

Administración de Proyectos de Software

Métricas en la Ingeniería de Software

E. Estévez - P. Fillotrani

Depto. Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur

segundo semestre 2019



Conceptos Iniciales

- ▶ las medidas y las métricas se pueden definir como:
herramientas que ayudan en la planificación y estimación de proyectos, y a su vez proporcionan datos cuantitativos sobre la calidad y productividad del proceso y del producto



Métricas en la Ingeniería de Software

Métricas



Medidas, Métricas e Indicadores

- ▶ **medidas**: manejan valores independientes
- ▶ **métricas**: manejan relaciones de medida e indican una medida de calidad o productividad
- ▶ ejemplo: *LOC/persona*
- ▶ **indicadores**: evalúan una o más métricas para sacar conclusiones respecto a algún aspecto del software o del proceso



¿Por qué medir?

- ▶ los **gerentes de proyecto** miden atributos del proceso y del producto poder responder:
 - ▶ cuándo el software estará disponible
 - ▶ si el presupuesto será cumplido o no
- ▶ los **clientes expertos** miden los aspectos del producto final para determinar si sus requerimientos fueron satisfechos
- ▶ los **encargados de mantenimiento** miden el producto actual para saber qué debe mejorarse o actualizarse



Medidas cotidianas

- ▶ medidas en **Economía**: determinan variaciones de precios y salarios
- ▶ medidas en **Sistemas de Radares**: permiten volar a los aviones
- ▶ medidas en **Medicina**: permiten a los doctores diagnosticar enfermedades
- ▶ medidas en **Sistemas Atmosféricos**: predicen el tiempo
- ▶ sin medidas no puede funcionar la tecnología



¿Qué es una Medida?

- ▶ **medición**: es el proceso por el cual se asignan números o símbolos a atributos de entidades del mundo real para que lo describan de acuerdo a reglas definidas claramente
- ▶ **entidad**: es un objeto o un evento del mundo real
- ▶ **atributo**: es una propiedad de una entidad
- ▶ se miden atributos de las entidades
- ▶ los números y símbolos utilizados son abstracciones que:
 - ▶ usamos para reflejar nuestra percepción del mundo real
 - ▶ preservan las relaciones que observamos entre las entidades



Medir

- ▶ la idea de medición hace los conceptos más visibles y por lo tanto más comprensibles y controlables
- ▶ el debate de cómo medir ya permite mayor comprensión
- ▶ existen dos clases de cuantificaciones:
 - ▶ **medición**: es una cuantificación directa
 - ▶ **cálculo**: es una cuantificación indirecta. Se toman las medidas y se combinan en un ítem cuantificado que refleja algún atributo
- ▶ ejemplo de medición: medir una habitación; ejemplo de cálculo: la valuación de una casa



Características de las Métricas

- ▶ las métricas deben ser:
 - ▶ **exactas**
 - ▶ **precisas** no se debe perder información en los redondeos ya que la información se desvirtúa
 - ▶ **consistentes** distintas mediciones del mismo atributo deben dar el mismo valor



Proceso de Adopción de Métricas

- ▶ **fase de aprendizaje**
 - ▶ no se tienen métricas, y es necesario analizar muchas medidas ya que no se sabe cuáles contribuirán a una métrica útil
 - ▶ esto implica mucho esfuerzo y poco beneficio
- ▶ **fase de uso** ya se tienen las métricas, poco esfuerzo y aumenta el beneficio



Medir en el Software

- ▶ **“No se puede controlar lo que no se puede medir”** De Marco, 1982
- ▶ Gerentes:
 - ▶ ¿cuánto cuesta este proceso?
 - ▶ ¿qué productividad tiene el personal?
 - ▶ ¿cuánto de bueno es el producto desarrollado?
 - ▶ ¿estará el usuario satisfecho?
 - ▶ ¿cómo se puede mejorar?
- ▶ Ingenieros:
 - ▶ ¿podemos verificar los requerimientos?
 - ▶ ¿hemos encontrado todos los errores?
 - ▶ ¿hemos cumplido los objetivos?
 - ▶ ¿qué pasará en el futuro?



Objetivos de Medir en el Software

- ▶ las medidas nos ayudan a:
 - ▶ **comprender** lo que sucede durante el desarrollo y mantenimiento
 - ▶ **controlar** proyectos
 - ▶ **mejorar** productos y procesos



Métricas en el Software

- ▶ estimación de Costo y Esfuerzo - Modelo COCOMO, de Putnam, de Albrecht
- ▶ medidas y modelos de productividad
- ▶ modelos y medidas de calidad
- ▶ modelos de confiabilidad



Teoría Representacional

- ▶ en cualquier actividad de medición, hay reglas a seguir
- ▶ las reglas nos ayudan a ser consistentes con nuestras mediciones, y nos proveen una base para la interpretación de los datos
- ▶ la teoría de medición nos dice las reglas
- ▶ el enfoque basado en reglas es común en muchas ciencias
- ▶ dependiendo del conjunto de reglas elegidas, existen distintas teorías
- ▶ se presentará la [teoría representacional de mediciones](#)
- ▶ la teoría representacional de las mediciones trata de formalizar nuestra intuición sobre la forma en que funcionan las cosas



Relaciones

- ▶ las medidas deberían representar atributos de las entidades que observamos
- ▶ la manipulación de estos datos deberían preservar las relaciones que observamos
- ▶ nuestra intuición es el punto de partida para todas las mediciones



Ejemplo

- ▶ por ejemplo, para aprender sobre altura, podemos decir que A es más alto que B sin medirlos
- ▶ en general, definimos una relación binaria. Ejemplo: “más alto que” es una relación binaria definida en el conjunto de pares de personas
- ▶ dadas cualesquiera dos personas, x e y , podemos observar que: x es más alto que y o y es más alto que x
- ▶ “más alto” que es una relación empírica para altura



Relación Empírica

- ▶ una **relación empírica** (binaria) es aquella para la cual hay un consenso razonable acerca de qué pares están en la relación
- ▶ las relaciones empíricas no necesariamente son binarias. Pueden ser unarias o de grado > 2
- ▶ ejemplo de relación unaria: es alto
- ▶ estas relaciones son mapeos del mundo real, empírico, a un mundo matemático formal
- ▶ el mapeo matemático debe preservar las relaciones que observamos



Medición

- ▶ formalmente se define medición como un mapeo del mundo empírico a un mundo formal, relacional
- ▶ una medida es el número o símbolo asignado a una entidad por este mapeo en orden a caracterizar un atributo
- ▶ algunas veces, las mediciones no son exactas
- ▶ dependen de características subjetivas de la persona que realiza la clasificación
- ▶ ejemplo: la tarea de catar vinos - se realiza una aseveración subjetiva, pero el resultado no es necesariamente una medida, en el sentido de la teoría de las mediciones



Medición

- ▶ la medición debe especificar el **dominio**, el **rango** y las **reglas** para realizar el mapeo
- ▶ el mundo real es el dominio del mapeo, y el mundo matemático es el rango
- ▶ **regla**: el mapeo debe preservar la relación
- ▶ esta regla es llamada: representación u homomorfismo



Medición

- ▶ **el mapeo debe preservar la relación**
- ▶ cuando se verifica la regla de representación, se puede definir al mapeo como una medida para el atributo
- ▶ esta condición asegura que un mapeo de medición M , debe mapear:
 - ▶ entidades en números
 - ▶ las relaciones empíricas en relaciones numéricas
 de tal manera que las relaciones empíricas preserven y sean preservadas por las relaciones numéricas
- ▶ por ejemplo A es más alto que B si y solo si $M(A) > M(B)$



Proceso de medir

1. identificar atributos de algunas entidades del mundo real
2. identificar la relación empírica para el atributo
3. identificar la relación numérica correspondiente a cada relación empírica
4. definir el mapeo de las entidades del mundo real a números
5. controlar que las relaciones numéricas preserven y sean preservadas por las relaciones empíricas



Modelos

- ▶ un **modelo** es una abstracción de la realidad, que permite eliminar detalles y visualizar una entidad o concepto desde una perspectiva particular
- ▶ el modelo del mapeo debería suplementarse con un modelo del dominio del mapeo, es decir con un modelo de cómo se relacionan las entidades con sus atributos
- ▶ ejemplo: para medir la longitud de un programa necesitamos un modelo del programa
- ▶ en el proceso de mediciones existe un peligro: focalizar demasiado en el sistema matemático formal y no lo suficiente en el empírico



Medidas directas e indirectas

- ▶ **medidas directas** en Ingeniería de Software:
 - ▶ longitud de código fuente (LOC)
 - ▶ duración del proceso de testeo (horas)
 - ▶ número de defectos descubiertos durante el testeo (número. de defectos)
- ▶ **medidas indirectas** en Ingeniería de Software:
 - ▶ productividad de programadores (LOC/unidad de tiempo) – es conflictiva
 - ▶ densidad de defectos (número de defectos/tamaño),
 - ▶ eficiencia en detección de errores (número de defectos detectados/número de defectos)



Medir para predecir

- ▶ cuando se habla de medición se presume que se calcula sobre una entidad existente
- ▶ en algunos casos necesitamos predecir un atributo de una entidad que no existe
- ▶ ejemplo: confiabilidad de un producto
- ▶ la diferencia entre medición para asegurar y predicción no siempre es clara. Ejemplo: un viaje en auto
- ▶ las mediciones para predicción siempre requieren alguna clase de modelo matemático que relacione los atributos a predecir con algún otro atributo de una entidad existente que se pueda medir
- ▶ los modelos no necesitan ser complejos para ser útiles



Medir para predecir

- ▶ la predicción del esfuerzo es universalmente necesaria para los líderes de proyecto
- ▶ ejemplo: predecir la cantidad de páginas de un listado de un programa fuente

$$\text{cantidad páginas} = \text{LOC}/\text{líneas por página}$$

- ▶ un **sistema de predicción** consiste de un modelo matemático junto con un conjunto de procedimientos de predicción para determinar parámetros desconocidos y para interpretar resultados
- ▶ algunas veces el mismo modelo es usado para evaluar y para predecir



Medir para predecir

- ▶ ejemplo: predecir el costo de un viaje en auto de Bahía Blanca a Buenos Aires
- ▶ se comienza por obtener algunas medidas:
 - ▶ d : la distancia entre Bahía Blanca y Buenos Aires
 - ▶ g : el costo del litro de gasoil
 - ▶ l : el promedio de distancia que se recorre con un litro de gasoil
- ▶ se puede predecir el costo del viaje como:

$$\text{costo} = d * g / l$$



Medir para predecir

- ▶ se usa un sistema de predicción que incluye
 - ▶ un modelo, la fórmula $\text{costo} = d * g / l$
 - ▶ un conjunto de procedimientos - para determinar los parámetros del modelo, cómo determinamos los valores d , g y l
 - ▶ procedimientos para interpretar los resultados, por ejemplo podemos determinar una función de probabilidades para determinar el margen de error
- ▶ aún usando el mismo modelo se pueden lograr distintos resultados si se utilizan procedimientos de predicción diferentes



Escalas de medición

- ▶ no todos los mapeos de mediciones son iguales. Las diferencias entre los mapeos pueden restringir la clase de análisis que se puede realizar
- ▶ una **escala de medición** esta dada por el mapeo de medición M , junto con el sistema de relación numérica
- ▶ el sistema de relación numérica, dado por el dominio y el rango, son obvios a partir del contexto
- ▶ a menudo, nos referimos sólo a M como la escala



Escalas de medición

- ▶ ¿cómo se determina que un sistema de relación numérico es mejor que otro? **respuesta pragmática:** en lo posible usar números reales
- ▶ **problema de representación:** ¿Cómo sabemos, si un sistema de relación empírico particular, tiene una representación en un sistema de relación numérico dado? **respuesta:** es tema de estudio de la teoría de mediciones
- ▶ **problema de unicidad:** ¿Qué hacer cuando tenemos varias representaciones posibles diferentes (varias escalas) en el mismo sistema de relación numérico? **respuesta:** la solución depende de nuestra habilidad para determinar qué representación es la más adecuada para medir un atributo



Tipos de Escala

- ▶ nominal
- ▶ ordinal
- ▶ de intervalos
- ▶ de razones
- ▶ absoluta



Tipos de Escala

- ▶ un sistema relacional se dice **más rico** que otro, si todas las relaciones del segundo están contenidos en el primero
- ▶ los tipos de escala mencionados, están enumerados en orden creciente de niveles de riqueza
- ▶ cuanto más rico es el sistema de relación empírico, más restrictivo es el conjunto de representaciones, y por lo tanto, más sofisticada la escala de medición



Tipos de Escala

- ▶ la idea detrás de las definiciones formales de tipos de escala es simple: tenemos una medida satisfactoria para un atributo, con respecto a un sistema de relación empírico
- ▶ queremos saber qué otras medidas existen que sean también aceptables
- ▶ ejemplo: medimos la longitud de objetos físicos en pulgadas. Existen otras medidas aceptables: cm., pies, millas, etc.
- ▶ un mapeo de un sistema de medición aceptable a otro sistema de medición se denomina transformación admisible
- ▶ ejemplo: en mediciones de longitud, la clase de transformaciones admisibles es restrictiva, y son de la forma $M' = a * M$



Escala Nominal

- ▶ se definen clases o categorías, y luego se ubica a cada entidad en una clase o categoría particular
- ▶ la escala nominal tiene 2 características importantes:
 - ▶ el sistema de relación empírico consiste sólo de diferentes clases - no existe noción de orden entre las clases
 - ▶ cualquier representación de distinción numérica o simbólica de las clases es una medida aceptable, pero no hay noción de magnitud asociada con los números o símbolos
- ▶ la medición de escala nominal ubica los elementos en un esquema de clasificación
- ▶ las clases no están ordenadas



Escala Nominal - ejemplo

- ▶ deseamos detectar el origen de los errores detectados en el software
- ▶ la escala de medición tiene a los errores como entidades y la “ubicación” como atributo
- ▶ el mapeo para determinar la ubicación será: “especificación”, “diseño” o “codificación” dependiendo de dónde fue introducido el error
- ▶ un error puede pertenecer sólo a una clase

$$M_1(x) = \begin{cases} 1 & \text{si } x \text{ es de especificación} \\ 2 & \text{si } x \text{ es de diseño} \\ 3 & \text{si } x \text{ es de codificación} \end{cases}$$



Escala Nominal - ejemplo

- ▶ la siguiente también es una medida aceptable:

$$M_2(x) = \begin{cases} 800 & \text{si } x \text{ es de especificación} \\ 55,6 & \text{si } x \text{ es de diseño} \\ 932 & \text{si } x \text{ es de codificación} \end{cases}$$



Escala Ordinal

- ▶ a menudo, es útil aumentar la escala nominal con información de orden de las clases o categorías
- ▶ la **escala ordinal** tiene las siguientes características:
 - ▶ el sistema de relación empírica consiste de clases que están ordenadas con respecto al atributo
 - ▶ cualquier mapeo que preserve el ordenamiento es aceptable
 - ▶ los números sólo representan un ranking
- ▶ no tienen sentido las operaciones matemáticas de suma, resta, multiplicación y división



Escala Ordinal - ejemplo

- ▶ el conjunto de entidades es un conjunto de módulos de software y el atributo a medir es la “complejidad”
- ▶ podemos definir 5 clases de complejidad de módulos: trivial, simple, moderado, complejo, incomprensible.
- ▶ existe un orden implícito que es “menos complejo que”
- ▶ en este caso, el mapeo de medición debe preservar el orden
- ▶ cualquier mapeo M debería ser una función creciente



Escala Ordinal - ejemplo

- ▶ medidas aceptables:

$$M_3(x) = \begin{cases} 1 & \text{trivial} \\ 2 & \text{simple} \\ 3 & \text{moderado} \\ 4 & \text{complejo} \\ 5 & \text{incomprensible} \end{cases} \quad M_4(x) = \begin{cases} 5 & \text{trivial} \\ 8 & \text{simple} \\ 33 & \text{moderado} \\ 90 & \text{complejo} \\ 784 & \text{incomprensible} \end{cases}$$



Escala Ordinal - ejemplo

- ▶ medidas no aceptables:

$$M_5(x) = \begin{cases} 1 & \text{trivial} \\ 2 & \text{simple} \\ 2 & \text{moderado} \\ 5 & \text{complejo} \\ 6 & \text{incomprensible} \end{cases} \quad M_6(x) = \begin{cases} 2 & \text{trivial} \\ 8 & \text{simple} \\ 4 & \text{moderado} \\ 6 & \text{complejo} \\ 10 & \text{incomprensible} \end{cases}$$



Escala de Intervalos

- ▶ esta escala captura información sobre el tamaño de los intervalos que separan las clases, de tal forma que permita comprender de alguna manera, la magnitud del salto de una clase a la otra
- ▶ la escala de intervalos tiene las siguientes características:
 - ▶ preserva el orden, al igual que una escala ordinal
 - ▶ preserva diferencias, pero no proporciones - conocemos las diferencias entre cualquier par de clases ordenadas en el rango de mapeo, pero no tiene sentido calcular la razón de dos clases
- ▶ la suma y la resta son aceptables, pero no las operaciones de multiplicación y división



Escala de Intervalos - ejemplo

- ▶ ejemplo de Medición de Temperatura: se miden 20 grados de temperatura en Bahía Blanca y 30 grados en Salta
- ▶ el intervalo de un grado es el mismo, si la temperatura aumenta de 20 a 21 grados en Bahía Blanca, el calor aumenta de la misma manera que si subiera de 30 a 31 grados en Salta



Escala de Intervalos - ejemplo

- ▶ se desea medir la complejidad, al igual que en el caso anterior
- ▶ se asume que la diferencia de complejidad entre trivial y simple, es la misma que entre simple y moderada
- ▶ cualquier medida de intervalos debe preservar estas diferencias. Una medida aceptable podría definirse como:

$$M_7(x) = \begin{cases} 0 & \text{trivial} \\ 2 & \text{simple} \\ 4 & \text{moderado} \\ 6 & \text{complejo} \\ 10 & \text{incomprensible} \end{cases}$$



Escala de Intervalos - transformación

- ▶ supongamos que un atributo es medible en esta escala, y M y M' son mapeos que satisfacen la condición de representación
- ▶ entonces, siempre podemos encontrar números a y b tal que:

$$M = a * M' + b$$

- ▶ este tipo de transformación se denomina transformación afin
- ▶ ejemplo: podemos transformar los grados Celsius a Fahrenheit utilizando la ecuación:

$$F = 9/5 * C + 32$$



Escala de Razones

- ▶ provee más información que la escala de intervalos
- ▶ ejemplo: se necesita saber que un proyecto consume dos veces más recursos que otro, o que un procesador es el doble de eficiente que otro
- ▶ en estos casos, se puede utilizar la escala de ratios
- ▶ la escala de razones tiene las siguientes características:
 - ▶ es un mapeo de medición que preserva el ordenamiento, el tamaño de los intervalos entre entidades, y las proporciones entre entidades
 - ▶ existe un elemento cero que representa la falta total del atributo
 - ▶ el mapeo de medición debe comenzar en cero y aumentar a intervalos iguales, conocidos como unidades



Escala de Razones

- ▶ se pueden aplicar operaciones aritméticas que tengan algún significado entre las clases en el rango del mapeo
- ▶ la diferencia fundamental con las escalas anteriores, es que en esta existe una relación empírica que captura ratios
- ▶ ejemplo: la longitud de objetos físicos es medible en una escala de ratios, permitiendo hacer enunciados que un objeto es el doble de largo que otro. Podemos medir en cm., metros, etc.
- ▶ en general, cualquier transformación posible para esta escala, es un mapeo de la forma $M = a * M'$, donde a es un escalar > 0
- ▶ el tipo de transformación se llama **transformación escalar**



Escala de Razones - ejemplo

- ▶ la longitud del código es medible en una escala de razones
- ▶ existe la relación empírica: “dos veces más largo”
- ▶ existe la noción del elemento cero: un código vacío
- ▶ podemos medir la longitud del código de varias maneras: *LOC*, *KLOC*
- ▶ supongamos que M es la medida en *LOC*, y M' es la medida en cantidad de caracteres
- ▶ podemos transformar M a M' , mediante $M = a * M'$, donde a es el promedio de caracteres de una línea de código



Escala Absoluta

- ▶ la escala absoluta es la más restrictiva de todas
- ▶ para dos medidas M y M' existe una única transformación posible: la transformación identidad
- ▶ existe una sola forma en la que la medida se puede tomar
- ▶ la escala absoluta tiene las siguientes características:
 - ▶ la medida se hace simplemente contando el número de elementos en el conjunto de entidades
 - ▶ siempre tiene la forma: número de ocurrencias de x en la entidad
 - ▶ existe sólo una medida posible: la cuenta actual
 - ▶ todos los análisis aritméticos del resultado son significativos



Escala Absoluta - ejemplos

- ▶ número de errores observados durante la etapa de testeo
- ▶ número de personas asignadas a un proyecto
- ▶ longitud de un programa en *LOC*???
- ▶ es un error medir *LOC* con una escala absoluta - Existen muchas formas de medir: *LOC*, *KLOC*, caracteres, etc
- ▶ *LOC* es una medida de escala absoluta del atributo “número de líneas de código” de un programa



¿Qué es una métrica?

- ▶ desde el momento en que uno asocia un número a una idea, se ha aprendido algo más
- ▶ una **métrica** es una indicación medible de algún aspecto cuantificable de un sistema
- ▶ aspecto cuantificable de un sistema: alcance, riesgo, costo, tiempo
- ▶ características de una métrica útil:
 - ▶ medible
 - ▶ independiente
 - ▶ controlable
 - ▶ precisa



Características: medible

- ▶ **medible**: el indicador debe ser medible para considerarlo una métrica. Si un factor es no medible es no cuántico
- ▶ definida una métrica, se debe estimar su valor antes de observarlo
- ▶ la diferencia entre la métrica no medida y un no cuántico, es que la métrica es escalable en forma uniforme, es mejorable, y resoluble por observaciones



Características: independiente

- ▶ debe ser independiente de la influencia consciente del personal del proyecto
- ▶ es independiente en la medida en que no se pueda cambiar salvo por avances del proyecto
- ▶ ejemplo: la entrega de documentos, no es independiente
- ▶ **principio de incertidumbre**: el medir cualquier parámetro y asociarlo con evidencia significativa afectará la utilidad del mismo



Características: controlable

- ▶ la colección y el análisis de datos de métricas es una actividad sujeta a error
- ▶ se deben guardar los datos crudos así como pistas de auditoría o datos de control: fecha de la medición, identidad del observador, autor de la tarea, etc
- ▶ se deben analizar los datos observables en tiempo de tal manera de poder realizar correcciones en el proceso



Características: precisa

- ▶ la precisión de las métricas, no debe ser maximizada, sino explícitamente señalada y registrada como parte de los datos recolectados
- ▶ la precisión puede indicarse como un rango, una tolerancia explícita o especificando el método de recolección de datos
- ▶ ejemplo: La longitud promedio de segmentos de código secuencial en el sistema XYZ es de 12.5 instrucciones
- ▶ la observación se realizó en una muestra de módulos entre 1000 y 6000 líneas de código
- ▶ los módulos incluidos pertenecen al subsistema XY1

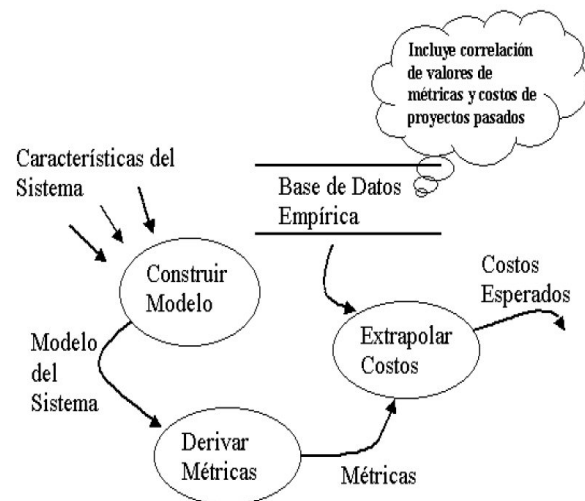


Características: resultados y predictores

- ▶ las métricas pueden clasificarse en: resultados y predictores
- ▶ un **resultado** es una métrica de costo, alcance o complejidad observada de un sistema terminado
- ▶ provienen de una exhaustiva y metódica recolección de datos del proyecto
- ▶ ejemplo: costo total, total de esfuerzo (manpower)
- ▶ un **predicador** es un métrica señalada en forma temprana, que tiene una fuerte correlación con algún resultado posterior
- ▶ provienen de los modelos de especificación del sistema



Proceso de Proyección de Costos



Proceso de Proyección de Costos

- ▶ el modelo se puede publicar. Pueden colaborar otras personas
- ▶ las características cuantitativas en un modelo público se pueden extraer de manera consistente
- ▶ imposible en un modelo intuitivo
- ▶ el modelo empírico puede crecer
- ▶ los participantes pueden aportar datos
- ▶ los datos se pueden recolectar por personas ajenas al proyecto
- ▶ se puede automatizar la parte de cálculo fuera de las proyecciones de costos



Uso de Predictores para FCE

- ▶ antes de que un predictor pueda ser observado, puede ser estimado
- ▶ el recolectar nuevos predictores, provee una oportunidad para producir predicciones más exactas
- ▶ esto se muestra en el gráfico de la transparencia siguiente
- ▶ **FCE** Factor de calidad de la estimación



Uso de Predictores para FCE

