


MÉTRICAS



El método es un proceso disciplinado para generar un conjunto de modelos que describen varios aspectos de un sistema de software en desarrollo, utilizando una notación bien definida.

Una metodología es una colección de métodos aplicados a lo largo del ciclo de vida del desarrollo y unificados por algún enfoque filosófico general.



El método de desarrollo de software tiene como componentes:

Conceptos básicos bien definidos para los que existen notaciones acordadas, que pueden usarse para describir artefactos de dominio y modelos del sistema previsto desde diferentes perspectivas.

Un proceso que describe los pasos para utilizar los conceptos centrales, que define los entregables intermedios y caracteriza las dependencias entre estos entregables.

Un criterio para medir el progreso que puede indicar cuándo se ha completado una actividad de desarrollo (subordinada).

MÉTRICAS EN GENERAL

"Las métricas de software se ocupan de las medidas del producto de software y el proceso mediante el cual se desarrolla".

El producto de software debe verse como un objeto abstracto que evoluciona desde una declaración inicial de necesidad hasta un sistema de software terminado, incluido el código fuente y objeto y las diversas formas de documentación producidas durante el desarrollo ".

Las métricas se pueden aplicar a cualquier artefacto construido durante el desarrollo, incluido no solo el código, sino también los modelos de análisis y diseño, así como sus componentes. También se pueden tener en cuenta los resultados derivados, como las contribuciones al análisis, el diseño y las bibliotecas de códigos ".

MÉTRICAS EN GENERAL

"El objetivo de las métricas de SW es la identificación y medición de los parámetros esenciales que afectan el desarrollo de SW".

- *Medición de los costos de desarrollo reales para un período de tiempo particular, posiblemente calificado por tipo de actividad de desarrollo*
- *Medición de fragmentos de desarrollo para predecir o estimar futuros costos de desarrollo.*
- *Medición de aspectos de calidad para predecir o estimar los costos de desarrollo posteriores para lograr una calidad aceptable del producto.*
- *Medición de aspectos de desarrollo para mejorar la conciencia general de “dónde estamos y hacia dónde vamos*

CARACTERÍSTICAS DESEABLES

1. Una métrica es elemental, ya que mide solo un aspecto bien definido.
2. Es objetiva, porque no depende del juicio de un humano y puede expresarse preferiblemente en un algoritmo ejecutable por máquina.
3. Se puede aplicar a un costo razonable.
4. Es intuitivo.
5. Es composicional; una métrica aplicada a un artefacto compuesto debería ser una especie de suma de la métrica aplicada a los componentes del artefacto.
6. Su dominio de valor es numérico y permite operaciones aritméticas significativas.

CLASIFICACIÓN

1. Las métricas de esfuerzo miden los recursos humanos, mecánicos, etc., que gastan durante un período de tiempo particular en un conjunto específico de tareas de desarrollo.
2. Las métricas de producto estáticas miden el volumen y la complejidad de los artefactos producidos en el ciclo de vida.
3. Las métricas de producto dinámicas miden el volumen de objetos fijos y transitorios de un sistema de destino en ejecución y la complejidad del gráfico de comunicación de los objetos.

REUTILIZACIÓN

No podemos simplemente asumir que los artefactos reutilizables se pueden recoger de forma gratuita.

Encontrar el artefacto correcto también puede no ser obvio.

Además, tener que ajustar un artefacto "un poquito" puede convertirse en una costosa pesadilla.

Por lo tanto, para tener en cuenta la reutilización, se deben introducir coeficientes de ajuste del esfuerzo de reutilización en las fórmulas de estimación del esfuerzo.

ESTIMACIÓN DE COSTOS DE DESARROLLO TEMPRANO

Relacionar métricas de productos y procesos para las fases de análisis y diseño; y viene como objetivo la construcción de estimadores de esfuerzo para actividades posteriores.

Table 11. Complexity Assignment for External Outputs

	1-5 Data Element Types	6-19 Data Element Types	20+ Data Element Types
0-1 File Types Referenced	Low	Low	Average
2-3 File Types Referenced	Low	Average	High
4+ File Types Referenced	Average	High	High

	Low	Average	High
External Input	*3	*4	*6
External Output	*4	*5	*7
Logical Internal File	*7	*10	*15
External Interface File	*5	*7	*10
External Inquiry	*3	*4	*6

VENTAJAS

Para los proyectos de software, cuanto mayor es el porcentaje de diseños y códigos que se inspeccionan, menor es la tasa de defectos en la última fase de la prueba formal de la máquina.

Cuanto más efectivas sean las revisiones de diseño y las inspecciones de código calificadas por el equipo de inspección, menor será la tasa de defectos en la fase posterior de la prueba formal de la máquina.