

Modelo COCOMO 2.0

- **Nuevo escenario**
 - cambios en tecnología de software
 - nuevos modelos de ciclo de vida
 - nuevas herramientas
 - reingeniería
 - generadores de aplicaciones
 - enfoques orientados a objetos
 - etc

Ingeniería de Software Avanzada

Modelo COCOMO 2.0

- **Características principales**
 - mayor variedad de técnicas y tecnologías
 - uso de diferentes modelos de tamaño según se avanza en el desarrollo y se conoce más del sistema
 - se basa en tres etapas principales de un proceso de desarrollo, reconociendo que es imposible conocer tamaño en LOC en forma temprana en el ciclo de vida
 - dado que es nuevo, no hay datos publicados acerca de su precisión

Ingeniería de Software Avanzada

Modelo COCOMO 2.0

- **Etapas 1**

- normalmente se conoce poco del tamaño probable del producto final, se trabaja con prototipos para resolver aspectos de alto riesgo, por lo que el tamaño se estima en puntos de objeto (u *object points*, que capturan el tamaño en términos de *generadores* de esfuerzo de alto nivel: número de tablas de datos de clientes y de servidores, % de pantallas e informes reusados desde proyectos anteriores)

Ingeniería de Software Avanzada

Modelo COCOMO 2.0

- **Etapas 2**

- se ha decidido continuar con el desarrollo, pero deben explorarse arquitecturas y conceptos de operación alternativos se conoce más que en la etapa anterior, pero no suficiente como para estimar con precisión ... se utilizan puntos de función como estimador de tamaño pues ofrecen una descripción mejor que los puntos de objeto para estimar la funcionalidad capturada en los requerimientos

Ingeniería de Software Avanzada

Modelo COCOMO 2.0

- **Etapas 3**

- ha comenzado el desarrollo y se cuenta con mucha más información, se estima el tamaño en términos de LOC y se utilizan los *cost drivers* de una manera similar al modelo original, incorporando modelos de reuso, nuevos *cost drivers*, nuevos valores de parámetros, tomando en cuenta además la mantención

Ingeniería de Software Avanzada

Cocomo 2: Etapa 1

- **Estimación de esfuerzo:**

$$E = NOP / PROD$$

- **E:** esfuerzo en personas-mes
- **NOP:** *New Object Points*, *NOP*
- **PROD:** razón de productividad

Ingeniería de Software Avanzada

Cocomo 2: Etapa 1

- **Paso 1: Identificación de los tipos de objeto:**
 - Ventanas
 - Informes
 - Componentes de tercera generación
- **Paso 2: Clasificación de cada tipo de objeto en niveles de complejidad.**

Ingeniería de Software Avanzada

Cocomo 2: Etapa 1

Ventanas			
Número de vistas Contenidas	Número y fuentes de tablas de datos		
	Total < 4	Total < 8	Total > 8
<3	simple	simple	media
3-7	simple	media	difícil
>8	media	difícil	difícil

Informes			
Número de vistas Contenidas	Número y fuentes de tablas de datos		
	Total < 4	Total < 8	Total > 8
0-1	simple	simple	media
2-3	simple	media	difícil
>4	media	difícil	difícil

Ingeniería de Software Avanzada

Cocoma 2: Etapa 1

- Paso 3: Calculo de los puntos objetos (*Object Points, OP*).

Tipo de objeto	Complejidad - Peso			
	Simple	Media	Difícil	Subtotal
Ventana	x 1	x 2	x 3	=
Informe	x 2	x 5	x 8	=
Componente de 3GL			x 10	=
Total OP				=

Ingeniería de Software Avanzada

Cocoma 2: Etapa 1

- Paso 4: Calculo de los Nuevos Puntos Objetos (*New Object Points, NOP*).

$$NOP = (OP) / (100 - \%r) / 100$$

,donde %r corresponde a los objetos de proyectos previos que serán reutilizados en el proyecto actual.

Ingeniería de Software Avanzada

Cocomo 2: Etapa 1

- **Paso 5: Determinación de la razón de productividad PROD:**

Experiencia y capacidad de los desarrolladores	Muy Baja	Baja	Nominal	Alta	Muy Alta
Madurez y capacidad de ICASE	Muy Baja	Baja	Nominal	Alta	Muy Alta
PROD	4	7	13	25	50

Ingeniería de Software Avanzada

Cocomo 2: Etapa 1

- **Paso 6: Calculo del esfuerzo.**

$$E = NOP / PROD$$

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Estimación del esfuerzo:**

$$PM_{\text{nominal}} = A * (\text{Tamaño})^B$$

,donde

- » **A:** constante (Temporalmente Etapa 2 A= 2,45; Etapa 3 A= 2,55)
- » **B:** factor de economía y des-economía
- » **Tamaño:** KSLOC

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Factor de escala de economía y des-economía:**

$$B = 0,91 + 0,01 * \sum w_i$$

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

•Factores de escala:

- » **Precedentedness (PREC):** captura la experiencia previa en cuanto al desarrollo de proyectos similares.
- » **Development Flexibility (FLEX):** cuán flexible es el proceso de desarrollo en relación con los requerimientos establecidos.
- » **Architecture/ Risk Resolution (RESL):** pondera el nivel de riesgo asociado al proyecto y el porcentaje de respuesta que es capaz de lograr la organización ante la ocurrencia de algún riesgo.
- » **Team Cohesion (TEAM):** captura el tipo de interacción al interior del proyecto y su impacto sobre él.
- » **Process Maturity (PMAT):** nivel de madurez de la organización en relación con las áreas de prácticas claves (*Key Practices Areas*, KPA) del CMM.

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

Scale Factors (w _i)	Muy Bajo	Bajo	Nominal	High	Very High	Extra High
	5	4	3	2	1	0
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	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 interaction
PMAT	Weighted average of Yes answers to CMM Maturity Questionnaire					

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- **Factor de escala PMAT:**

- **KPA (18):**

Gestión de los requerimientos, Planificación del proyecto, Monitoreo y seguimiento del proyecto, Gestión de subcontratos, SQA, SCM, Organización centrada en el proceso, Definición del proceso, Programas de capacitación, Gestión de software integrada, Coordinación intergrupal, Ingeniería del producto de software, Revisiones por pares, Gestión cuantitativa del proceso, Gestión de calidad del software, prevención de defectos, Gestión del cambio tecnológico, Gestión del cambio del proceso.

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- Cada una de estas áreas es catalogada como:

- » Casi siempre (>90%)
 - » Frecuentemente (60-90%)
 - » Cerca de la mitad de las veces(40-60%)
 - » Ocasionalmente (10-40%)
 - » Rara vez (<10%)
 - » No se aplica
 - » No se sabe

- $PMAT = 5 - \sum (((KPA\% * i) / 100) * 5 / 18) \quad , i=1, \dots, 18$

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Estimación de tamaño:**
 - **LOC:**
 - » Checklist de definición de LOC del SEI
 - **Function Points**
 - » Se contabilizan sólo los UFP, los cuales son transformados a LOC mediante tablas pre-establecidas.

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Estimación de tamaño, Ajuste para reuso**
 - **Entradas:**
 - » **ASLOC:** amount of software to be adapted
 - » **DM:** % design modified
 - » **CM:** % code modified
 - » **IM:** % of modification to the original integration effort required for integrating the reused software

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- **Estimación de tamaño, Ajuste para reuso**
 - **SU:** software understanding
 - **AA:** nivel de evaluación y asimilación requerido para determinar cuando un modulo es apropiado o no para la aplicación y para integrar su descripción a la descripción del producto.
 - **UNFM:** programmer's relative unfamiliarity with the software.

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

	Muy Bajo	Bajo	Nominal	High	Very High
SU (%)	50	40	30	20	10
Structure	Very low cohesion, high coupling, spaghetti code	Moderately low cohesion, high coupling	Reasonably well structured, some weak areas	High cohesion, low coupling	Strong modularity, information hiding in data/control structures
Application Clarity	No match between program and application world views	Some correlation between program and application	Moderate correlation between program and application	Good correlation between program and application	Clear match between program and application world views
Self Descriptiveness	Obscure code; documentation missing, obscure or obsolete	Some code commentary and headers; some useful documentation	Moderate levels of code commentary, headers, documentations	Good code commentary and headers, useful documentation; some weak areas	Self descriptive code, documentation up to date, well organized with design rationale

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

AA Increment	Level of AA Effort
0	None
2	Basic module search and documentation
4	Some module Test and Evaluation, documentation
6	Considerable module Test and Evaluation, documentation
8	Extensive module Test and Evaluation, documentation

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

UNFM Increment	Level of Unfamiliarity
0,0	Completely familiar
0,2	Mostly familiar
0,4	Somewhat familiar
0,6	Considerably familiar
0,8	Mostly unfamiliar
1,0	Completely unfamiliar

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- Estimación de tamaño, Ajuste para reuso

$$ESLOC = [ASLOC(AA + AAF(1 + 0,02(SU)(UNFM)))]/100, \quad AAF \leq 0,5$$

$$ESLOC = [ASLOC(AA + AAF + (SU)(UNFM))]/100, \quad AAF > 0,5$$

$$AAF = 0,4(DM) + 0,3(CM) + 0,3(IM)$$

, donde

» **ESLOC**: líneas de código equivalentes

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- Estimación de tamaño, Ajuste para reingeniería

$$PM_{\text{nominal}} = A * (\text{Tamaño})^B + [ASLOC (AT/100)/ATPROD]$$

ASLOC: amount of software to be adapted

AT: % of code that is re-engineering by automatic translation

ATPROD: productivity for automated translation (temporalmente, 2400)

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

Re-engineering target	AT
Batch processing	96%
Batch with SORT	90%
Batch with DBMS	88%
Batch, SORT, DBMS	82%
Interactive	50%

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- **Ajuste del esfuerzo nominal**

- **Etapas 2:**

$$PM_{ajustado} = PM_{nominal} * \prod EM_i, i=1, \dots, 7$$

- **Etapas 3:**

$$PM_{ajustado} = PM_{nominal} * \prod EM_i, i=1, \dots, 17$$

EM: effort multipliers (cost drivers)

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- **Ajuste del esfuerzo nominal: Etapa 2**

- **PERS (*Personal Capability*)**: captura las potencialidades de analistas y programadores (ACAP y PCAP) y la tasa de rotación del personal (PCON).
- **RCPX (*Product Reliability and Complexity*)**: contempla la influencia de la confiabilidad del software (RELY), el tamaño de la base de datos (DATA), la complejidad del producto (CPLX), y la documentación requerida (DOCU).
- **RUSE (*Required Reuse*)**: esfuerzo adicional requerido para construir componentes que deberán ser reutilizados tanto en el proyecto como por proyectos futuros. (RUSE).
- **PDIF (*Platform Difficulty*)**: refleja es esfuerzo adicional requerido como producto de las restricciones de tiempo de ejecución (TIME), almacenamiento principal (STOR), y volatilidad de la plataforma (PVOL).

Ingeniería de Software Avanzada

Cocomo 2: Etapas 2 y 3

- **PREX (*Personnel Experience*)**: abarca la experiencia en relación con la aplicación (AEXP), la plataforma (PEXP), y el lenguaje y las herramientas para el desarrollo (LTEX).
- **FCIL (*Facilities*)**: captura el uso de herramientas de apoyo al desarrollo del software (TOOL) y si el proceso de desarrollo es realizado en lugares múltiples (SITE).
- **SCED (*Schedule*)**: busca capturar restricciones de tiempo de desarrollo impuestas.

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Ajuste del esfuerzo nominal: Etapa 3**
 - Factores de producto
 - » RELY, DATA, CPLX, RUSE, DOCU
 - Factores de plataforma
 - » TIME, STOR, PVOL
 - Factores de plataforma
 - » ACAP, PCAP, AEXP, PEXP, LTEX, PCON
 - Factores del proyecto
 - » TOOL, SITE, SCED

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Estimación del tiempo de desarrollo**

$$TDEV = [3,67 * (PM_x)^{(0,28 + 0,2 * (B - 1,01))}] * SCED\% / 100$$

- **TDEV:** tiempo de desarrollo desde la etapa de requerimientos hasta la aceptación del producto en relación con los requerimientos.
- **PM_x:** estimación de personas mes excluyendo el multiplicador SCED

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Mantenición:**

$$(\text{Tamaño})_M = [(\text{Tamaño del Código Base}) * \text{MCF}] \text{MAF}$$

- **MCF:** maintenance change factor, % de cambios en el código base (similar al ACT de Cocomo 81)

$$\text{MCF} = (\text{Tamaño agregado} + \text{Tamaño modificado}) / \text{Tamaño del Código Base}$$

Ingeniería de Software Avanzada

Cocoma 2: Etapas 2 y 3

- **Mantenición:**

- **MAF :** maintenance adjustment factors

$$\text{MAF} = 1 + ((\text{SU}/100) * \text{UNFM})$$

- $\text{PM}_M = A * (\text{Tamaño}_M)^B * \prod \text{EM}_{i_i}, i=1, \dots, 17$
- $\text{PM}_M = T_M * \text{FSP}_M$

Ingeniería de Software Avanzada

Cocomo 2

- **Rangos de esfuerzo:**

Modelo	Estimación optimista	Estimación pesimista
Etapas 1	0,5 E	2,0 E
Etapas 2	0,67 E	1,5 E
Etapas 3	0,80 E	1,25 E

Ingeniería de Software Avanzada

Cocomo 2

- **Trabajos futuros:**

- **COCOTS (COnstructive COTS)**
- **COQUALMO (COnstructive Quality Model)**
- **CORADMO (COnstructive Rapid Application Development MOdel)**
- **COSSEMO (COnstructive Staged Schedule and Effort MOdel)**
- **COPROMO (COnstructive PROductivity improvement Model)**

Ingeniería de Software Avanzada