

Proceso Personal para el Desarrollo de Software

Estimaciones, Método PROBE



Universidad Carlos III De Madrid
Principios de Ingeniería Informática
María Isabel Sánchez Segura
José Arturo Mora-Soto
Juan Carlos Alonso Durán

Temario

- Planificación
- Por qué estimar?
- El método PROBE

Planificación

- Guía y organiza el trabajo.
 - Qué debe hacerse?
 - Cuándo debe hacerse?
 - Quién debe hacerlo?
- Facilita el seguimiento.
- Asegura que las tareas clave no sean pasadas por alto.
- Proporciona la base para convenir en el alcance del trabajo, cronogramas y recursos.
- Permite incurrir en compromisos realizables.
 - La mala planificación es una de las razones de fracaso de proyectos.

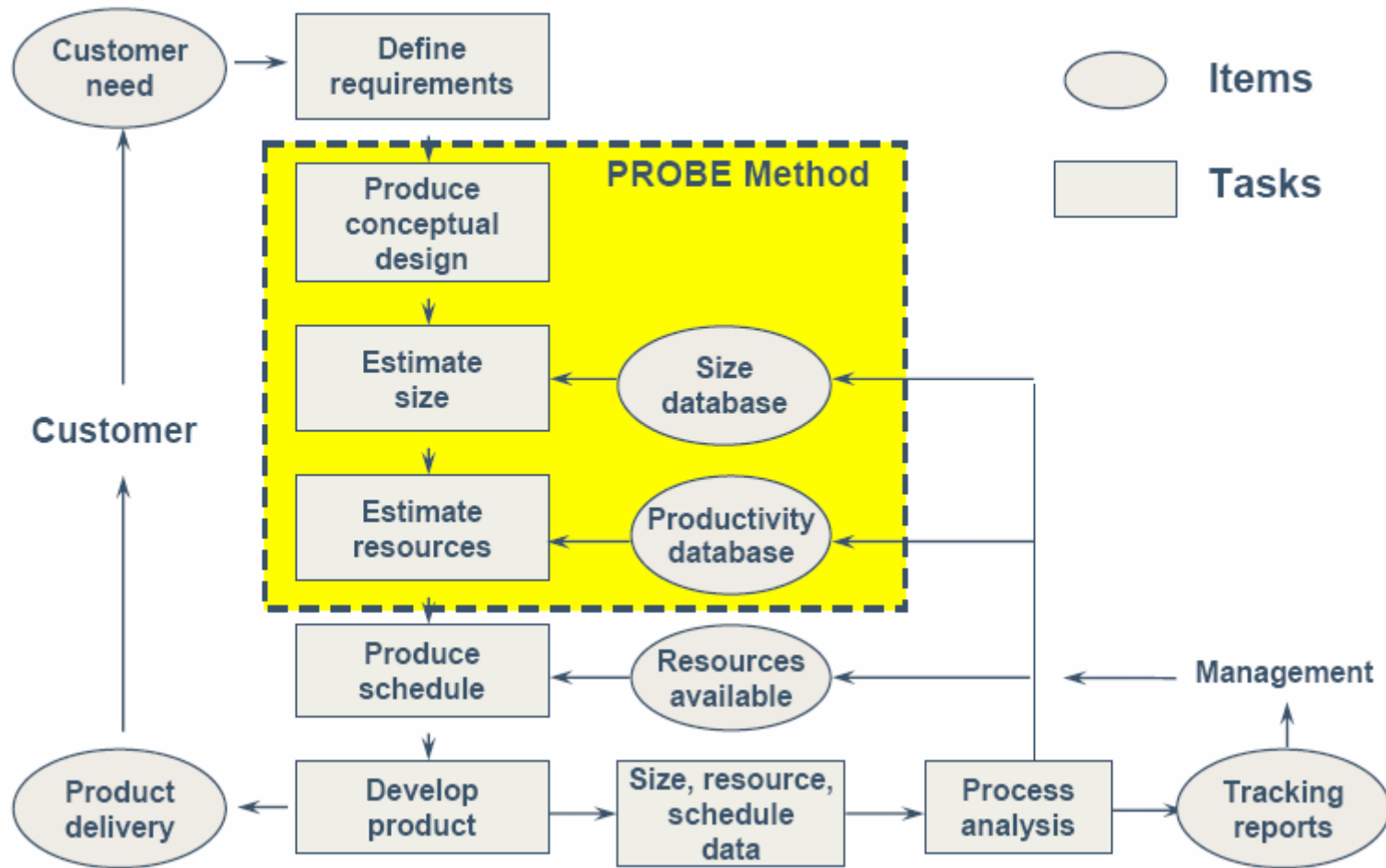
Proceso de planificación

- Antes de armar la planificación se deben conocer los requerimientos.
 - Cuanto mejores y más claros sean los requerimientos, mejor será la planificación.
 - Con requerimientos mal definidos, es esperable realizar actualizaciones frecuentes del plan.
- Es práctica común basar la planificación en una estimación de tamaño, del producto a desarrollar.

Proceso de planificación

- ❑ A menudo la causa de una mala planificación es una mala estimación de tamaño, del producto.
- ❑ Los planes más exactos son aquellos que se basan en estimaciones y datos históricos.
- ❑ Si se cuenta con estimaciones y datos históricos es posible,
 - Identificar datos de algún proyecto similar.
 - Utilizar estos datos para la estimación a realizar.

Framework para Planificación



19/06/2007

Instituto de Computación - Facultad de Ingeniería



Universidad Carlos III De Madrid
 Principios de Ingeniería Informática
 María Isabel Sánchez Segura
 José Arturo Mora-Soto
 Juan Carlos Alonso Durán

Estimación

- Estimamos principalmente para planificar.
- El tamaño es un buen predictor de los recursos necesarios para desarrollar un producto.
- La relación entre el tamaño y los recursos a motivado la creación de herramientas, para el modelado de costos.

Estimación

- Estimar es un proceso incierto.
 - No se conoce a priori el tamaño del producto final.
 - Cuanto más tempranas son las estimaciones, más inciertas.
 - Las estimaciones pueden ser influenciadas por presiones.

- Estimar es un proceso intuitivo de aprendizaje.
 - La habilidad mejora con experiencia y datos.
 - Algunas personas serán mejores en estimar que otras.

Estimación

- Estimar es una habilidad.
 - Debe ser aprendida, practicada y mantenida.
 - La mejora será gradual.
 - Podría pasar que nunca lleguemos a hacer buenas estimaciones.

- El objetivo es ser consistente.
 - Conocer y entender las variaciones de nuestras estimaciones.
 - Se busca un equilibrio entre las subestimaciones y las sobreestimaciones.

Estimación

- Ventajas de utilizar métodos definidos de estimación
 - Prácticas conocidas que se pueden mejorar.
 - Marco para recopilar datos de estimaciones realizadas.
 - Se pueden utilizar consistentemente datos históricos que permiten producir estimaciones balanceadas.

Métodos de Estimación

- WIDEBAND-DELPHI (Rand Corporation)
 - Un grupo de ingenieros expertos, individualmente realizan estimaciones de tamaño sobre el mismo producto.
 - Luego se usa el proceso Delphi, para llegar a una estimación consensuada.
 - Todos reciben las especificaciones del producto y un formulario de estimación.
 - Se reúnen para discutir el producto, objetivos, etc.
 - Anónima e individualmente hacen sus estimaciones.
 - Un moderador recopila la información, arma una tabla y la envía a los expertos.
 - Los expertos se reúnen para analizar los resultados.
 - Se vuelve al paso tres hasta que se converja a una estimación

Métodos de Estimación

□ FUZZY-LOGIC (Putnam)

- Los estimadores evalúan un producto y verifican si su tamaño es comparable con productos previamente desarrollados.
- Se mantiene una tabla donde se categorizan los productos previamente desarrollados, por su tamaño.
 - Se necesitan datos históricos de un gran número de productos.
- Se identifica a cuál de las categorías se aproxima más el nuevo producto a desarrollar.
 - Problema: El tamaño de la aplicación más grande aumenta un orden de magnitud cada 10 años.

Métodos de Estimación

□ STANDARD-COMPONENT (Putnam)

- Se describe como una forma de organizar los datos históricos, para refinar las estimaciones progresivamente.
- Se comienza recopilando datos de distintos tipos de componentes de un programa, módulos, pantallas, reportes, etc.
- Luego se determina cuantos de estos componentes se utilizarán en el nuevo programa.
- Por último se aplica una fórmula

$$\text{cant est.} = (\text{cant menor esp} + (4 * \text{cant esp}) + \text{cant mayor esp}) / 6$$

Métodos de Estimación

□ PUNTOS DE FUNCION

- El más popular de los métodos de estimación.
- Identifica cinco componentes básicos de todo software comercial.
 - **Entradas:** Formularios o pantallas de entrada de datos
 - **Salidas:** Pantallas o reportes hacia el usuario
 - **Preguntas:** Formularios de ayuda para el usuario
 - **Archivos de Datos:** Colecciones o registros que la aplicación modifica
 - **Interfaces:** Archivos compartidos con otras aplicaciones.
- Cada uno de estos componentes tiene un peso definido

Métodos de Estimación

- Para la estimación se revisan los requerimientos y se cuenta la cantidad de cada uno de estos componentes que tendrá el nuevo producto.
- Multiplicando la cantidad de componentes de un tipo, por su peso se obtiene el número de puntos de función en cada categoría.
- Se recomienda también ajustar este valor, multiplicándolo por un factor de 0 a 5 según la complejidad.
- Problemas:
 - No pueden ser directamente medidos.
 - No son sensibles a las decisiones de implementación.

PROBE

- ❑ PROBE significa PROxy Based Estimating.
- ❑ Se basa en los conceptos introducidos por los cuatro métodos más conocidos de estimación.
- ❑ PROBE utiliza proxies para estimar el tamaño de producto y tiempo de desarrollo.
- ❑ Un proxy adecuado, ayuda a realizar estimaciones mas exactas.

PROBE - Proxies

- Para una buena estimación se necesita conocer el producto.
- Generalmente es difícil visualizar los detalles del producto, en la primeras etapas del proyecto.

- Alternativas
 - Esperar y estimar cuando se conozca el detalle.
 - Adivinar.
 - Identificar un proxy apropiado.

PROBE - Proxies

- ❑ Un proxy apropiado debería estar fuertemente correlacionado con el costo de desarrollo.
- ❑ Un proxy apropiado debería ser fácilmente visualizable tempranamente en el desarrollo.
- ❑ Debería ser una entidad física que pueda ser medida.

Ejemplo: Costos de Construcción

□ Problema

- El constructor necesita conocer el tamaño del área a construir (en metros cuadrados) para estimar el costo de la construcción.
- Los clientes normalmente no pueden describir sus necesidades en metros cuadrados.
- Generalmente se describe el tipo y número de habitaciones que se desean.

□ Solución

- Se usan las habitaciones como proxy.
- Se usan datos históricos para traducir de cuartos a metros cuadrados.

PROBE – Estimación con Proxies

- La forma común de estimar es
 - Descomponer el producto a construir, en distintas piezas.
 - Relacionar estas piezas, con piezas desarrolladas anteriormente.
 - Utilizar el tamaño de las piezas desarrolladas anteriormente, para estimar el tamaño de las nuevas piezas.
- Para esto se necesita conocer, rangos de tamaños para los diferentes tipos de piezas que se desarrollan habitualmente.

PROBE – Estimación con Proxies

- Una vez que se tiene el proxy seleccionado
 - Obtener datos de tamaño sobre el proxy.
 - Organizar los datos y categorizarlos, para poder utilizarlos en estimaciones.
 - Utilizar los datos para estimar el tamaño de los proxies en el nuevo producto.
 - Combinar las estimaciones para los proxies y obtener la estimación para el producto.
 - Hacer una estimación de recursos.
 - Armar el plan de proyecto.
- El método PROBE muestra como seguir estos pasos.

Ejemplo: Requerimientos

□ Construcción de una casa

Cuartos:	1 grande, 2 mediano, y 1 pequeño
Baños:	1 grande and 2 mediano
Cocinas:	1 mediano
Living:	1 grande
Comedor:	1 mediano
Sala de estar:	1 grande

Ejemplo: Datos Históricos

	Pequeño	Mediano	Grande
Cuartos	8	13	18
Baños	3	6	12
Cocinas	9	12	15
Living	14	23	37
Comedores	9	13	18
Salas de Estar	14	22	31

Ejemplo: Cálculos

Cuarto	1 grande	=	1 x	18	=	18
Cuarto	2 mediano	=	2 x	13	=	26
Cuarto	1 pequeño	=	1 x	8	=	8
Baño	1 grande	=	1 x	12	=	12
Baño	2 mediano	=	2 x	6	=	12
Cocina	1 mediano	=	1 x	12	=	12
Living	1 grande	=	1 x	37	=	37
Comedor	1 mediano	=	1 x	13	=	13
Estar	1 grande	=	1 x	31	=	31
			Total		=	157

Ejemplo: La estimación

- ❑ El primer paso de la estimación provee al constructor los datos sobre el tamaño del sitio en función del proxy.
- ❑ Sin embargo, hay muchos otros costos en la construcción de hogares.
- ❑ Los constructores tienen típicamente datos extensos para relacionar tamaño del sitio con los demás costos de construcción.

PSP - PROBE

- PSP utiliza el método PROBE para estimar tamaño y recursos (tiempo).
 - Se utilizan estas estimaciones en la planificación de nuevos programas.
- Se utilizan como proxies las clases u objetos que componen el producto.
 - Correlación con horas de desarrollo
 - El número de clases correlaciona razonablemente bien.
 - El tamaño de las clases tiene una fuerte correlación con el tiempo de desarrollo.
 - El tamaño de clase puede ser estimado usando datos históricos.
 - Pueden ser visualizadas tempranamente en el desarrollo

Método de estimación PROBE

- Objetivo: proporcionar al ingeniero un mecanismo fiable y repetible de estimación de esfuerzo
 - PSP asume que el rendimiento de un ingeniero es comparable al obtenido en pasadas realizaciones.
 - PSP crea un marco que permite al ingeniero utilizar registros históricos, en nuevos planes.
 - Se basa en la existencia de un registro personal de objetos, desarrollados por el ingeniero, con datos de tamaño y productividad.

Ejemplo C++ Rangos de Tamaños

Type	LOC per item				
	VS	S	M	L	VL
Calculation	2.34	5.13	11.25	24.66	54.04
Data	2.60	4.79	8.84	16.31	30.09
I/O	9.01	12.06	16.15	21.62	28.93
Logic	7.55	10.98	15.98	23.25	33.83
Set-up	3.88	5.04	6.56	8.53	11.09
Text	3.75	8.00	17.07	36.41	77.66

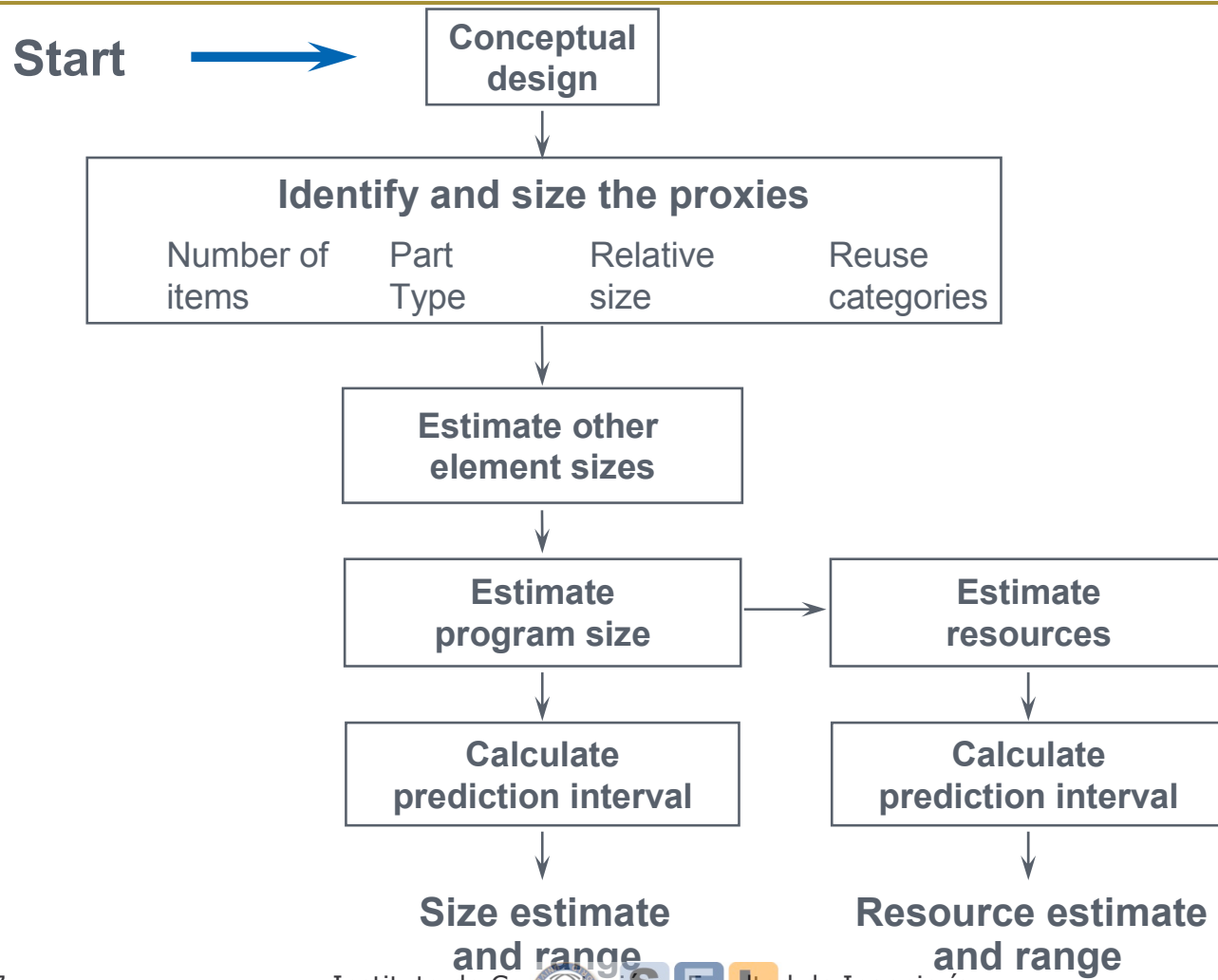
19/06/2007

Instituto de Computación - Facultad de Ingeniería

28



Método de estimación PROBE



Diseño Conceptual

- El primer paso en el proceso de estimación es realizar el diseño conceptual.
 - Relacionar los requerimientos con el producto.
 - Definir los componentes del producto que producirán las funciones deseadas.
 - Estimar el tamaño de lo que se planea construir.

- Si no se entiende el producto lo suficiente para hacer un diseño conceptual, no se sabe lo suficiente para hacer un plan.

Diseño Conceptual

- ❑ Para la mayoría de los proyectos, el diseño conceptual puede ser creado relativamente rápido.
- ❑ Identificar las funcionalidades del producto y las piezas de programa necesarias para producirlas.
 - “Si tuviera las piezas siguientes, podría construir este producto.”
- ❑ Comparar estas piezas con productos realizados y estimar sus tamaños.
- ❑ Finalmente, combinadas las estimaciones de cada pieza darán el tamaño total estimado.

Identificar Objetos

- En el diseño conceptual se representa y nombra cada objeto que compone el producto final esperado.
- El paso siguiente es identificar objetos similares, dentro de la base de datos de objetos, construidos en desarrollos previos.
- Sobre esta base se estima aproximadamente, en que categoría de tamaño caen los objetos a construir.

Estimar

- El producto terminado va a tener más que los objetos del diseño conceptual
 - Rutina principal
 - Clases de constantes
 - Imports.
- El tamaño de este código adicional es generalmente proporcional al tamaño del programa.
- Por esta razón se debe aplicar un factor de ajuste, basado en datos históricos.
- Para esto se utiliza el método de **Regresión Lineal**

$$Tamaño_real_esperado = \beta_0 + \beta_1 * tamaño_estimado$$

Estimar

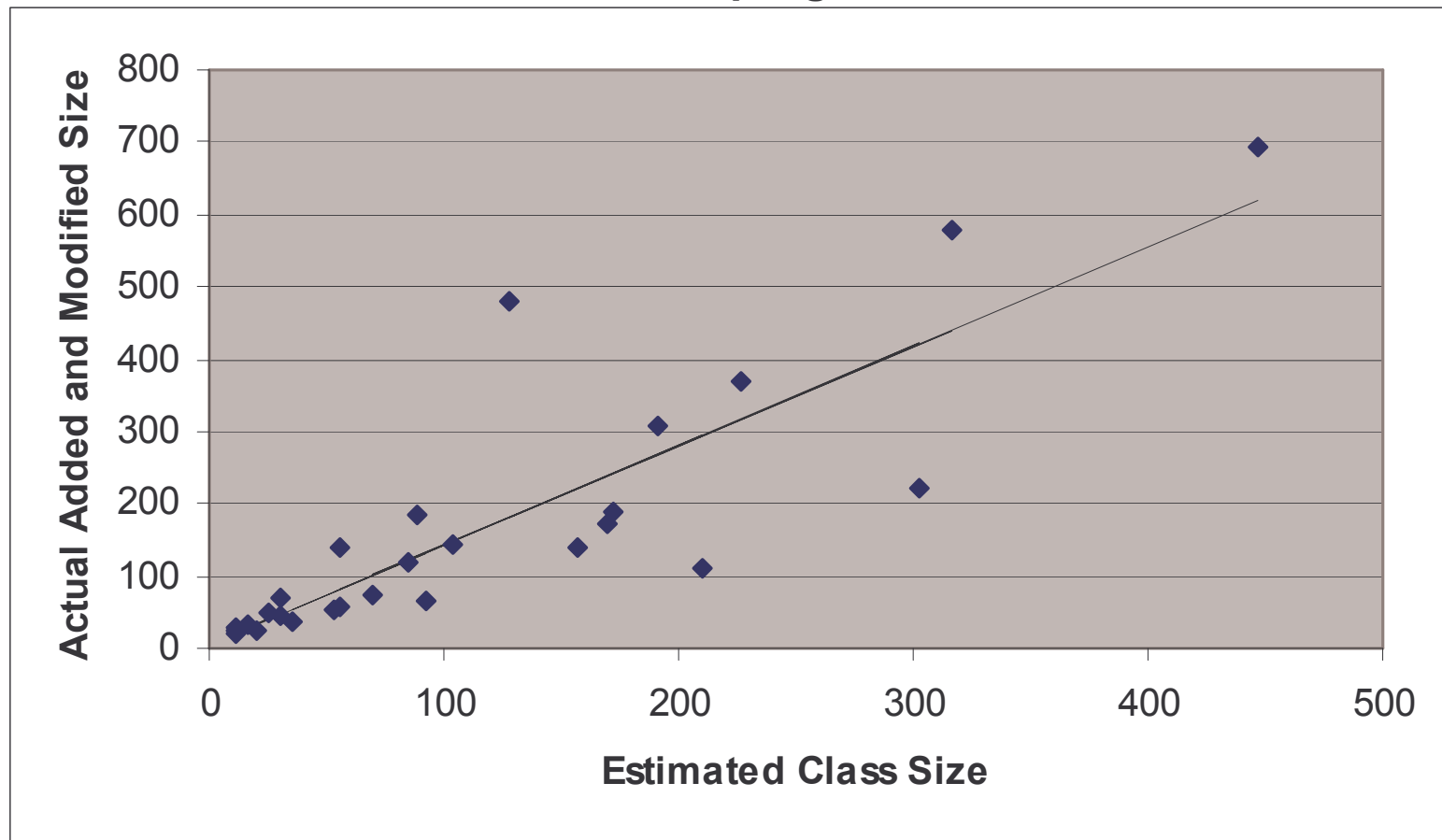
□ Regresión lineal

- Es la forma óptima de aproximar una línea a un conjunto de datos.
- La línea de Regresión Lineal es la línea donde la distancia de todos los puntos a esa línea se reduce al mínimo.
- La ecuación de esta línea es:

$$y = \beta_0 + \beta_1 x$$

Estimar

27 C++ programs



Estimar tiempo y tamaño

- Para estimar el tamaño de un programa y el tiempo de desarrollo, primero hay que encontrar la ecuación para la línea de Regresión Lineal.
- Calcular los parámetros de regresión lineal β_0 y β_1 basado en datos históricos.
 - Para x , se utilizan tamaños estimados por proxy.
 - Para y , se utilizan datos reales
 - LOC Added & Modified para estimar tamaño
 - Tiempo total de desarrollo para estimar tiempo.

Estimar tiempo y tamaño

$$\beta_1 = \frac{\left(\sum_{i=1}^n x_i y_i \right) - (n x_{avg} y_{avg})}{\left(\sum_{i=1}^n x_i^2 \right) - (n x_{avg}^2)} \quad \beta_0 = y_{avg} - \beta_1 x_{avg}$$

- β_0 y β_1 se calculan a partir de datos históricos utilizando las fórmulas anteriores
- x e y son conjuntos de datos.
 x es un conjunto de estimaciones por proxy e y es un conjunto de datos reales
- n es el número de programas a considerar
- x_{avg} es la media del conjunto x
- y_{avg} es la media del conjunto y

Estimar tiempo y tamaño

- Se calculan dos juegos de parámetros β_0 y β_1 , unos para estimar tiempo, otros para estimar tamaño.
- Luego se utiliza la fórmula de la regresión lineal para determinar tamaño y tiempo real esperado.
- E es el tamaño estimado utilizando el proxy.

$$y_k = \beta_0 + \beta_1 * x_k$$

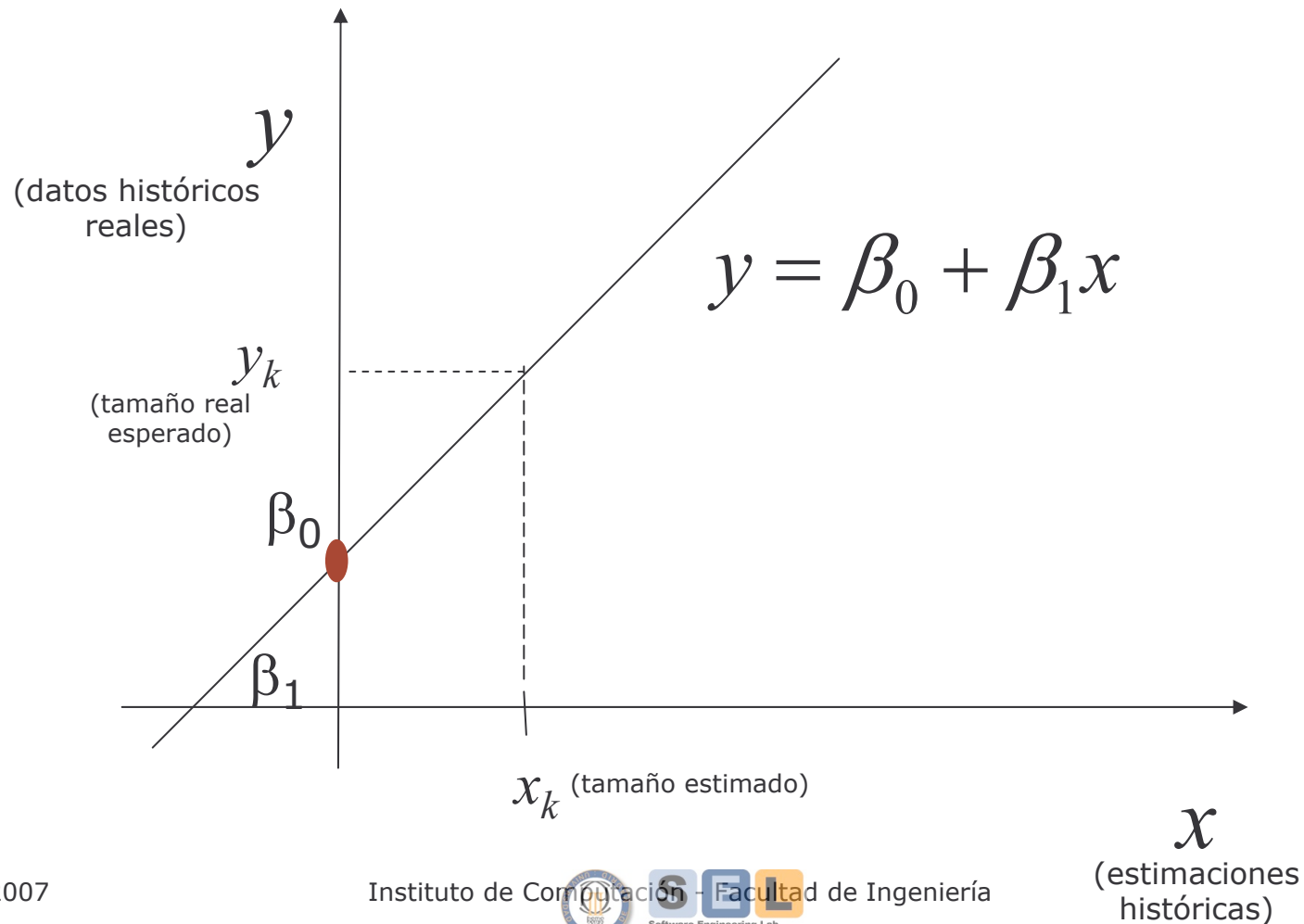
$$\textit{Added \& Modified _ Size} = \beta_{0size} + \beta_{1size} * (E)$$

$$\textit{Development _ Time} = \beta_{0time} + \beta_{1time} * (E)$$

Estimar tiempo y tamaño

- Para poder utilizar regresión lineal
 - β_0 debería ser substancialmente menor que el tamaño o tiempo de desarrollo esperados.
 - Es el tamaño (tiempo) real cuando el tamaño (tiempo) estimado es 0.
 - Es un ajuste fijo, por lo tanto debería ser menor en comparación.
 - $\beta_0 = 62$ significa que históricamente en promedio estimamos 62 líneas menos.
 - β_1 si estimamos tamaño, debería ser positivo y relativamente cercano a 1.
 - $\beta_1 = 1.3$ implica que históricamente los programas han sido 30% mayores que lo estimado
 - $1/\beta_1$ si estimamos tiempo, debería ser relativamente cercano al valor histórico de LOC/hora.
 - Utilizar los 10 valores más recientes.

Estimar tiempo y tamaño



19/06/2007

Instituto de Computación - Facultad de Ingeniería



Universidad Carlos III De Madrid
Principios de Ingeniería Informática
María Isabel Sánchez Segura
José Arturo Mora-Soto
Juan Carlos Alonso Durán

40

Estimar tiempo y tamaño

- Para una estimación completa calculamos
 - Tamaño de código agregado y modificado esperado, utilizando los parámetros de regresión.
 - Tamaño total de programa esperado, incluyendo código agregado, modificado, borrado, base y reutilizado.
 - Tamaño de código nuevo reutilizable, que será agregado a las librerías
 - Una estimación completa incluye estimar el tiempo de desarrollo, utilizando parámetros de regresión.

Intervalo de predicción

- Una vez hecha la estimación, es necesario evaluar su calidad.
- Para ello se calcula un intervalo alrededor de la estimación, en el cuál es probable que caiga la medida real.
- Con el tamaño y tiempo estimado se calcula
 - intervalo superior de predicción del 70% (UPI) e inferior (LPI), para el tamaño estimado.
 - intervalo superior de predicción del 70% (UPI) e inferior (LPI), para el tiempo de desarrollo.

Resumen

- ❑ Estimaciones exactas de tamaño ayudan a generar una correcta planificación del desarrollo.
- ❑ La habilidad de estimar mejora con la práctica.
- ❑ Un proceso definido y medido provee una base para la mejora.
- ❑ Para hacer estimaciones exactas se deben utilizar datos históricos y aplicar un método reconocido y probado de estimación.
- ❑ El método PROBE asiste en este proceso.

Resumen

- PROBE utiliza datos históricos, regresión lineal e intervalos de predicción para producir estimaciones con cierto grado de exactitud.
 - La regresión provee la mejor aproximación de una línea hacia los datos históricos.
 - La varianza de los datos determina el error probable en la estimación.
 - Cuanto más grande la varianza, más grande será el error probable.