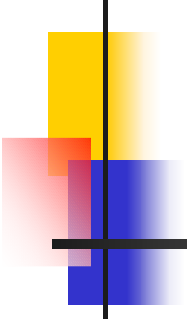


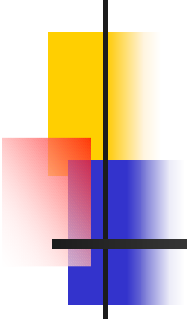
MÉTODOS DE ESTIMACIÓN



MÉTODOS DE ESTIMACIÓN

1. Introducción
2. Precisión y exactitud de las estimaciones
3. Estimación de costes
4. Técnica Delphi
5. Técnicas de descomposición
6. Modelos de coste y esfuerzo
 - Modelos de regresión
 - Modelo de Bailey-Basili
 - Modelo COCOMO
 - Modelo SLIM de Putnam

ANEXO



1. Introducción

Una estimación es la predicción más optimista con una probabilidad distinta de cero de ser cierta.

Una estimación es una predicción que tiene la misma probabilidad de estar por encima o por debajo del valor actual (Tom DeMarco)

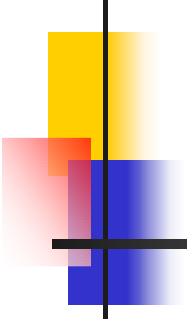
- Los objetivos de la estimación de proyectos son **reducir los costes** e incrementar los niveles de servicio y de **calidad**.
- **Midiendo** determinados aspectos del proceso de software se puede tener una visión de alto nivel de lo que sucederá durante el desarrollo.
 - Las mediciones de procesos anteriores permiten realizar predicciones sobre los actuales.
 - Las mediciones de atributos de proceso en fases iniciales del desarrollo permiten realizar predicciones sobre fases posteriores.
- Las predicciones de proceso conducen la toma de decisiones antes del comienzo del desarrollo, durante el proceso de desarrollo, durante la transición del producto al cliente y a lo largo de la fase de mantenimiento.

1. Introducción

- La figura siguiente muestra algunas de las predicciones que pueden ser útiles a lo largo del ciclo de vida.



Predicciones necesarias durante el ciclo de vida



2. Precisión y exactitud de las estimaciones

La **precisión** hace referencia al número de cifras significativas que tiene una medida.

La **exactitud** se refiere a la cercanía de una medida a su objetivo.

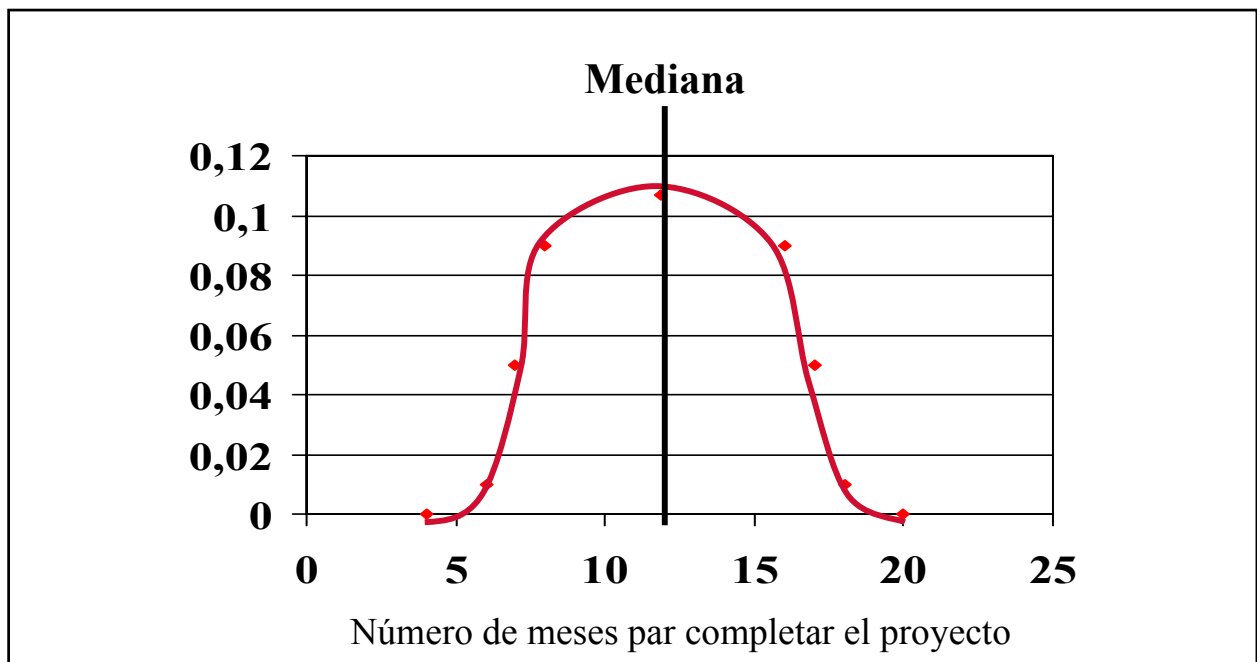
- Una predicción es útil si tiene una exactitud razonable.
- Es preferible expresar las estimaciones como un **rango** que como un número simple.
- *Ejemplo:*

Se desea estimar el tiempo necesario para finalizar un proyecto con un conjunto dado de requisitos y se dispone de datos de tiempo de finalización de **un gran número de proyectos** con los **mismos requisitos**.

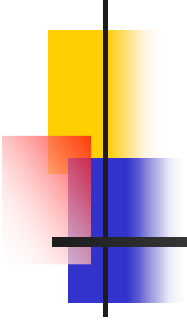
- Se puede representar la función de densidad de probabilidad para el tiempo de finalización t (figura siguiente).
- La probabilidad de que un proyecto de las mismas características se complete en un intervalo de tiempo $[t_1, t_2]$ es el área bajo la curva entre t_1 y t_2 .
- La probabilidad solo puede determinarse para un intervalo determinado. Para un valor de tiempo concreto el área bajo la curva sería cero.

2. Precisión y exactitud de las estimaciones

- Una estimación se define como la **mediana** de la distribución.
- La estimación debe entenderse como el centro de un intervalo que se expresa mediante tres valores:
 - El valor más probable: mediana de la distribución
 - Los límites superior e inferior del valor: intervalos de confianza.



Representación de la función de densidad de probabilidad



2. Precisión y exactitud de las estimaciones

- **Evaluación de la exactitud de la estimación:** comparación de los valores actuales con los valores estimados.

- ***Error relativo de la estimación:***

$$RE = (A - E) / A$$

donde A: valor actual
 E: valor estimado

- ***Error relativo medio:***

$$\overline{RE} = (1/n) \sum_{i=1}^n RE_i$$

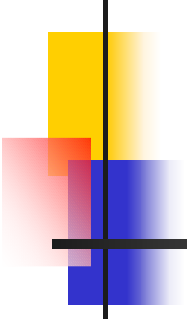
- ***Magnitud del error.*** MRE = valor absoluto de RE

- ***Magnitud media del error relativo:***

$$\overline{MRE} = (1/n) \sum_{i=1}^n MRE_i$$

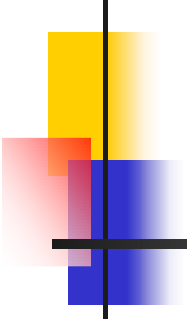
- ***Calidad de la predicción*** para un conjunto de ***n*** proyectos, de los cuales ***k*** es el número de proyectos cuya magnitud media del error relativo es menor o igual a ***q***:

$$PRED(q) = k / n$$



3. Estimación de costes

- La **estimación de costes** implica la realización de predicciones sobre la cantidad más probable de **esfuerzo**, **tiempo** y niveles de **personal** que se requieren para construir un sistema de software.
- Las estimaciones de coste se realizan a lo largo de todo el ciclo de vida.
 - **Las estimaciones preliminares:**
 - Se requieren para hacer una **oferta** o determinar la viabilidad de un proyecto
 - Son las **más difíciles** de hacer y las **menos exactas y las más vinculantes**
 - Se realizan estimaciones “gruesas” del esfuerzo y se divide el esfuerzo total por actividades.
 - **Las estimaciones más detalladas:**
 - Se requieren cuando ha comenzado el proyecto para realizar la planificación
 - El esfuerzo y duración de tareas individuales se compara con los valores estimados.
 - Periódicamente se se realizan **reestimaciones** y se reasignan recursos si fuese necesario.



3. Estimación de costes

FORMAS DE ABORDAR LA ESTIMACIÓN DE COSTES:

- **Opinión de expertos:**
 - Un desarrollador o gestor describe los parámetros del proyecto y los expertos hacen estimaciones basadas en su experiencia.
 - **Técnica Delphi:** permite sistematizar y mejorar la opinión de los expertos consultados.
- **Analogía:**
 - Enfoque más formal que el de la opinión de expertos.
 - Los expertos comparan el proyecto propuesto con uno o más proyectos anteriores intentando encontrar similitudes y diferencias particulares.
- **Base Histórica de proyectos**
- **Técnicas de descomposición:**
 - Las estimaciones se hacen sobre cada componente en que se descompone el software o sobre tareas de bajo nivel en que se descomponen las tareas.
 - Las estimaciones de bajo nivel se combinan para producir una estimación del proyecto completo. Es decir, el coste total del proyecto es el resultado de sumar las estimaciones de todos los componentes en los que se ha dividido el proyecto.

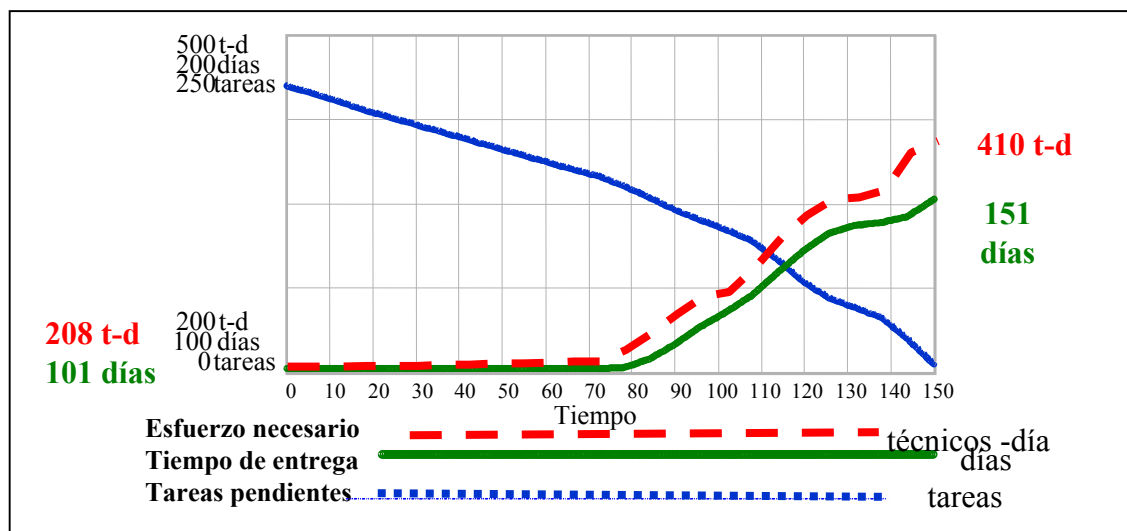
3. Estimación de costes (continuación)

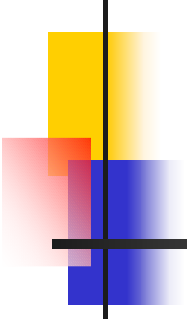
- **Modelos empíricos:**

- Técnicas que identifican los factores clave que contribuyen al esfuerzo y generan una fórmula matemática que relaciona esos factores con el esfuerzo.
- Los modelos se basan normalmente en **experiencias pasadas**.

- **Simuladores:**

- Están basados en modelos dinámicos.
- Permiten simular el comportamiento del proyecto a lo largo del tiempo.

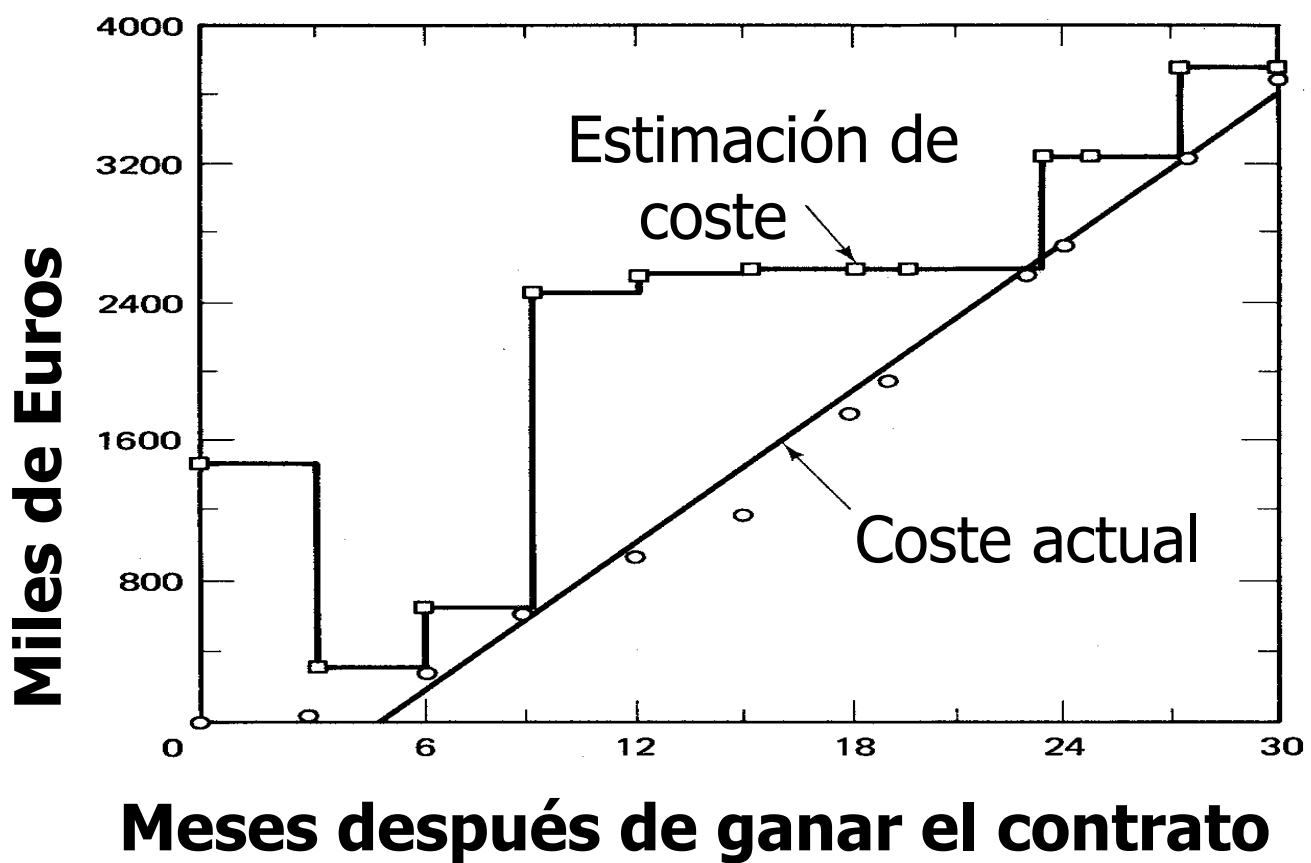


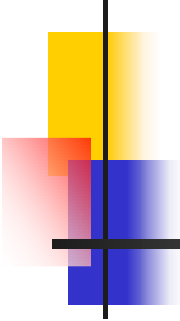


4. Técnica Delphi

- Creada en los años cuarenta en EE.UU. Se comienza a utilizar en el campo del software a partir de 1981 por Barry Boehm. Los pasos a seguir son:
 - Un coordinador proporciona a cada experto una especificación del proyecto propuesto y un impreso para expresar su opinión.
 - Los expertos rellenan el impreso de manera anónima. Pueden hacer preguntas al coordinador pero no entre ellos.
 - El coordinador ofrece a cada experto el valor medio de las opiniones recogidas. Se pide una nueva estimación anónima indicando las razones de las posibles modificaciones.
 - Se repite el proceso de recogida de opiniones hasta llegar a un consenso. No se realizan reuniones en grupo en ningún momento.
- Delphi de banda ancha: refinamiento de la técnica Delphi propuesta por Barry Boehm. Los pasos a seguir son:
 1. El coordinador proporciona a cada experto una especificación del proyecto propuesto y un impreso para expresar su opinión.
 2. *El coordinador reúne a los expertos para que intercambien puntos de vista sobre el proyecto.*
 3. Los expertos rellenan el impreso de manera anónima.
 4. El coordinador ofrece a cada experto el valor medio de las opiniones recogidas. Se pide una nueva estimación anónima, *sin indicar* las razones de las posibles modificaciones.
 5. *El coordinador convoca una reunión para que los expertos discutan las razones de las diferencias entre sus estimaciones.*
 6. Se rellenan anónimamente los impresos y se repite los puntos 4, 5 y 6 hasta llegar a un consenso.
- La ventaja principal de esta técnica es que recoge la opinión experta sobre el proyecto actual basada en experiencias anteriores difíciles de evaluar por otros medios (características especiales del personal, peculiaridades del proyecto,...).
- El principal inconveniente se encuentra en la subjetividad o inexperiencia de las personas elegidas en la consulta.

5. Estimación





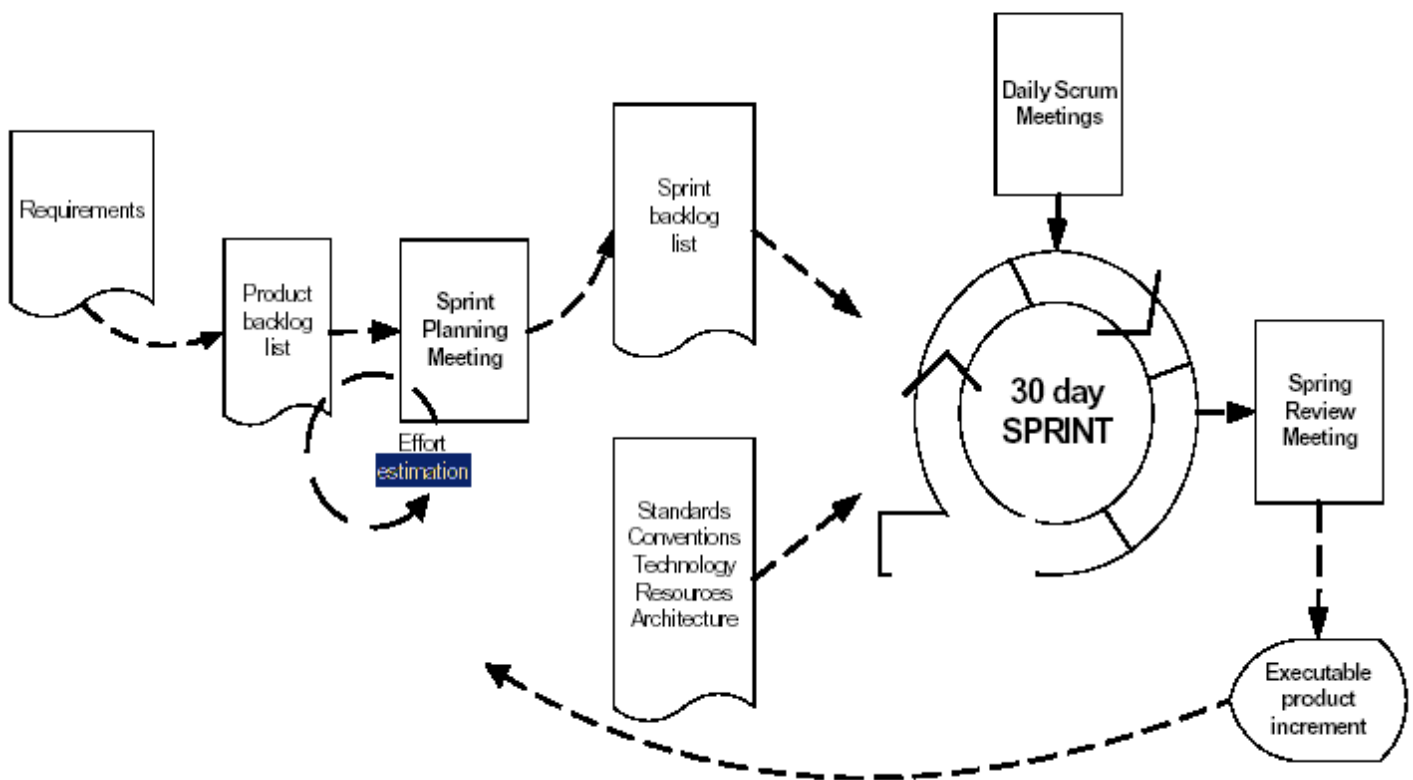
5. Estimación

Los errores más comunes en estimación de costes Software:

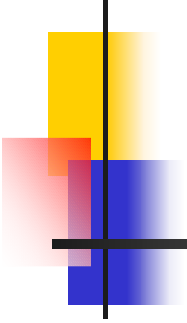
- Desestimar el tiempo y esfuerzo necesario para hacer una buena estimación.
- **Requisitos imprecisos. Requisitos van creciendo.**
- No reconocimiento de que los proyectos actuales serán y son diferentes que los anteriores.
- El tamaño suele ser estimado a la baja.
- Estimaciones forzadas por los recursos disponibles (Si hay que acabar el proyecto en 12 meses y se dispone de 5 técnicos, se estima en esfuerzo como 60 técnicos-día). (NO ACEPTABLE)
- Usar la estimación del precio ganador (NO ACEPTABLE).

Fuente: [Somerville01, pag. 519]

5. Estimación de esfuerzo Agile Modelling



Proceso de desarrollo scrum
<http://www.controlchaos.com/>

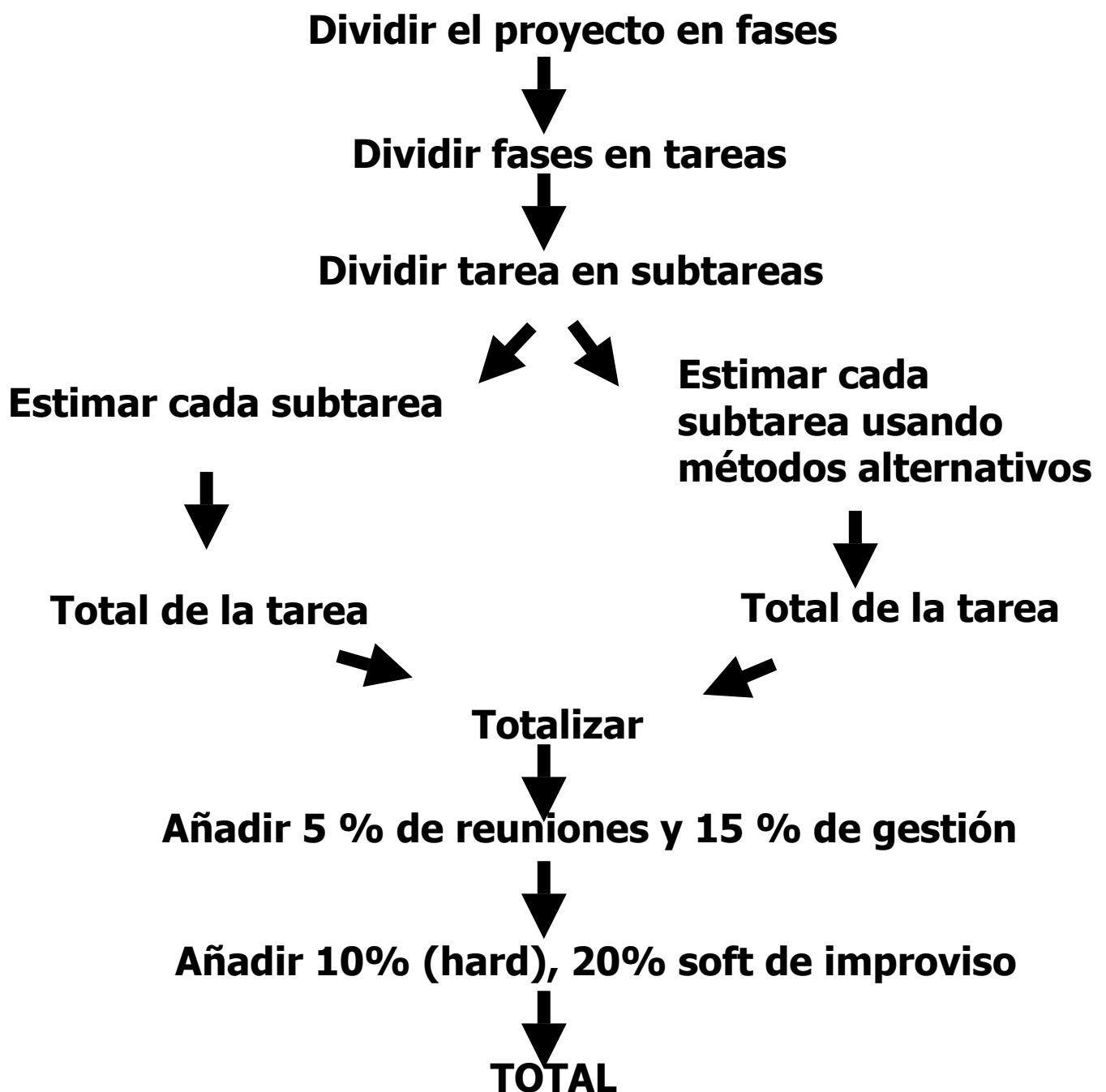


5. Técnicas de descomposición

- La estimación del proyecto completo se calcula mediante la suma de las cantidades parciales (enfoque **abajo-arriba**/bottom-up).
- En la estimación **intervienen los responsables de cada componente** y/o fase del proyecto.
- Lo más adecuado es utilizar las técnicas de descomposición estructurada (EDT/WBS, DFT/WFD). (se verán en práctica)
- Técnicas de descomposición:
 - **Del proyecto** (o por fases)
 - **Del producto** (o por módulos)
 - **Del proyecto y del producto** (por fases y por módulos). Es una combinación de las anteriores.
- Entre las ventajas se encuentran:
 - La posibilidad de que el responsable del componente a estimar participe en dicha estimación.
 - Ayuda a analizar con detalle cada componente.
- Entre los inconvenientes se encuentran:
 - La dificultad para contemplar los costes de:
 - Actividades relacionadas con el proyecto como lectura de código, revisión, reuniones,... (según Barry Boehm supone el **40% del proyecto**).
 - Actividades no relacionadas con el proyecto relacionados con los hábitos de trabajo (según Barry Boehm supone el **30% del proyecto**).
- Las técnicas de descomposición también admiten el criterio de arriba-abajo (top-down).

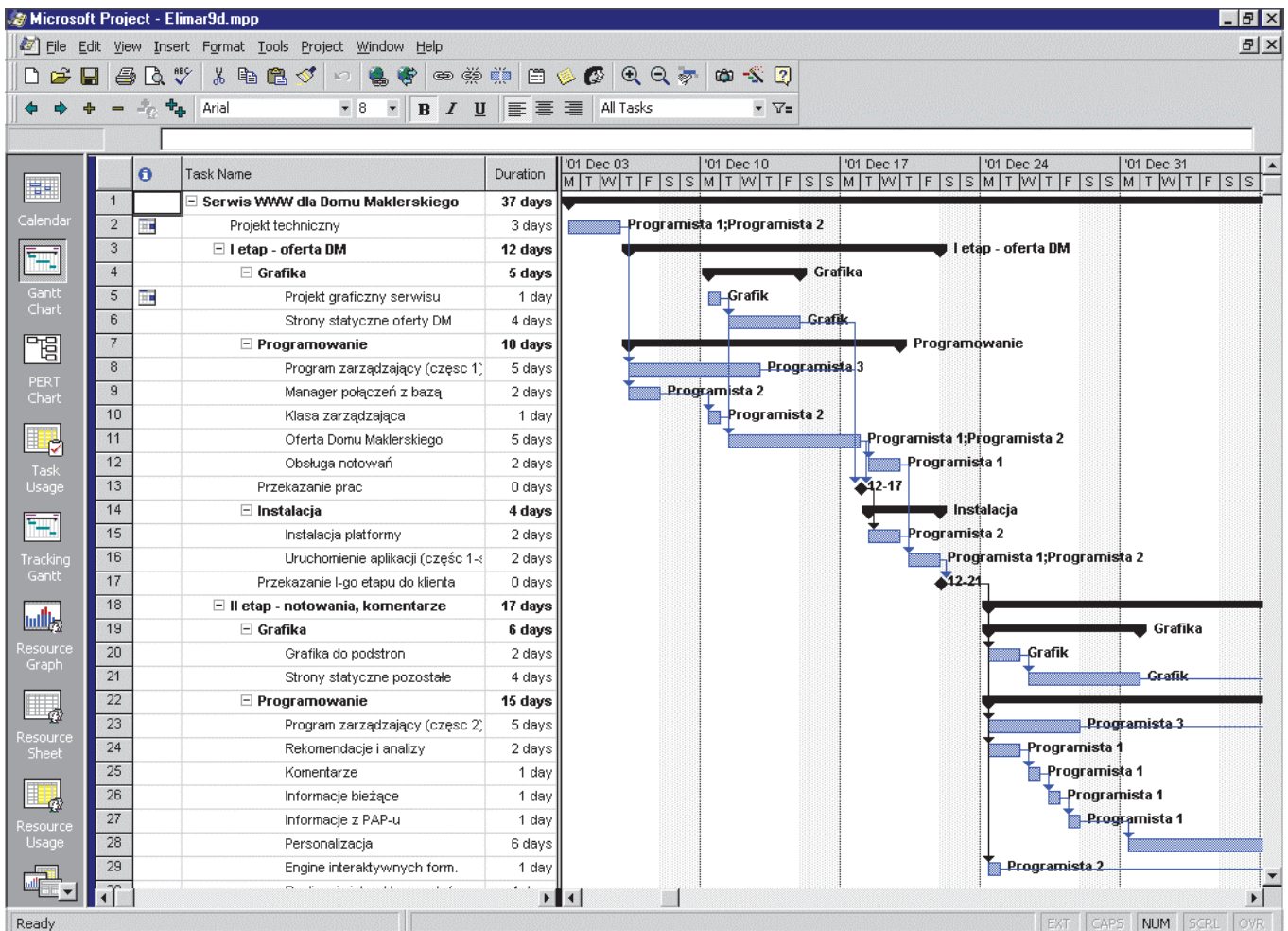


5. Técnicas de descomposición



Herramientas para descomponer y gestionar

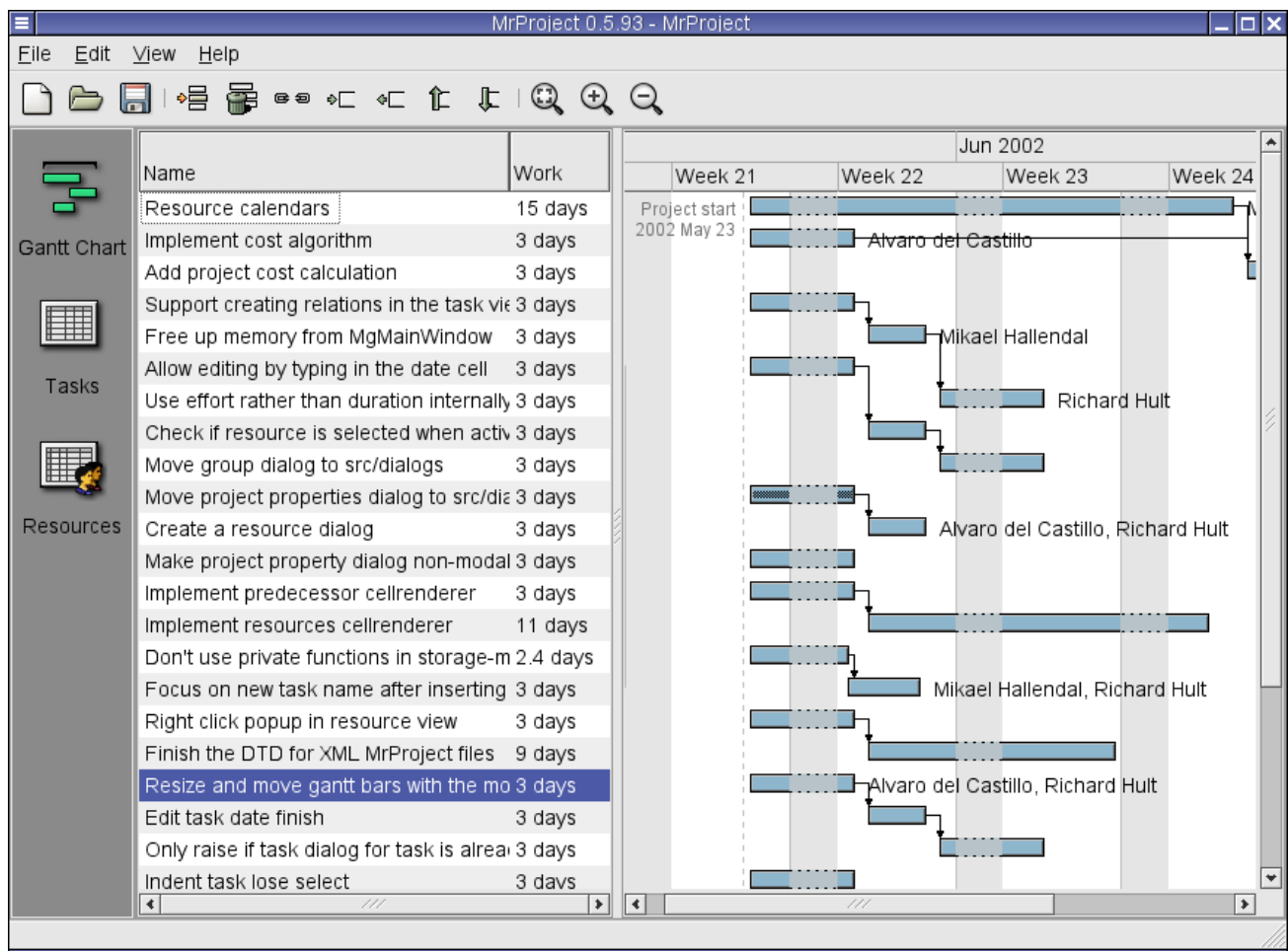
MS Project



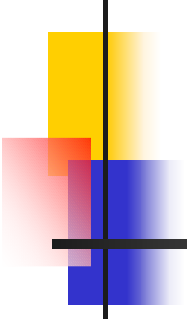
<http://office.microsoft.com>
<ftp://orion.eii.us.es/>

Herramientas para descomponer y gestionar

mrproject



<http://mrproject.codefactory.se/>



6. Modelos de coste y esfuerzo

- Hay dos tipos de modelos que se utilizan para estimar el esfuerzo:
 - **Modelos de coste:**
 - Proporcionan **estimaciones directas** del esfuerzo o la duración del proyecto.
 - Están basados en datos **empíricos** reflejados en factores que contribuyen al coste total.
 - Tienen una entrada primaria (generalmente una medida del **tamaño** del producto) y varios **factores de ajuste** secundarios denominados generalmente *guías de coste*.
 - El modelo **COCOMO** es un modelo empírico de coste.
 - **Modelos restrictivos:**
 - Demuestran la relación con el tiempo entre dos o más parámetros de esfuerzo, duración o nivel de personal.
 - La curva de Rayleigh se usa en modelos de este tipo como base de la relación entre esfuerzo y tiempo.

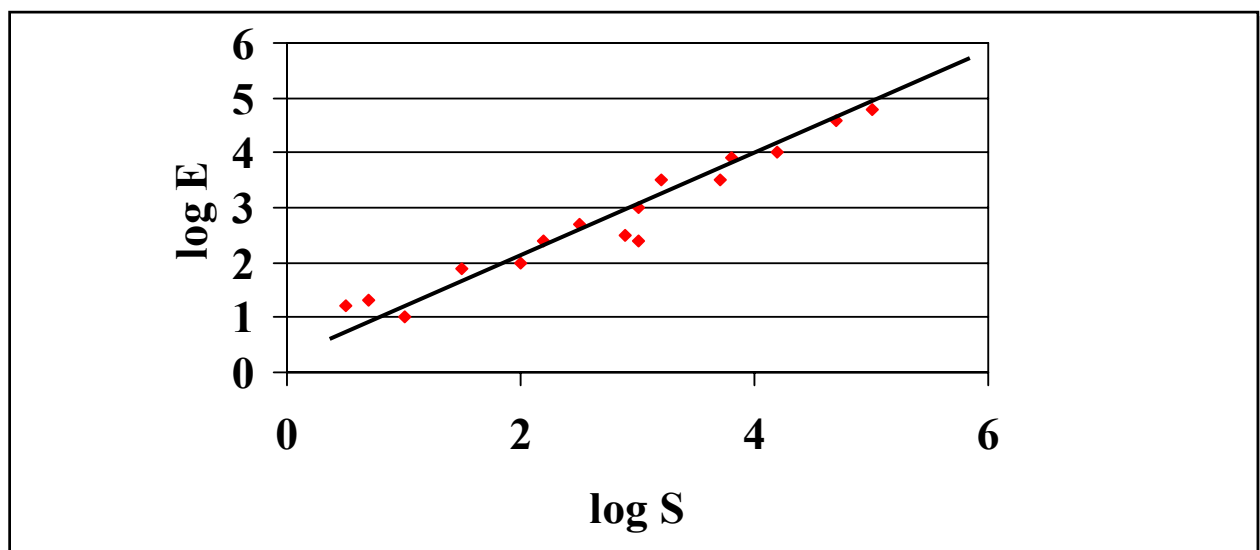
6. Modelos de coste y esfuerzo: modelos de regresión

- Los primeros intentos de construcción de modelos de coste usaban técnicas de regresión.
- Examinando las relaciones entre atributos medidos en muchos proyectos anteriores se construye una ecuación que relacione algunos factores.
- Una vez que se ha definido la ecuación base la estimación puede ajustarse mediante otros factores secundarios.
- Una ecuación de este tipo es la que relaciona el esfuerzo con el tamaño del producto:

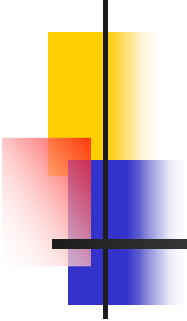
$$E = a S^b$$

- La función exponencial puede transformarse en una función lineal:

$$\log E = \log a + b \log S$$



Representación de la función de densidad de probabilidad



6. Modelos de coste y esfuerzo: modelos de regresión

- El siguiente paso es identificar los factores que causan variación entre el esfuerzo actual y el estimado.
- El análisis de esos factores puede ayudar a identificar parámetros adicionales que se añaden al modelo como ***guías de coste***.
- A cada uno de los factores se les asigna un peso tomando como base el juicio de expertos y datos empíricos.
- Los factores ponderados se aplican a la ecuación del esfuerzo de la forma:

$$E = (a S^b) F$$

donde F es el ***factor de ajuste del esfuerzo*** calculado como el producto del valor de las *guías de coste*.

- El cálculo de F sólo es válido cuando los factores individuales son independientes.
- Dos modelos que utilizan este enfoque son:
 - El modelo COCOMO original en sus versiones intermedia y avanzada.
 - El modelo de Bailey-Basili.
- **LOS MODELOS DE ESTIMACIÓN EMPÍRICOS SE BASAN EN SU MAYORÍA EN ECUACIONES EXPONENCIALES: La relación entre tamaño y esfuerzo NO es lineal, es exponencial.**



6. Modelos de coste y esfuerzo: modelo COCOMO

El modelo COCOMO (COConstructive COst MOdel) fue propuesto por Boehm (1981) y revisado posteriormente dando lugar a la versión COCOMO 2.0

Modelo COCOMO original

- Lo constituye tres modelos:
 - **Básico**: Calcula el esfuerzo en función del tamaño estimado (LDC).
 - **Intermedio**: Calcula el esfuerzo en función del tamaño estimado y de "guías de coste". Las guías de coste evalúan un conjunto de atributos del producto, del hard, del personal y del proyecto.
 - **Avanzado**: Modificación del modelo intermedio para considerar el impacto de las guías de coste en cada fase.
- Definidos para tres tipos de proyectos:
 - **Modo orgánico**: proyectos pequeños y medianos, mucha experiencia, pocas restricciones, realizado por equipos pequeños, entorno familiar.
 - **Modo semiacoplado**: proyectos intermedios en tamaño y complejidad, varios niveles de experiencia.
 - **Modo empotrado**: proyectos complejos y muy restrictivos. Proyectos innovadores. Desarrollados dentro de un conjunto estricto de hard, de soft y restricciones operativas.



6. Modelos de coste y esfuerzo: modelo COCOMO

Modelo COCOMO original

- Modelo ***Básico***:

$$E = a (KDSI)^b$$

DSI: Delivered Source Instructions

Se utiliza en las primeras estimaciones.

- Modelo **Intermedio**:

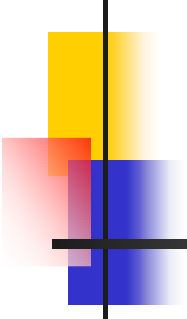
$$E = a (KDSI)^b F$$

F: factor de ajuste: el factor de ajuste se calcula considerando **15 factores** o “guías de coste” que se agrupan en cuatro categorías:

- Atributos del producto (3)
- Atributos computacionales (4)
- Atributos de personal (5)
- Atributos del proyecto (3)

Cada factor se sitúa en una escala de seis valores que va de muy bajo a extra alto. A los valores bajos se le asocia un multiplicador menor que 1, a los medios 1 y a los valores altos mayor que 1.

<i>Proyecto de software</i>	<i>a</i>	<i>b</i>
Orgánico	2,4	1,05
Semiacoplado	3,0	1,12
Empotrado	3,6	1,20



6. Modelos de coste y esfuerzo: modelo COCOMO

Modelo COCOMO original (continuación)

- Para estimar la duración se usa un modelo de restricción que predice el tiempo a partir del esfuerzo:

$$D = a (E)^b$$

D : duración en meses

E : esfuerzo en **técnicos-mes**

- a y b dependen del modelo de desarrollo, según:

<i>Proyecto de software</i>	<i>a</i>	<i>b</i>
Orgánico	2,5	0.38
Semiacoplado	2,5	0.35
Empotrado	2,5	0.32

- La ecuación proporciona la estimación óptima de la duración del proyecto para un esfuerzo determinado.
- Una guía de coste valora el efecto de reducir o incrementar artificialmente el proyecto respecto al valor óptimo.
- **En COCOMO se asume que:**
 - Las LDC excluye comentarios y utilidades no modificadas.
 - Las fases consideradas son de diseño a pruebas. **Asume que las especificaciones no se modifican en dichas fases.**
 - Un técnico-mes equivale a **152 horas** de trabajo.

6. Ecuaciones del modelo COCOMO

<i>Modo</i>	<i>Esfuerzo</i>	<i>Tiempo</i>
Organico	$E = 2.4 * (KDSI)^{1.05}$	$TDEV = 2.5 * (E)^{0.38}$
Semiacoplado	$E = 3.0 * (KDSI)^{1.12}$	$TDEV = 2.5 * (E)^{0.35}$
Empotrado	$E = 3.6 * (KDSI)^{1.20}$	$TDEV = 2.5 * (E)^{0.32}$

<i>Modo</i>	<i>Esfuerzo</i>	<i>Tiempo</i>
Organico	$E = EAF * 3.2 * (KDSI)^{1.05}$	$TDEV = 2.5 * (E)^{0.38}$
Semi-Acoplado	$E = EAF * 3.0 * (KDSI)^{1.12}$	$TDEV = 2.5 * (E)^{0.35}$
Empotrado	$E = EAF * 2.8 * (KDSI)^{1.20}$	$TDEV = 2.5 * (E)^{0.32}$

6. Modelos de coste y esfuerzo: modelo de Bailey-Basili

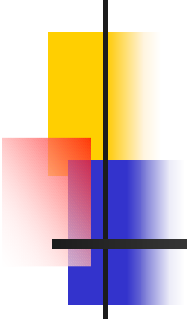
- Bailey y Basili (81) sugirieron una técnica para obtener un modelo de coste a partir de sus propios datos.
- La ecuación del esfuerzo obtenida a partir de **18 grandes proyectos** es:

$$E = 5.5 + 0.73 S^{1.16}$$

- La ecuación se ajusta mediante un factor de ajuste del esfuerzo calculado a partir de los atributos de la tabla siguiente.
 - A cada entrada en la tabla se le da una puntuación de 0 a 5.
 - Los valores obtenidos se usan para ajustar la ecuación:

$$\text{Ajuste del esfuerzo} = a \text{ METH} + b \text{ CPLX} + c \text{ EXP} + d$$

<i>Metodología (METH)</i>	<i>Complejidad acumulada (CPLX)</i>	<i>Experiencia acumulada (EXP)</i>
Diagramas de árboles	Complejidad de la interfaz de usuario	Cualificación del programador
Diseño top-down	Complejidad de la aplicación	Experiencia del programador con la máquina.
Documentación formal	Complejidad del flujo de programa	Experiencia del programador en el lenguaje
Equipos con programador jefe	Complejidad de comunicación interna	Experiencia del programador en la aplicación
Entrenamiento formal	Complejidad de la base de datos	Experiencia del equipo
Formalismos de diseño	Complejidad de la comunicación externa	
Lectura de código	Cambios en el diseño solicitados por el usuario	
Carpetas de desarrollo de unidad		
Planes de prueba formales		



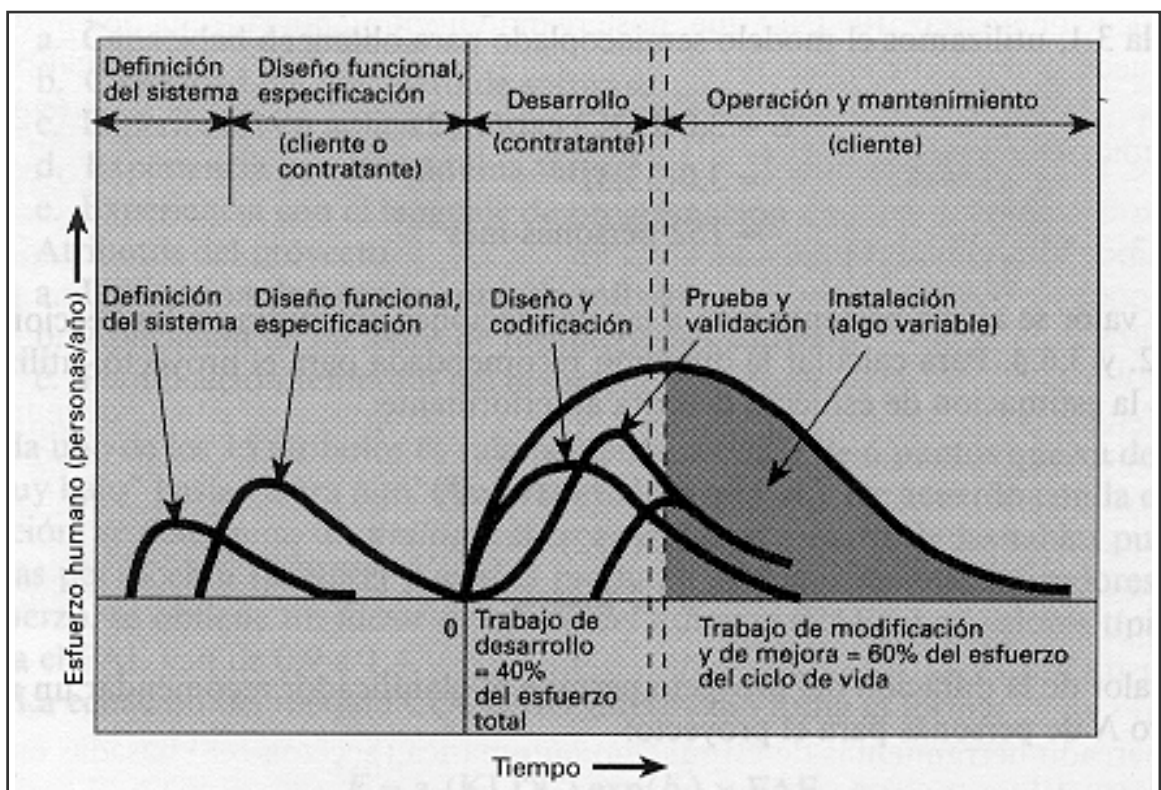
6. Modelos de coste y esfuerzo: Otros modelos

Watson-Felix	$E = 5,2 \times (\text{KLDC})^{0.91}$
<i>Doty</i>	$E = 5,288 \times (\text{KLDC})^{1.047}$
<i>Albretch y Gaffney</i>	$E = -13.39 + 0.0545 \text{ PF}$
<i>Kemerer</i>	$E = 60.62 \times 7.728 \times 10^{-8} \text{ PF}^3$

Fuente: [Pressman 2001, pag 90]

6. Modelos de coste y esfuerzo: modelo SLIM

- Putnam desarrolló un modelo de estimación del esfuerzo total y del tiempo de finalización **para proyectos muy grandes**. Las ecuaciones básicas se pueden ajustar para pequeños proyectos.
- El modelo asume que el esfuerzo para proyectos de desarrollo de software se distribuye de forma similar a una colección de curvas de **Rayleigh**, una para cada actividad del desarrollo.



Curvas de Rayleigh

6. Modelos de coste y esfuerzo: modelo SLIM

- A partir de la fórmula básica de la curva de Rayleigh, Putnam usó observaciones empíricas sobre la productividad para obtener su "ecuación de software" :

$$\textbf{Tamaño} = C K^{1/3} t_d^{4/3}$$

C: factor de tecnología

- 2000 entornos de desarrollo pobres
- 8000 entornos de desarrollo buenos
- 11000 entornos de desarrollo excelentes

K: esfuerzo total medido en **técnicos-año**

t_d : tiempo de finalización del proyecto en años.

La ecuación permite valorar el efecto de modificar el tiempo de entrega y el esfuerzo total necesario para completar el proyecto.

- Putnam propone otra ecuación para estimar el tiempo o la duración:

$$D_o = K / t_d^3$$

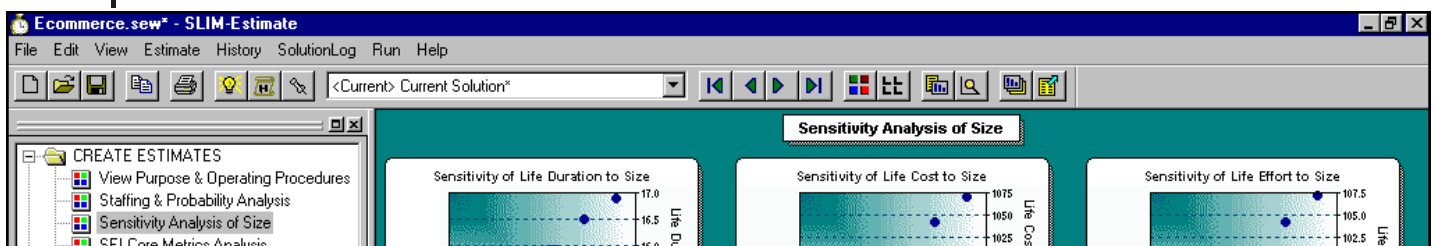
D_o : Constante denominada *aceleración de mano de obra*

- Combinando las ecuaciones anteriores se obtiene:

$$K = (S/C)^{9/7} D_o^{4/7}$$

- El modelo SLIM usa curvas diferentes para distintas fases del ciclo de vida.

Herramientas para estimar



SOLUTION PANEL <Current Solution>

	C&T	Life Cycle	
Duration	12.7	15.9	months
Effort	83	101	MM
Cost	833	1011	\$ (K)
Peak Staff	8.9	8.9	people
MTTD	1.2	3.0	Days
Start Date	02/08/2001	01/01/2001	

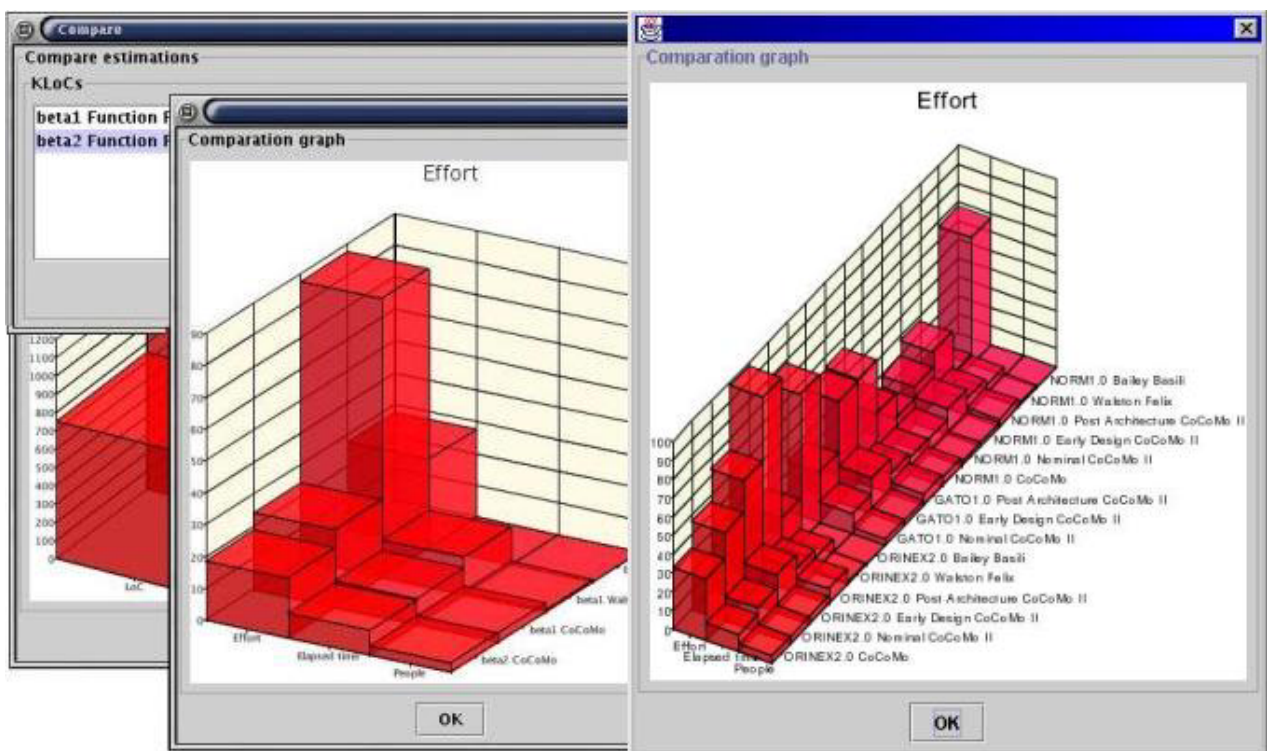
PI=15.1 MBI=1.9 Eff Obj=274



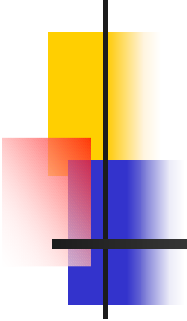
http://www.qsma.com/content_slim_estimate.html

Herramientas para estimar

Effort estimator



<http://qb0x.net/projects/EffortEstimator/>



BIBLIOGRAFÍA

Boehm, B.W., *Software Engineering Economics*, Prentice Hall, Englewood Cliffs, NJ, 1981.

Burnett, K., *The project management paradigm*, Springer-Verlag, 1998.

McConnell, S., *Desarrollo y gestión de proyectos informáticos*, Mc Graw Hill 1997.

Pressman, R.S., *Ingeniería del Software. Un enfoque práctico*, Mc Graw Hill, 2001.

Sommerville, I. *Software Engineering. 6th edition*. Addison Wesley, 2001.