

CALIDAD DEL SOFTWARE



1

CALIDAD

- Aseguramiento de calidad: definición de procesos y estándares que deben conducir a la obtención de productos de alta calidad
- Control de calidad: aplicación de procesos de calidad para eliminar productos que no cuentan con el nivel de calidad requerido



MANEJO DE LA CALIDAD DEL SOFTWARE

- Se refiere a lograr un nivel de calidad requerido en el producto de software
- Involucra a la definición de estándares de calidad apropiados y procedimientos que permitan asegurar que estos se cumplan.
- Debe llevar a desarrollar una cultura de calidad en donde la calidad es responsabilidad de todos



QUE ES CALIDAD?

- Calidad significa que un producto debe cumplir con sus especificaciones
- Para sistemas de software
 - Algunos requerimientos de calidad son difíciles de especificar (eficiencia, mantenibilidad, reusabilidad, etc.)
 - Las especificaciones del Software son usualmente incompletas y a menudo inconsistentes



EL COMPROMISO DE CALIDAD

- No podemos esperar a que las especificaciones mejoren para poner atención al manejo de la calidad
- Debe haber procedimientos que permitan mejorar la calidad aunque la especificaciones no sean perfectas
- El manejo de la calidad no solo se refiere a reducir defectos sino también a mejorar otras cualidades del producto



ACTIVIDADES DE MANEJO DE CALIDAD

○ Aseguramiento de calidad

- Establecer procedimientos organizacionales y estándares para la calidad

○ Planeación de calidad

- Seleccionar procedimientos aplicables y estándares para un proyecto en particular y modificar estos como sean requeridos

○ Control de calidad

- Garantizar que procedimientos y estándares son seguidos por el equipo de desarrollo de software

El manejo de calidad debe ser separado del manejo del proyecto para asegurar independencia



ATRIBUTOS DE LA CALIDAD DEL SOFTWARE

Protección	Comprensibilidad	Portabilidad
Seguridad	Comprobabilidad	Usabilidad
Fiabilidad	Adaptabilidad	Reusabilidad
Flexibilidad	Modularidad	Eficiencia
Robustez	Complejidad	Facilidad para que el usuario aprenda a utilizarlo



CALIDAD BASADA EN PROCESOS

- Relacion directa entre procesos y productos
- Más complejo para software debido a:
 - Se requiere la aplicación de habilidades individuales y experiencia, la cual es importante para el desarrollo de software
 - Factores externos en las que una aplicación es novedosa o la necesidad para acelerar el calendario de desarrollo puede empeorar la calidad del producto
- Debe tenerse especial cuidado de no imponer estándares inapropiados

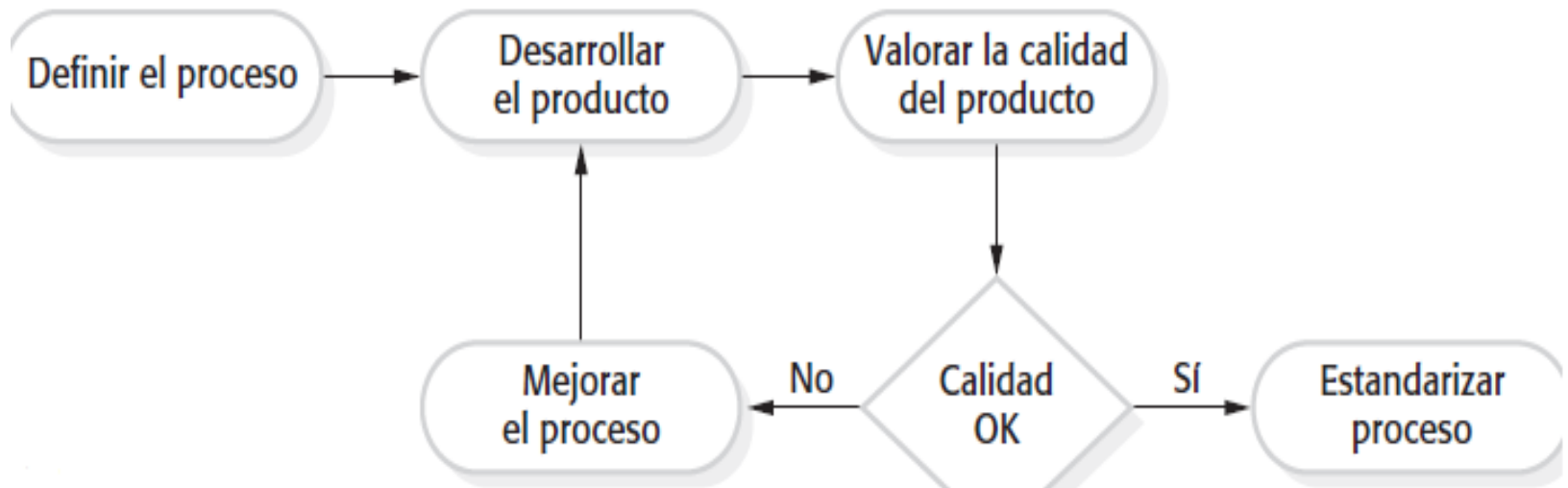


CALIDAD DE PROCESOS PRACTICA

- Definir procesos de estándares que indiquen como llevar a cabo las revisiones, la administración de la configuración, etc.
- Monitoriar el proceso de desarrollo para asegurar que se están siguiendo los estándares
- Reportar estos procesos a la administración del proyecto



CALIDAD BASADA EN PROCESOS



ESTÁNDARES DE SOFTWARE

- Son clave para un efectivo manejo de calidad
- Puede ser un proyecto internacional, nacional, o organizacional
- Los estándares de producto definen características que todos los componentes deberán tener



IMPORTANCIA DE LOS ESTÁNDARES

- Reúne las mejores practicas.
- Evita la repetición de errores pasados
- Proporciona un marco para el análisis de calidad - involucra verificar la conformidad con estándares
- Proporcionar continuidad. El personal nuevo puede entender a la organización entendiendo a los estándares aplicados



ESTÁNDARES DE PRODUCTOS Y PROCESOS

Estándares de producto	Estándares de proceso
Formato de revisión de diseño	Realizar revisión de diseño
Estructura de documento de requerimientos	Enviar nuevo código para construcción de sistema
Formato de encabezado por método	Proceso de liberación de versión
Estilo de programación Java	Proceso de aprobación del plan del proyecto
Formato de plan de proyecto	Proceso de control de cambio
Formato de solicitud de cambio	Proceso de registro de prueba



PROBLEMAS CON ESTÁNDARES

- No son vistos como relevantes ni se encuentran actualizados por los ingenieros de software
- Involucra muchas formas burocrática
- No soportado por herramientas de software por lo que se requieren actividades para mantener los estándares



DESARROLLO DE ESTÁNDARES

- Involucra a los desarrolladores. Los Ingenieros deberán entender la racionalidad bajo un estándar
- Revisión de estándares y su uso regularmente. Los estándares pueden rápidamente estar desactualizados lo cual reduce su credibilidad entre sus usuarios
- Los estándares detallados deberán tener asociado una herramienta de soporte.

Excesivo trabajo manual es el mas significativo reclamo en contra de los estándares



ISO 9000 - ISO 9001

- Conjunto de estándares internacionales para el manejo de calidad
- Aplicable a un rango de organizaciones desde industrias de servicio a industrias de manufactura
- La ISO 9001 es aplicable a organizaciones del cual diseñan, desarrollan y mantienen productos
- ISO 9001 es un modelo genérico del proceso de calidad. Está instanciado para cada organización



CERTIFICACIÓN ISO

- Los Estándares de calidad y procedimientos deberán ser documentados en un manual organizacional de calidad
- Personal externo puede certificar que una organización conforma con los estándares ISO 9000/9001
- Los clientes demandan cada vez mas que sus desarrolladores tengan la certificación ISO 9000/9001



ISO 9001

Procesos de entrega de producto

Adquisición
empresarial

Diseño y
desarrollo

Prueba

Producción y
entrega

Servicio y
soporte

Procesos de soporte

Administración
empresarial

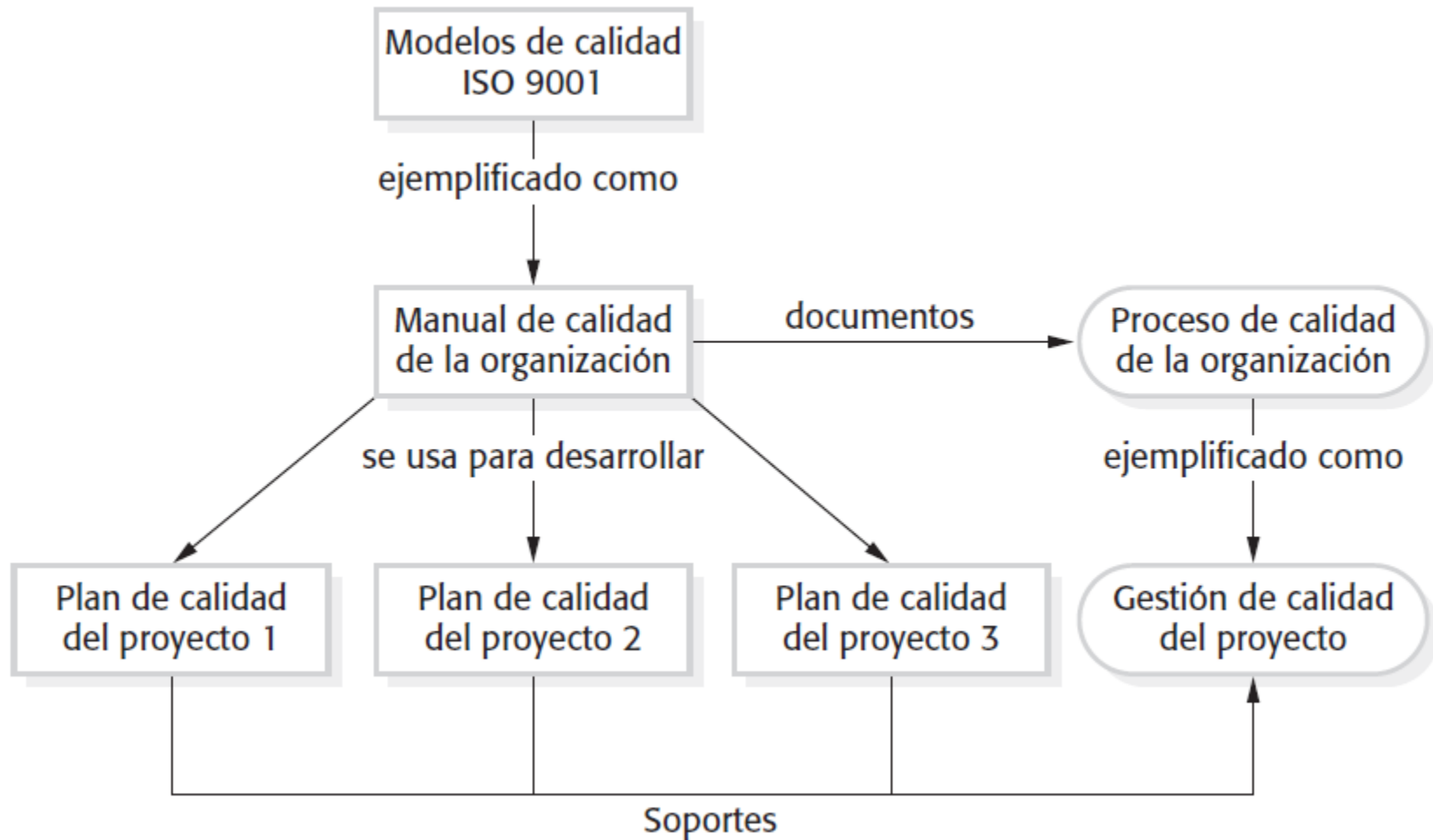
Administración
del proveedor

Administración
del inventario

Administración
de la configuración



ISO 9001 Y MANEJO DE CALIDAD



REVISIÓN

- El principal método de validación de la calidad de un proceso o de un producto
- Un grupo debe examinar parte o toda su documentación para buscar problemas potenciales
- Hay diferentes tipos de revisiones con diferentes objetivos
 - Inspecciones para remover defectos (producto)
 - Revisiones para estimación de progresos (procesos y producto)
 - Revisiones de calidad (estándares y producto)



PROCEDIMIENTOS DE REVISIÓN

- Función de calidad - Es parte del proceso general de administración de calidad
- Función de administración del proyecto - proveen información para los administradores del proyecto
- Funciones de comunicación y entretenimiento - Paso de conocimientos entre miembros de desarrollo del equipo



REVISIONES DE CALIDAD

- Un grupo de personas cuidadosamente examinará cada parte o todo un sistema de software y su documentación asociada
- Código, diseños, especificaciones, planes de prueba, estándares, etc. todo puede ser revisado
- Documentos o Software puede ser “firmados” en cada revisión lo cual significa que la administración ha aprobado el progreso de la siguiente etapa del desarrollo



REVISIONES DE CALIDAD

- El objetivo es descubrir defectos en el sistema e inconsistencias
- Cualquier documento producido en el proceso puede ser revisado
- El equipo de revisión deberá ser relativamente pequeño y las revisiones deberán ser relativamente cortas
- La revisión deberá ser grabada y almacenada



TIPOS DE REVISIONES

Review type	Principal purpose
Design or program inspections	To detect detailed errors in the design or code and to check whether standards have been followed. The review should be driven by a checklist of possible errors.
Progress reviews	To provide information for management about the overall progress of the project. This is both a process and a product review and is concerned with costs, plans and schedules.
Quality reviews	To carry out a technical analysis of product components or documentation to find faults or mismatches between the specification and the design, code or documentation. It may also be concerned with broader quality issues such as adherence to standards and other quality attributes.



INSPECCIONES DEL PROGRAMA

Clase de falla	Comprobación de inspección
Fallas de datos	<ul style="list-style-type: none">• ¿Todas las variables del programa se inician antes de usar sus valores?• ¿Todas las constantes tienen nombre?• ¿La cota superior de los arreglos es igual al tamaño del arreglo o Valor - 1?• Si se usan cadenas de caracteres, ¿se asigna explícitamente un delimitador?• ¿Existe alguna posibilidad de desbordamiento de buffer?
Fallas de control	<ul style="list-style-type: none">• Para cada enunciado condicional, ¿la condición es correcta?• ¿Hay certeza de que termine cada ciclo?• ¿Los enunciados compuestos están correctamente colocados entre paréntesis?• En caso de enunciados, ¿se justifican todos los casos posibles?• Si después de cada caso en los enunciados se requiere un paréntesis, ¿éste se incluyó?
Fallas de entrada/salida	<ul style="list-style-type: none">• ¿Se usan todas las variables de entrada?• ¿A todas las variables de salida se les asigna un valor antes de que se produzcan?• ¿Entradas inesperadas pueden causar corrupción?
Fallas de interfaz	<ul style="list-style-type: none">• ¿Todas las llamadas a función y método tienen el número correcto de parámetros?• ¿Los tipos de parámetro formal y real coinciden?• ¿Los parámetros están en el orden correcto?• Si los componentes acceden a memoria compartida, ¿tienen el mismo modelo de estructura de memoria compartida?
Fallas de gestión de almacenamiento	<ul style="list-style-type: none">• Si se modifica una estructura vinculada, ¿todos los vínculos se reasignan correctamente?• Si se usa almacenamiento dinámico, ¿el espacio se asignó correctamente?• ¿El espacio se cancela explícitamente después de que ya no se requiere?
Fallas de gestión de excepción	<ul style="list-style-type: none">• ¿Se tomaron en cuenta todas las posibles condiciones de error?

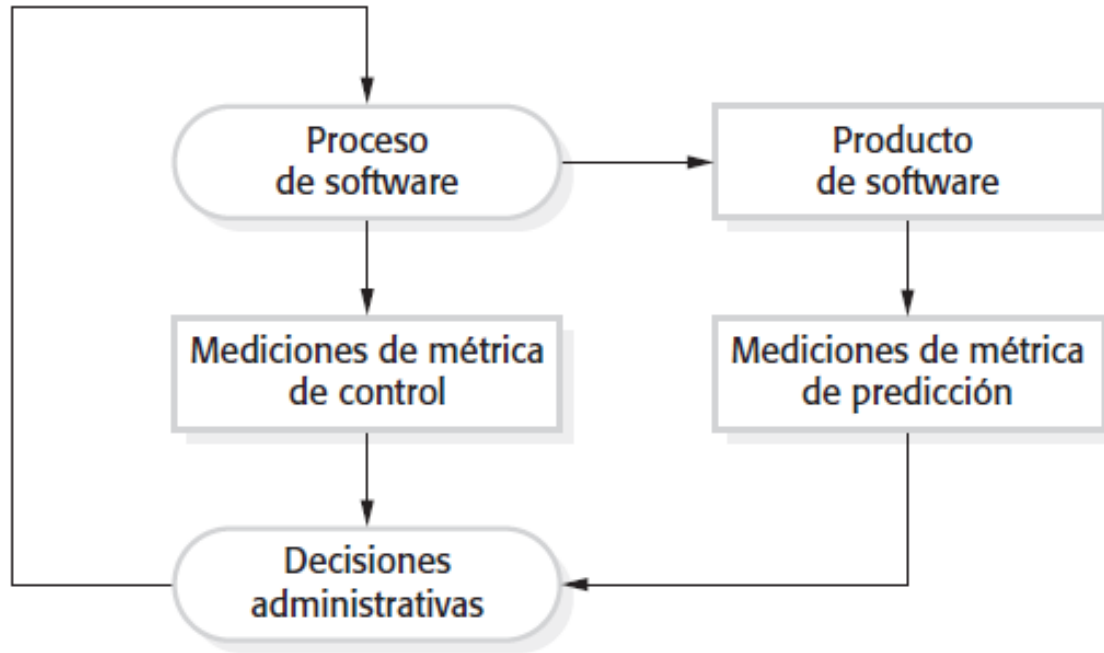


MÉTRICAS DE LA CALIDAD DEL PRODUCTO

- Una métrica de calidad deberá ser una forma de predicción de la calidad del producto
- La mayoría de las métricas de calidad existentes son las métricas de la calidad del diseño las cuales se relacionan con la medición del acoplamiento o la complejidad del diseño



MÉTRICAS DE CONTROL Y PREDICCIÓN



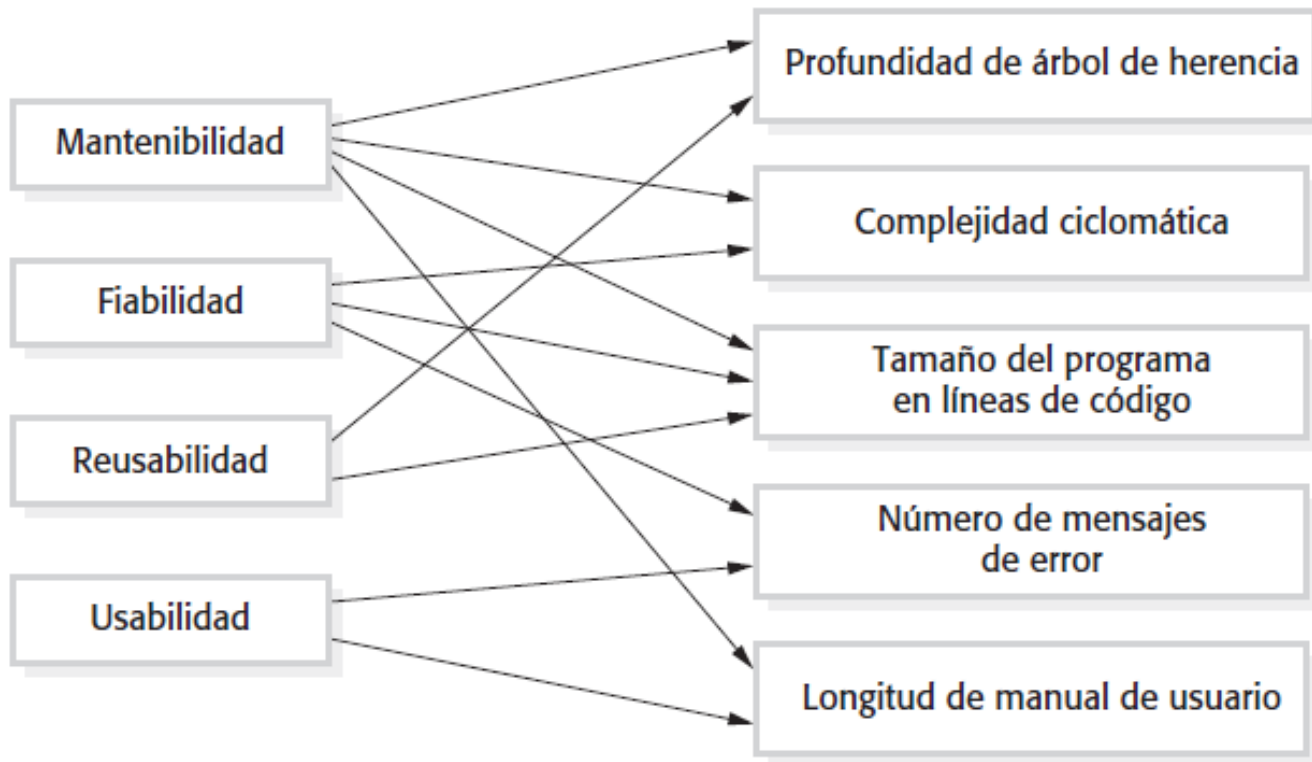
SUPOSICIÓN DE MÉTRICAS

- Una propiedad del software puede ser medida
- Existe una relación entre lo que se puede medir y que se quiere conocer
- Esta relación ha sido formalizada y validada
- Puede ser difícil relacionar que puede ser medido en cuanto a atributos deseables de calidad



ATRIBUTOS INTERNOS Y EXTERNOS

Atributos de calidad externos



MÉTRICAS ESTÁTICAS DE PRODUCTOS DE SOFTWARE

Métrica de software	Descripción
Fan-in/Fan-out	Fan-in (abanico de entrada) es una medida del número de funciones o métodos que llaman a otra función o método (por ejemplo, X). Fan-out (abanico de salida) es el número de funciones a las que llama la función X. Un valor alto para fan-in significa que X está estrechamente acoplado con el resto del diseño y que los cambios a X tendrán extensos efectos dominó. Un valor alto de fan-out sugiere que la complejidad global de X puede ser alta debido a la complejidad de la lógica de control necesaria para coordinar los componentes llamados.
Longitud de código	Ésta es una medida del tamaño de un programa. Por lo general, cuanto más grande sea el tamaño del código de un componente, más probable será que el componente sea complejo y proclive a errores. Se ha demostrado que la longitud del código es una de las métricas más fiables para predecir la proclividad al error en los componentes.
Complejidad ciclomática	Ésta es una medida de la complejidad del control de un programa. Tal complejidad del control puede relacionarse con la comprensibilidad del programa. En el capítulo 8 se estudia la complejidad ciclomática.
Longitud de identificadores	Ésta es una medida de la longitud promedio de los identificadores (nombres para variables, clases, métodos, etcétera) en un programa. Cuanto más largos sean los identificadores, es más probable que sean significativos y, por ende, más comprensible será el programa.
Profundidad de anidado condicional	Ésta es una medida de la profundidad de anidado de los enunciados if en un programa. Los enunciados if profundamente anidados son difíciles de entender y proclives potencialmente a errores.
Índice Fog	Ésta es una medida de la longitud promedio de las palabras y oraciones en los documentos. Cuanto más alto sea el valor del Índice Fog de un documento, más difícil será entender el documento.

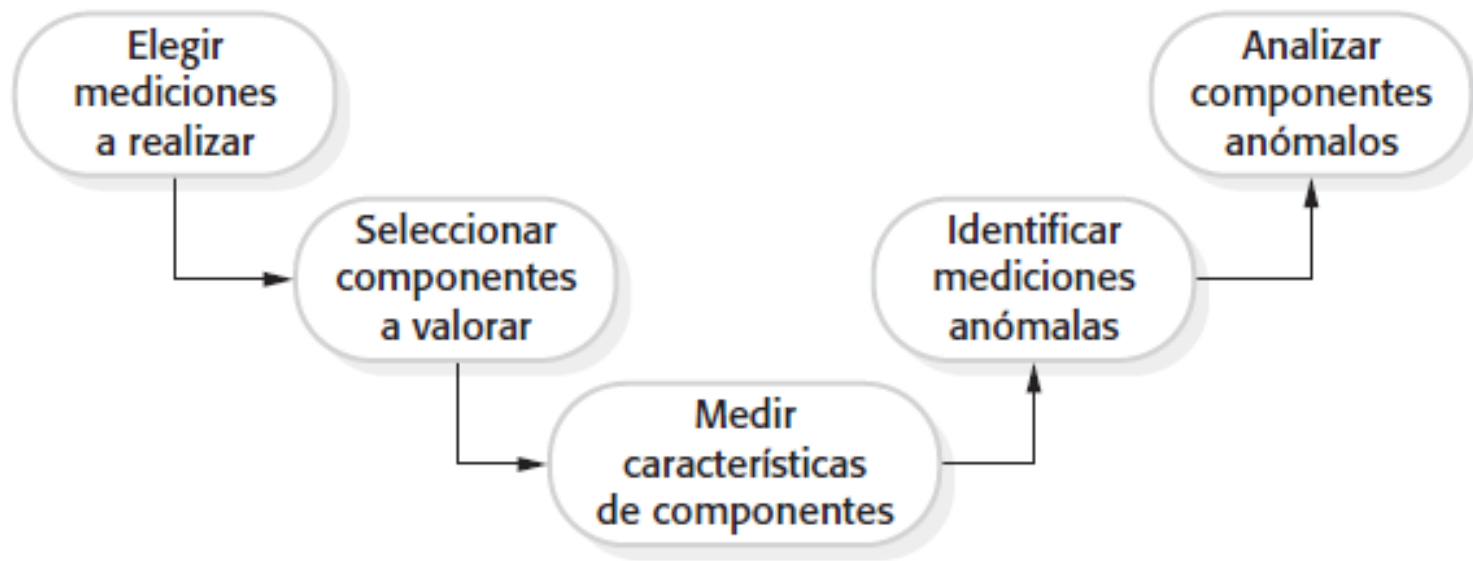


MÉTRICAS ORIENTADAS

Métrica orientada a objetos	Descripción
Métodos ponderados por clase (<i>weighted methods per class</i> , WMC)	Este es el número de métodos en cada clase, ponderado por la complejidad de cada método. Por lo tanto, un método simple puede tener una complejidad de 1, y un método grande y complejo tendrá un valor mucho mayor. Cuanto más grande sea el valor para esta métrica, más compleja será la clase de objeto. Es más probable que los objetos complejos sean más difíciles de entender. Tal vez no sean lógicamente cohesivos, por lo que no pueden reutilizarse de manera efectiva como superclases en un árbol de herencia.
Profundidad de árbol de herencia (<i>depth of inheritance tree</i> , DIT)	Esto representa el número de niveles discretos en el árbol de herencia en que las subclases heredan atributos y operaciones (métodos) de las superclases. Cuanto más profundo sea el árbol de herencia, más complejo será el diseño. Es posible que tengan que comprenderse muchas clases de objetos para entender las clases de objetos en las hojas del árbol.
Número de hijos (<i>number of children</i> , NOC)	Esta es una medida del número de subclases inmediatas en una clase. Mide la amplitud de una jerarquía de clase, mientras que DIT mide su profundidad. Un valor alto de NOC puede indicar mayor reutilización. Podría significar que debe realizarse más esfuerzo para validar las clases base, debido al número de subclases que dependen de ellas.
Acoplamiento entre clases de objetos (<i>coupling between object classes</i> , CBO)	Las clases están acopladas cuando los métodos en una clase usan los métodos o variables de instancia definidos en una clase diferente. CBO es una medida de cuánto acoplamiento existe. Un valor alto para CBO significa que las clases son estrechamente dependientes y, por lo tanto, es más probable que el hecho de cambiar una clase afecte a otras clases en el programa.
Respuesta por clase (<i>response for a class</i> , RFC)	RFC es una medida del número de métodos que potencialmente podrían ejecutarse en respuesta a un mensaje recibido por un objeto de dicha clase. Nuevamente, RFC se relaciona con la complejidad. Cuanto más alto sea el valor para RFC, más compleja será una clase y, por ende, es más probable que incluya errores.
Falta de cohesión en métodos (<i>lack of cohesion in methods</i> , LCOM)	LCOM se calcula al considerar pares de métodos en una clase. LCOM es la diferencia entre el número de pares de método sin compartir atributos y el número de pares de método con atributos compartidos. El valor de esta métrica se debate ampliamente y existe en muchas variaciones. No es claro si realmente agrega alguna información útil además de la proporcionada por otras métricas.



PROCESO DE MEDICIÓN DEL PRODUCTO



MANTENIBILIDAD DEL DISEÑO

- Cohesión

- Cuanto están relacionadas las partes forman un componente.

- Acoplamiento

- Que tan independiente es un componente

- Entendibilidad

- Que tan fácil es entender las funciones de un componente

- Adaptabilidad

- Que tan fácil es cambiar un componente



MÉTRICAS DE ACOPLAMIENTO

- Asociado con la medida 'fan-in y fan-out'.
- Alta entrada (fan-in)- implica un alto acoplamiento debido a las dependencias de los módulos.
- Alta salida (fan-out)- implica un alto acoplamiento debido a la complejidad del control



MÉTRICAS DE CALIDAD EN UN PROGRAMA

- Las métricas de diseño son también aplicables a programas
- Otras métricas incluyen
 - Longitud. El tamaño del código fuente del programa
 - Complejidad ciclomatica. La complejidad de control de un programa
 - Longitud de identificadores
 - Profundidad de condicionales anidados
- Los valores anómalos de las métricas sugieren que un componente presenta defectos o es difícil de entender



CONSIDERACIONES PARA LAS MÉTRICAS

- La longitud del código es simple pero la experimentación ha sugerido que representa un buen predictor de problemas
- La complejidad ciclomatica puede ser engañosa
- Nombres largos deberán incrementar la entendibilidad de un programa
- Condicionales profundamente anidadas son difíciles de entender.



MÉTRICAS DE COMPLEJIDAD

La complejidad ciclomatica es una medida de cuanto una estructura es compleja:

M = Complejidad ciclomática.

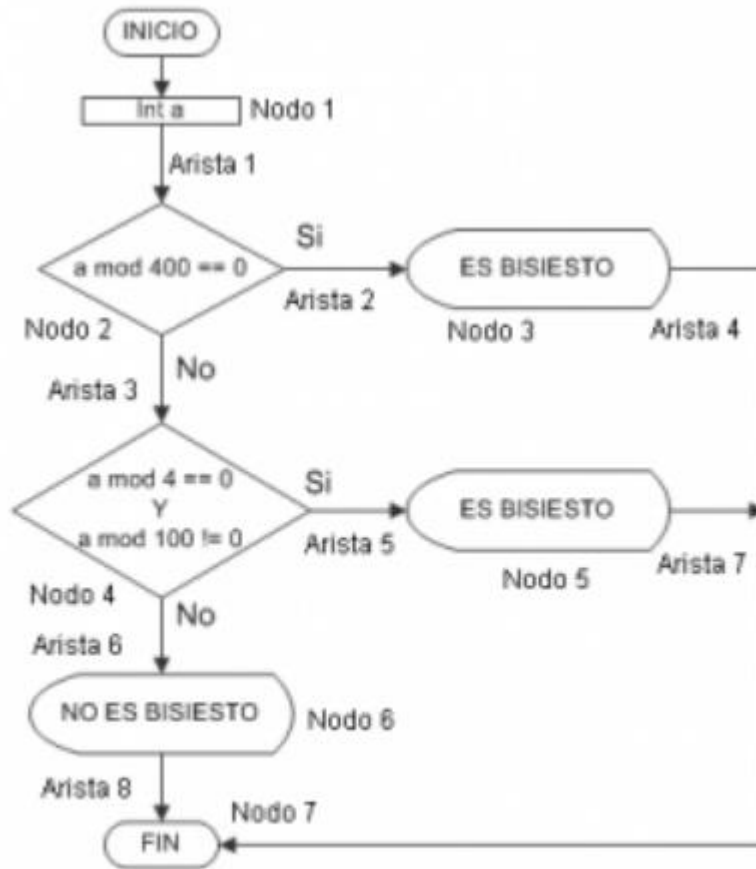
A = Número de arcos del grafo.

N = Número de nodos del grafo

$$M = A - N + 2$$



COMPLEJIDAD CICLOMATICA



$$M=8-7+2$$

Grafo numerando nodos y aristas

MÉTRICAS DE CALIDAD DE LA DOCUMENTACIÓN

- La legibilidad es importante en la documentación
- El índice “Gunnings Fog” es una medida de la legibilidad
 - Basada en la longitud de las frases y el numero de sílabas en una palabra
- Esto puede causar malas interpretaciones cuando se aplica a la documentación técnica.



MADURES DE LAS MÉTRICAS

- Las métricas todavía tienen un valor limitado y no ampliamente aceptado
- Las relaciones entre lo que se puede medir y lo que se quiere conocer no está bien comprendido aun
- Hace falta poner de acuerdo a las organizaciones sobre las métricas necesarias en el proceso de software

