D$$

Intermediate COCOMO

- **Definition:** An extension of the Basic COCOMO model that refines effort and duration estimates by considering additional project factors, such as product attributes, hardware constraints, and personnel/team attributes.

- Introduces 15 cost drivers to adjust the estimation, these drivers plays a major role in computation of effort estimation.

- Error Adjustment Factor (EAF) accounts for the influence of cost drivers.



Mode	a	b
Organic	3.2	1.05
Semi-detached	3.0	1.12
Embedded	2.8	1.20

20 KLOC / EAF=1.25	Organic	Semidetached	Embedded
Effort	92.9 PM	107.4 PM	127.4 PM

$D_E = a\ (KLOC)^b$ X EAF(Error Adjustment Factor)

Detailed COCOMO

- **Definition:** The most comprehensive and accurate COCOMO model that divides a software project into multiple components or modules, and accounts for interactions between cost drivers and project phases.

- Estimates effort and duration for each component using the intermediate COCOMO model, then sums them up.

- Considers software reuse hardware constraints and personel/team attributes in estimation

Other Empirical models

$E = 5.2 \text{ X } (KLOC)^{0.91}$ Waltson & Felix Model

$E = 5.2 + 0.73X(KLOC)^{1.16}$ Bailey-Basili Model

$E = 3.2 \text{ X } (KLOC)^{1.05}$ Simple Bohem Model

$E = 5.288 \text{ X } (KLOC)^{1.047}$ Doty Model