

Ingeniería de Software I

Calidad

Jocelyn Simmonds

Departamento de Ciencias de la Computación

- Queremos construir software de alta calidad ...
 - ¿Pero qué significa **calidad**?
- Calidad es un concepto complejo y multifacético
 - Puede describirse desde diversas perspectivas

Visiones de calidad

- Visión trascendental
 - puede ser reconocida pero no definida
- Visión del usuario
 - grado de adecuación al propósito
- Visión del productor
 - conformidad con la especificación
- Visión del producto
 - ligada a características inherentes del mismo
- Visión basada en valor
 - ¿cuánto está dispuesto a pagar el cliente?

Visiones de Calidad – Trascendental

- Ideal filosófico de calidad
- “Productos que deleitan a nuestros clientes”
- “El grado o standard de excelencia”
 - subjetivo
 - no se puede analizar
 - no se puede medir
- No es apropiada para software

Visiones de Calidad – Del usuario

- Software que satisface las necesidades de los usuarios
- Se basa en tareas del usuario
 - a veces muy dependiente del contexto
- Apoyado por
 - modelos de confiabilidad
 - modelos de desempeño
 - laboratorios de **usabilidad**

Visiones de Calidad – Del productor

- Foco en el proceso de software
 - énfasis en **right first time**
 - minimizar el re-trabajo antes y después del **release**
- Evaluación de calidad virtualmente independiente del producto
 - CMMI
 - ISO 9001
- **Conjetura base**: La calidad del producto es un efecto **lateral** de la calidad del proceso

Visiones de Calidad – Del producto

- Características internas determinan la calidad
 - componentes de alta calidad → productos de alta calidad
- Desde el punto de vista de las métricas
 - control de las propiedades internas del producto
p.ej. tamaño, complejidad, cobertura de pruebas
 - resulta en mejoras en la calidad del producto
- Problema
 - poca evidencia empírica que pruebe que las mediciones internas afectan las características externas

Visión basada en valor

- Visión de jefes de proyecto
 - balancear requisitos con presupuestos y plazos
 - manejar conflictos originados en cambios en los requisitos
 - apoyar ciertas estrategias de desarrollo
 - diseñar al costo
 - **adecuación** de los requisitos
- Visión del comprador
 - valor por el dinero para una organización particular
 - se basa en evaluación de beneficios del producto
 - independiente de características internas del producto

Medición de calidad

- Las visiones de calidad afectan la definición
- Definiciones afectan las mediciones
- Definición ISO
 - **calidad**: la totalidad de características de un producto de software que determinan su capacidad para satisfacer necesidades establecidas e implícitas
- Software es un concepto nuevo
 - normalmente, el negocio no sabe de software
 - muchas veces, la gente de software no comprende el negocio
 - luego, hacer todo explícito!

¿Cualidades de un software de calidad?

¿Qué creen ustedes?

Algunas cualidades ...

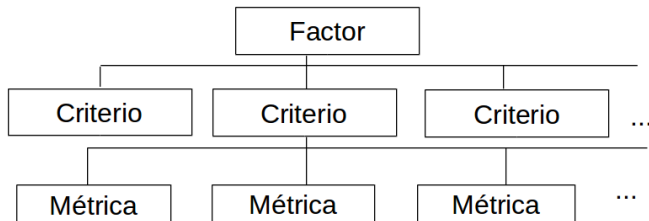
A tiempo
Completo
Correcto
De confianza
Robusto
Eficiente
Usable
Interoperable
Seguro
Escalable

Verifiable
Reutilizable
Portátil
Visible
Productivo
Modular
Comprensible
Mantenible
Documentado
Libre de errores

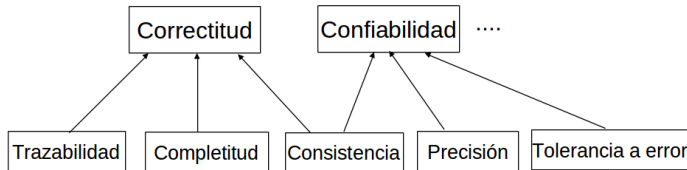
Modelos de Calidad

Modelos de calidad

- Modelos de calidad intentan reconciliar visiones
- Normalmente, identifican calidad de producto importante para los usuarios
- Ligadas a características y atributos externos



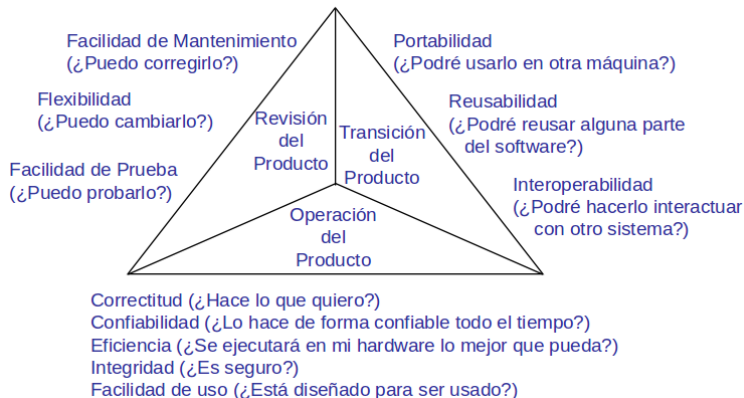
Modelos de Calidad – Factores & criterios



- Un criterio puede afectar más de un factor
- Problemas
 - ¿cómo se sabe cuándo un modelo está completo?
 - ¿cómo se decide qué criterios afectan qué factor?
 - ¿cómo se decide qué es factor y qué es criterio?

Modelo de calidad de McCall (1/2)

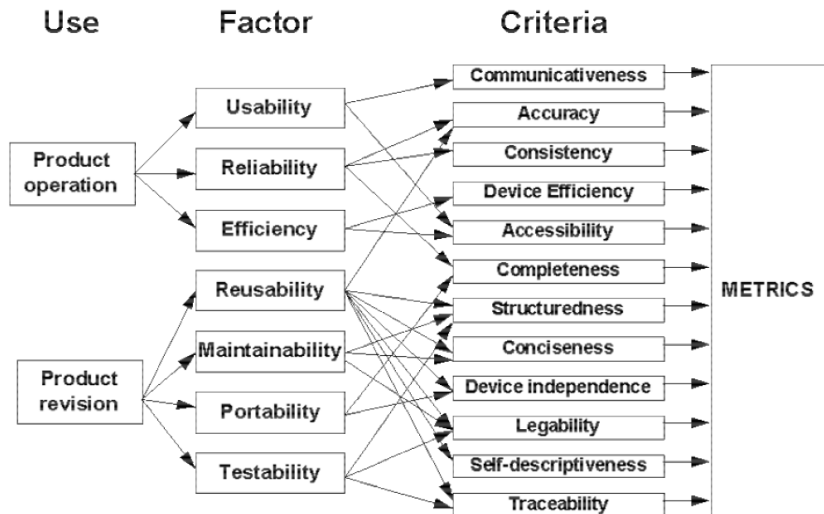
Factores de calidad de software de McCall



- Modelo antiguo (1978), pero muy influyente!
- 11 factores de calidad, 23 criterios de calidad

Modelo de calidad de McCall (2/2)

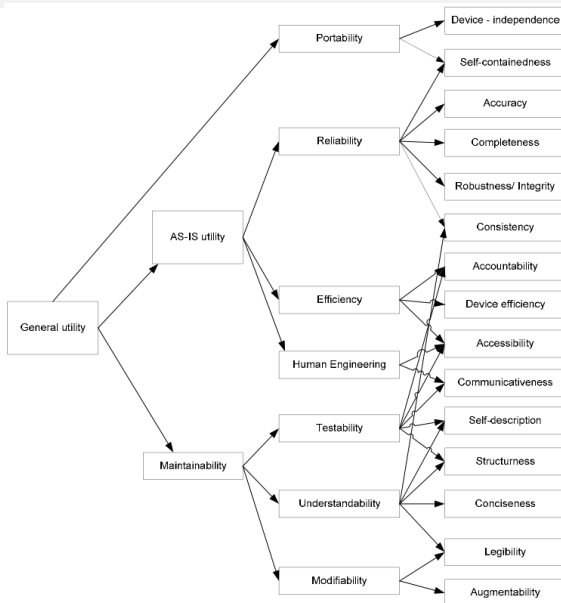
Ejemplo de un modelo McCall de factores, criterios y métricas



Modelo de Calidad de Boehm (1978) (1/2)

- Una extensión del modelo de McCall, con énfasis en la mantención
- Define **característica** de calidad: una característica de calidad es un conjunto de atributos de un producto de software a través del cual su calidad es descrita y evaluada.
- Una característica de calidad de software puede ser refinada en múltiples niveles de sub-características.
- Este modelo considera tres características de calidad de alto nivel:
 - Portability: ¿Se puede seguir usando el SW si cambia el ambiente?
 - AS-IS utility: ¿Qué tan bien se puede usar el SW?
 - Maintainability: ¿Cuán fácil/difícil es hacerle mantención al SW?

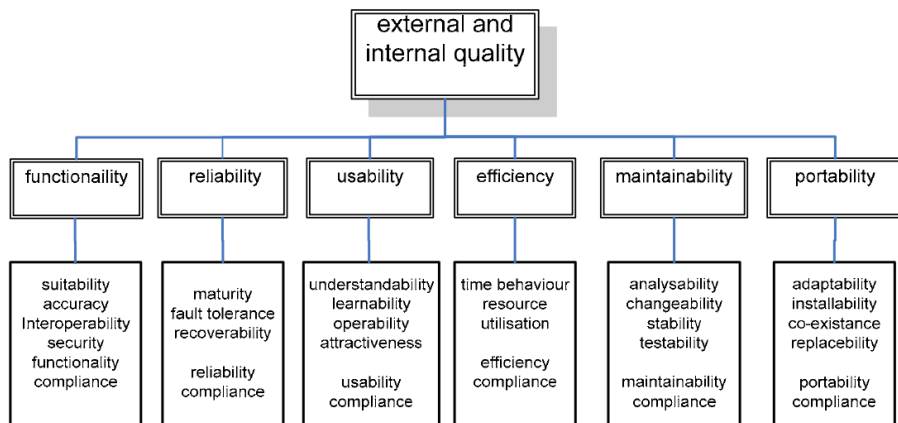
Modelo de Calidad de Boehm (2/2)



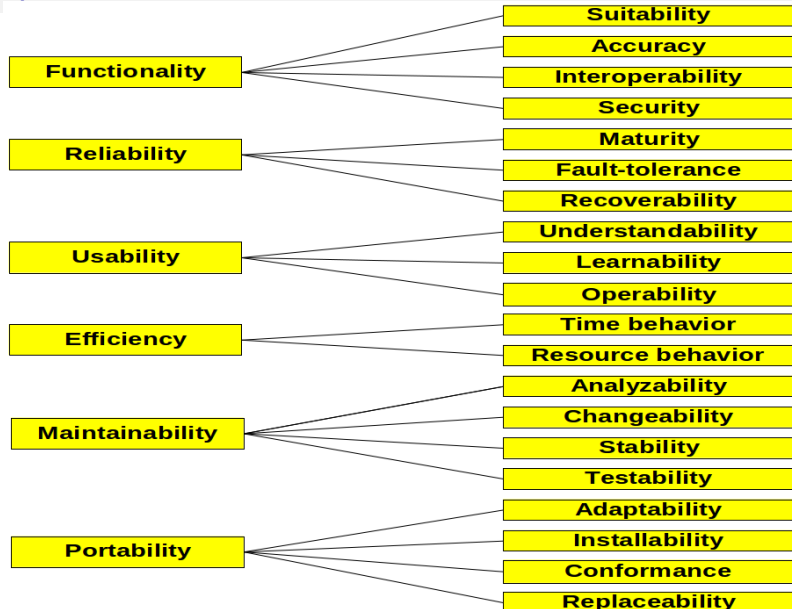
Norma ISO 9126

- Norma ISO/IEC 9126-1 (2001)
- Estándar reciente para calidad de productos de software
- 6 características básicas, descompuestas en 20 subcaracterísticas (cada una se relaciona con solo una característica)
 - funcionalidad
 - confiabilidad
 - usabilidad
 - eficiencia
 - mantenibilidad
 - portabilidad
- Refleja una visión de usuario, mientras el modelo de McCall refleja una visión de producto

Norma ISO 9126 - subcaracterísticas



Ejemplo de uso de la Norma ISO 9126



Algunos problemas:

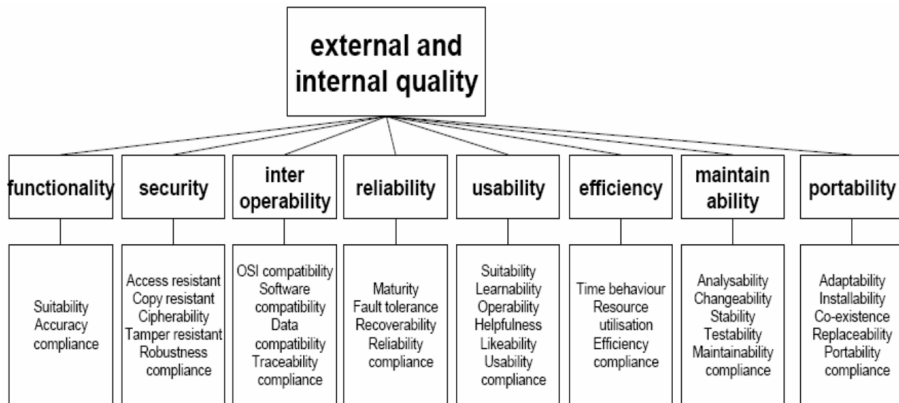
- No es claro si el modelo es completo
 - no incluye reusabilidad
 - mantenibilidad abarca mejoras y corrección de defectos
- El proceso de selección y medición de métricas puede ser demasiado subjetivo
 - no especifica como seleccionar métricas
 - ni como deben ser medidas
- No es claro si algunas sub-características deben ser características
 - interoperabilidad v/s portabilidad . . .
 - tampoco es claro el mapeo entre características y métricas

El modelo SQuaRE

Software Product Quality Requirements and Evaluation:

- Creado para subsanar problemas con la ISO 9126
- Fusión de varios estándares ISO existentes:
 - ISO 9126
 - ISO 14598 Software Engineering - Product evaluation
 - ISO 15939 Software Engineering - Software Measurement Process
- Incluye ejemplos para orientar el uso práctico del modelo
- Serie de documentos, agrupados en cinco temáticas:
 - Quality Management: modelos, términos, definiciones comunes
 - Quality Model: versión actualizada de ISO 9126
 - Quality Measures: fusión de ISO 9126 y ISO 15939
 - Quality Requirements: estándar para apoyar la especificación de requisitos de calidad
 - Quality Evaluation: recomendaciones y directrices para la evaluación de la calidad de productos de software

El modelo SQuaRE



Ejemplo: Calidad de una Arquitectura

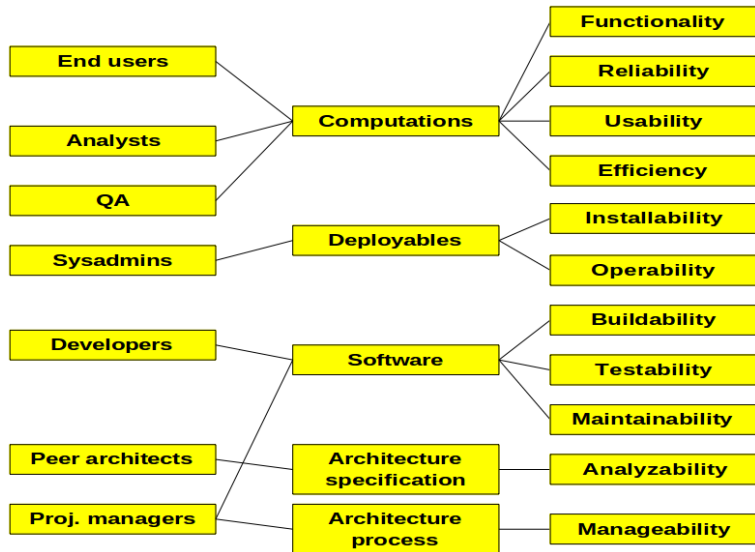
- La solución (computaciones) debe ...
 - Satisfacer los requisitos (según analistas)
- La solución (ejecutables) debe ...
 - Ser administrable (según administradores)
- La solución (software) debe ...
 - Ser “construible” (según programadores & diseñadores)
 - Ser “testable” (según QA)
- La descripción de la solución debe ...
 - Ser evaluable (según otros arquitectos)
- El proceso de construcción debe ...
 - Ser manejable (según jefe de proyecto)

Calidad de Arquitectura (5 niveles)

Es útil distinguir entre los productos del proceso

- Computaciones
 - sistema en ejecución (colección de programas colaborando)
- Ejecutables
 - colección de artefactos instalables/entregables
 - binarios, archivos de configuración run-time ...
- Software
 - programas, configuración build-time
- Especificación de software
 - descripciones de software
- Proceso de arquitectura

Calidad de Arquitectura (5 niveles)



- La calidad es un concepto complejo
 - significa cosas distintas para personas distintas
 - no hay medición única y simple
- Costos
 - de la calidad: prevención y evaluación
 - de la no-calidad: re-trabajo

Calidad: Reflexiones

- Importante que los beneficios superen a los costos
 - “Lo perfecto es enemigo de lo bueno”
- Implicancia: necesidad de invertir para tener ganancias

