



ISO 9126 vs. SQuaRE

José Joaquín Ruiz Morilla

Calidad y Medición de Sistemas de Información

INDICE

<i>Conclusiones</i>	18
<i>Crítica / Valoración</i>	13
<i>Estado del Arte</i>	4
<i>Introducción</i>	3
<i>Tendencias futuras</i>	19

1. INTRODUCCION

El objetivo de este trabajo es realizar un análisis de las diferencias entre las normas ISO 9126 y SQuaRE. Para ello, iremos realizando una descripción de la evolución que ha tenido cada una de ellas, hasta finalmente realizar un completo análisis entre ellas. ISO 9126 se publicó en 1991 con el objeto de “promover un entorno que permita la evaluación de la calidad del software”. En 1994 se entendió que era necesaria una modificación y adaptación de la norma. En esta versión es dónde se introducen por primera vez los conceptos de calidad interna y calidad externa. Además se creó una nueva norma (ISO 14598) que asumía el modelo del proceso de evaluación antes incluido en la norma ISO 9126. La norma ISO 9126 está dividida en cuatro partes:

- ISO 9126-1. Modelo de calidad.
- ISO 9126-2. Métricas externas.
- ISO 9126-3. Métricas internas.
- ISO 9126-4. Calidad en las métricas de uso. (no la trataremos en este documento)

Sólo la primera parte, ISO 9126-1, es un estándar aprobado y publicado, siendo el resto de partes de la norma, informes que se encuentran en la fase llamada Technical Report(TR).

El modelo SQuaRE (Software Product Quality Requirements and Evaluation) es una revisión de ISO/IEC 9126-1:2001, y conserva las mismas características de calidad de software. Las series SQuaRE tiene las siguientes divisiones

- ISO/IEC 2500n. División de dirección de calidad.
- ISO/IEC 2501n. División del modelo de calidad.
- ISO/IEC 2502n. División de medida de calidad.
- ISO/IEC 2503n. División de requisitos de calidad.
- ISO/IEC 2504n. División de evaluación de calidad.

Este modelo será usado en conjunción con las otras partes de las series SQuaRE (ISO/IEC 25000 – ISO/IEC 25050), y con ISO/IEC 14598 hasta que sea reemplazado por las series SQuaRE. SQuaRE se dedica solamente a la calidad del producto software. La división de dirección de calidad trata con productos software, y está separada de la dirección de calidad de los procesos que son definidos en la familia ISO 9000.

SQuaRE nace con el objetivo de responder a las necesidades de los usuarios a través de un conjunto de documentos unificados cubriendo tres procesos de calidad complementarios: especificación de requisitos, medidas y evaluación. Por lo tanto, SQuaRE se creó para satisfacer una serie de necesidades que existían (y comentaremos más adelante) con la ISO 9126 y la ISO/IEC 14598, donde ellos (ISO 9126 y ISO/IEC 14598) pertenecen a la primera generación de estándares de calidad de un producto software. Por consiguiente, SQuaRE pertenece a la segunda generación de calidad de un producto software.

Nos centraremos en la división 2501n que presenta un detallado modelo de calidad incluyendo características de calidad interna, externa y calidad en uso. Además de proporcionar una guía sobre el uso del modelo de calidad.

2. ESTADO DEL ARTE

Como ya comentamos en la introducción, la ISO 9126 se dividía en cuatro partes (de las cuales veremos las tres primeras): modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso. Comenzaremos a describir la primera parte, la **ISO 9126-1**. Esta norma define un modelo de calidad basado en dos partes bien identificadas:

- Calidad interna y externa.
- Calidad de uso.

La calidad interna, entendida como la “totalidad de las características del producto software desde un punto de vista interno”, y la calidad externa definida como la “totalidad de las características de producto software desde un punto de vista externa” influyen en la calidad del proceso, al mismo tiempo que la calidad de uso influye sobre las anteriores. La calidad interna, externa y de uso están relacionadas, una se sustenta en la otra como capas sucesivas. La calidad del proceso influye en la calidad del producto que a su vez es relevante en la calidad de uso (ver figura 1).

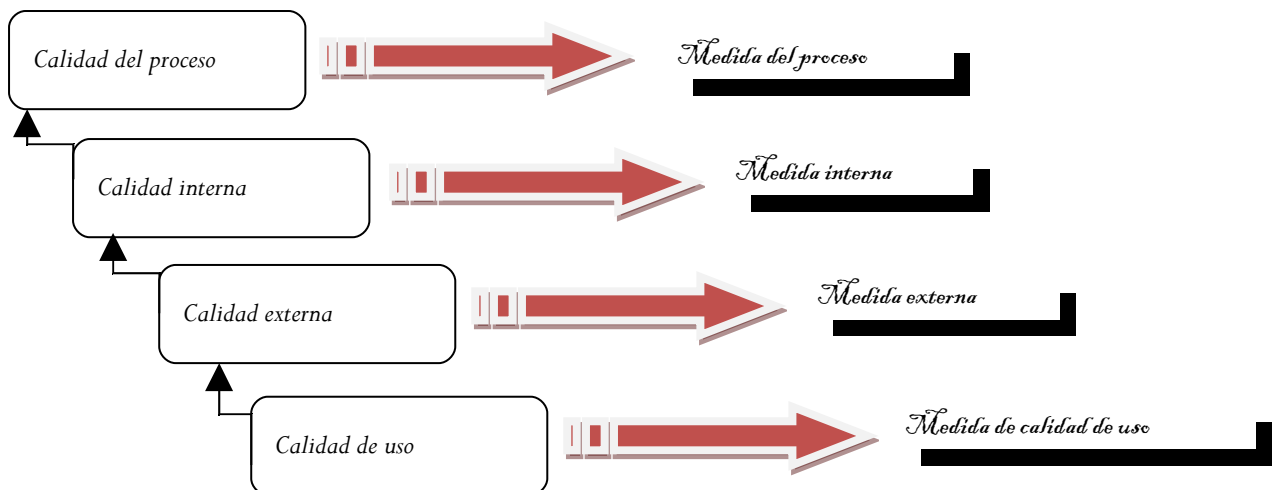


Figura 1: Calidad en el ciclo de vida

El proceso de calidad contribuye a mejorar la calidad del producto, y la calidad del producto contribuye a mejorar la calidad de uso. Evaluando la calidad en uso de un producto puede proporcionar una respuesta a mejorar un producto, y evaluando un producto puede proporcionar una respuesta para mejorar un proceso.

En las etapas tempranas del desarrollo solo los recursos y el proceso pueden ser medidos. Cuando los productos intermedios (código fuente, especificación,...) estén disponibles, pueden ser evaluados por niveles de métricas internas. Estas métricas pueden usarse para predecir los valores de las métricas externas.

La calidad de un producto software debería ser evaluado usando un modelo de calidad. ISO 9126-1 propone un modelo de calidad categorizando la calidad de los atributos software en seis características (funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad), las cuales son subdivididas en subcaracterísticas. Las subcaracterísticas pueden ser medidas con métricas internas o externas.

<i>Calidad del software (interna y externa)</i>					
<i>Funcionalidad</i>	<i>Fiabilidad</i>	<i>Usabilidad</i>	<i>Eficiencia</i>	<i>Mantenibilidad</i>	<i>Portabilidad</i>
<i>Adecuación</i>	<i>Madurez</i>	<i>Fácil comprensión</i>	<i>Comportamiento o frente al tiempo</i>	<i>Facilidad de análisis</i>	<i>Adaptabilidad</i>
<i>Exactitud</i>	<i>Tolerancia a fallos</i>	<i>Fácil aprendizaje</i>	<i>Uso de recursos</i>	<i>Capacidad para cambios</i>	<i>Facilidad de instalación</i>
<i>Interoperatividad</i>	<i>Capacidad de recuperación</i>	<i>Operatividad</i>	<i>Adherencia a normas</i>	<i>Estabilidad</i>	<i>Coexistencia</i>
<i>Seguridad</i>	<i>Adherencia a normas</i>	<i>Software atractivo</i>		<i>Facilidad para pruebas</i>	<i>Facilidad de reemplazo</i>
<i>Adherencia a normas</i>		<i>Adherencia a normas</i>		<i>Adherencia a normas</i>	<i>Adherencia a normas</i>

Figura 2: Calidad interna y externa

Para cada característica y subcaracterística, la capacidad de el software es determinada por un conjunto de atributos internos que pueden ser medidos (ISO 9126-3). Además, las características y subcaracterísticas pueden ser medidas externamente (ISO 9126-2).

A continuación pasamos a describir cada una de las características y subcaracterísticas:

Funcionalidad. Se define como un conjunto de atributos que atañen a la existencia de un conjunto de funciones y sus propiedades específicas. Estas funciones son las que satisfacen las necesidades implícitas y establecidas. Esta característica del software puede ser desglosada en varias características:

- Adecuación. Capacidad del software de proporcionar un conjunto apropiado de funciones para tareas específicas y objetivos del usuario.
- Exactitud. Capacidad del software para proporcionar resultados correctos o que necesitan un determinado grado de precisión.
- Interoperatividad. Capacidad del software de interactuar con uno o más sistemas especificados.
- Seguridad. Capacidad del software de proteger la información y los datos.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes.

Fiabilidad. Conjunto de atributos que atañen a la capacidad del software para mantener su nivel de prestación bajo condiciones establecidas durante un tiempo establecido. Se descompone en las siguientes características:

- Madurez. Capacidad del software para evitar fallos como resultados de defectos del software
- Tolerancia a fallos. Capacidad del software para mantener un nivel especificado de rendimiento en casos de fallos del software
- Capacidad de recuperación. Capacidad para restablecer el nivel de rendimiento y de recuperación de datos afectados directamente en el caso de un fallo.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes en leyes o prescripciones similares

Facilidad de uso. Capacidad del producto software de ser entendido, aprendido, usado y atraer al usuario, cuando es utilizado bajo ciertas condiciones específicas. Se descompone en:

- Fácil comprensión. La capacidad del software que permite al usuario si el producto es aceptable y cómo puede ser usado para tareas particulares y determinadas condiciones de uso.
- Fácil aprendizaje. Capacidad del producto software que permite al usuario aprender la aplicación software
- Operatividad. Capacidad del producto software que permite al usuario controlar y usar la aplicación software
- Software atractivo. Capacidad del producto software de ser atractivo al usuario.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes en leyes o prescripciones similares.

Eficiencia. Capacidad del producto software para proporcionar un rendimiento apropiado relacionado con el total de recursos utilizados bajo condiciones establecidas. Se subdivide en las siguientes características:

- Comportamiento frente al tiempo. Capacidad del producto software para proporcionar una respuesta y un tiempo de procesamiento apropiados al desarrollar sus funciones bajo condiciones establecidas.
- Uso de recursos. Capacidad del producto software para utilizar un apropiado número de recursos y tiempo de ejecución cuando el software desarrolla sus funciones bajo condiciones establecidas.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes en leyes o prescripciones similares.

Mantenibilidad. Capacidad del producto software para ser modificado. Se descompone en las siguientes características:

- Facilidad de análisis. Capacidad del producto software para diagnosticar deficiencias o causas de fallos en el software
- Capacidad para cambios. Capacidad del producto software que permite la ejecución de una modificación específica en ella misma.
- Estabilidad. Capacidad del producto de software para evitar defectos no esperados debidos a modificaciones en el mismo.
- Facilidades para pruebas. Capacidad del producto software que permite al software que ha sido modificado ser evaluado.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes en leyes o prescripciones similares.

Portabilidad. Capacidad del producto software para ser transferido de un entorno a otro. El entorno se interpreta tanto a nivel software y hardware, como aquel entorno relacionado con la organización. Se divide en:

- Adaptabilidad. Capacidad del producto software para ser adaptado a diferentes entornos especificados sin aplicar acciones alejadas de aquellas que el propio software proporcione.
- Facilidad de instalación. Capacidad del producto software para ser instalado en un entorno específico.
- Coexistencia. Capacidad del producto software de coexistir con otros programas independientes en un entorno común y compartiendo recursos también comunes.
- Facilidad de reemplazo. Capacidad del producto software de ser utilizado en lugar de otro producto software específico para el mismo propósito que éste y en un entorno similar.
- Adherencia a normas. Capacidad del software relacionada con el grado de conformidad con estándares, convenciones o regulaciones existentes en leyes o prescripciones similares.

Una vez explicado la parte de calidad interna y externa del modelo de calidad, nos adentraremos en la calidad de uso. La calidad de uso es definida como “la capacidad del software que posibilita la obtención de objetivos específicos con efectividad, productividad, satisfacción y seguridad”.

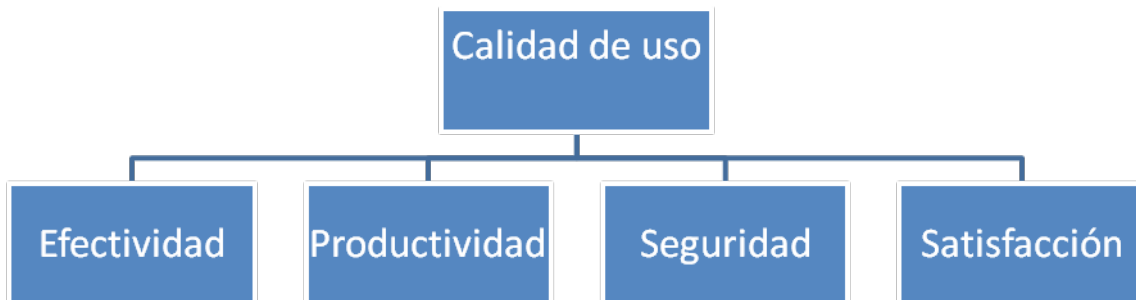


Figura 3: Calidad de uso.

Describimos a continuación las características (no contiene subcaracterísticas) de la calidad de uso:

Eficacia. Capacidad del software para permitir a los usuarios alcanzar objetivos específicos con precisión y completamente en un contexto específico de uso.

Productividad. Capacidad del producto software para permitir a los usuarios emplear recursos apropiados con relación a la eficacia alcanzada en un contexto específico de uso.

Seguridad. La capacidad del producto software para alcanzar niveles aceptables de riesgo hacia la gente, negocio, software, propiedad o medio ambiente, en un contexto específico de uso.

Satisfacción. La capacidad del producto software para satisfacer al usuario en un contexto específico de uso.

Las características en las que la norma ISO 9126-1 descompone la calidad son influidas por atributos internos y externos propios de dichas características. Los atributos internos son indicadores de los atributos externos. Un atributo interno puede influir a una o más características y una característica puede verse influida por uno o más atributos. Las características y subcaracterísticas son medidas, por tanto, a través de sus correspondientes atributos.

La norma define las métricas internas como aquellas medidas que se realizan sobre un producto software no ejecutable, tal como la norma indica “un producto software intermedio debería ser evaluado usando métricas internas”. Las métricas externas son medidas del producto software obtenidas del comportamiento del sistema en la fase de ejecución del mismo. Las métricas de la

calidad del uso, como tercer gran concepto propuesto por la norma, miden la extensión en la que un producto alcanza las necesidades expuestas por el usuario de forma específica en relación a los objetivos de efectividad, seguridad, productividad y satisfacción.

La norma **9126-2** es un reporte técnico que contiene la terminología relacionada con las medidas de las métricas, el uso de las métricas en el proceso del ciclo de la vida y unos conjuntos básicos introductorios de métricas externas para cada característica y subcaracterística de calidad de software. Este informe proporciona al usuario una guía (o dirección) de métricas para la evaluación de planificación, selección de métricas, diseño de métricas, aplicación de métricas y interpretación de medidas de datos.

Es recomendable que las métricas internas tengan fuerte relación con las métricas externas para que puedan ser usadas para predecir los valores de las métricas externas.

La interpretación de las medidas se puede realizar de tres formas:

- Medida directa. Una medida directa es una medida de un atributo que no depende de las medidas de otros atributos.
- Medida indirecta. Una medida indirecta es derivada de medidas de uno o más atributos.
- Indicadores. Son aquellas medidas que pueden ser estimadas o predichas desde otras medidas.

Las métricas tienen unas propiedades deseables que se detallan a continuación:

- Fiabilidad.
- Indicabilidad.
- Disponibilidad.
- Corrección.
- Imparcialidad.

El conjunto de métricas que contiene está organizadas por características y subcaracterísticas, donde cada métrica contiene:

- A. Nombre.
- B. Propósito.
- C. Método de aplicación.
- D. Medida, fórmula y cómputo de datos.
- E. Interpretación del valor medido.
- F. Tipo de escala.
- G. Tipo de medida.
- H. Fuente de medida.
- I. Referencia a ISO/IEC 12207 SLCP.
- J. Audiencia.

La norma **9126-3** proporciona métricas internas para medir los atributos de las características de calidad definidas en la norma 9126-1. Con las siguientes cualidades:

- Se aplican a un producto de software no ejecutable.
- Se aplican durante las etapas de desarrollo.
- Permiten medir la calidad de los entregables intermedios.
- Permiten predecir la calidad del producto final.
- Permiten al usuario iniciar acciones correctivas temprano en el ciclo de desarrollo.

El conjunto de métricas están organizadas igualmente por características y subcaracterísticas, donde tiene los mismos campos que la norma ISO 9126-2. Por lo tanto existirá métricas de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. Las propiedades

deseables son las siguientes: confiable, repetible, reproducible, disponible, indicable, correcta y con significado. Los pasos que se sugieren son los siguientes:

- A. Identificación de los requisitos de calidad.
- B. Especificación de la evaluación.
- C. Diseño de la evaluación.
- D. Ejecución de la evaluación.
- E. Retroalimentación a la organización.

Una vez descrita la ISO 9126, desarrollaremos el modelo **SQuaRE**. Se trata de una revisión de la ISO 9126-1, y tiene las mismas características de calidad de software. Hay dos aspectos importantes en el campo de la calidad del software, el producto y el proceso. SQuaRE se centra en el lado del producto. SQuaRE hereda el modelo de calidad de la ISO 9126-1. El modelo de ciclo de vida de la calidad del producto software se basa en la calidad del producto software en tres fases principales del ciclo de vida del producto software: producto bajo desarrollo, producto en operación y producto en uso. La fase de un producto bajo desarrollo está sujeto a la calidad interna del software. La fase del un producto en operación está sujeto a la calidad externa del software, y la fase de un producto en uso está sujeto de calidad de uso.

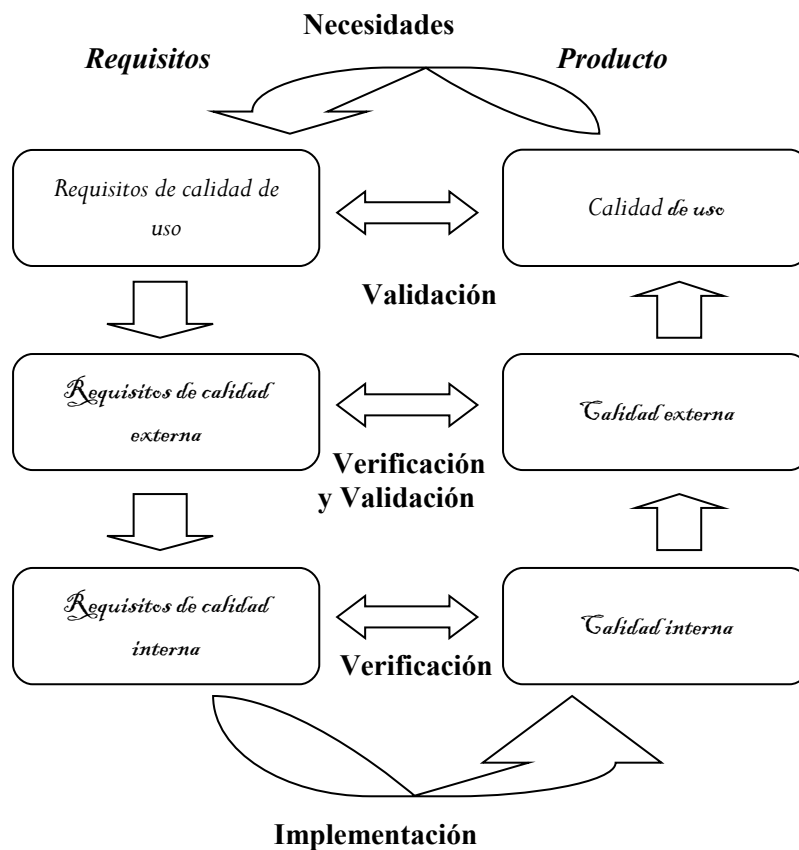


Figura 4: El modelo de ciclo de vida de la calidad del producto software.

El modelo de ciclo de vida de la calidad del producto software también indica que la implementación de la calidad del software requiere un proceso similar al desarrollo de un proceso de software para cada tipo de calidad: requisitos, implementación y validación de los resultados.

Los requisitos de calidad de uso especifican el requerido nivel de calidad desde el punto de vista del usuario final. Los requisitos de calidad de software externa especifican el requerido nivel de

calidad desde una vista externa. Y por último, los requisitos de calidad de software interno especifican el nivel de calidad requerida desde la vista interna de un producto.

El modelo de calidad SQuaRE categoriza la calidad del software en características, las cuales están subdivididas en subcaracterísticas y atributos de calidad.

En este punto es donde tenemos que mostrar un modelo de calidad distinto para cada una de las versiones de SQuaRE. Existen tres versiones de SQuaRE:

- ISO/IEC 25010 Mayo 2007.
- ISO/IEC 25010 Julio 2007.
- ISO/IEC 25010 Julio 2008 (última versión).

<i>Calidad del software (interna y externa)</i>							
<i>Funcionalidad</i>	<i>Seguridad</i>	<i>Interoperabilidad</i>	<i>Fiabilidad</i>	<i>Usabilidad</i>	<i>Eficiencia</i>	<i>Mantenibilidad</i>	<i>Portabilidad</i>
<i>Adecuación</i>	<i>Adherencia a normas</i>	<i>Adherencia a normas</i>	<i>Madurez</i>	<i>Comprensibilidad</i>	<i>Tiempo de respuesta</i>	<i>Capacidad de análisis</i>	<i>Adaptabilidad</i>
<i>Exactitud</i>			<i>Tolerancia a fallos</i>	<i>Capacidad de aprendizaje</i>	<i>Utilización de recursos</i>	<i>Capacidad a cambios</i>	<i>Capacidad a instalación</i>
<i>Adherencia a normas</i>			<i>Recuperabilidad</i>	<i>Operabilidad</i>	<i>Adherencia a normas</i>	<i>Estabilidad</i>	<i>Capacidad a coexistencia</i>
			<i>Adherencia a normas</i>	<i>Atractivo</i>		<i>Capacidad a testing</i>	<i>Adherencia a normas</i>
				<i>Adherencia a normas</i>		<i>Adherencia a normas</i>	

Figura 5: Modelo de calidad para la calidad interna y externa para la versión ISO/IEC 25010 Mayo 2007.

<i>Calidad del software (interna y externa)</i>							
<i>Funcionalidad</i>	<i>Seguridad</i>	<i>Interoperabilidad</i>	<i>Fiabilidad</i>	<i>Usabilidad</i>	<i>Eficiencia</i>	<i>Mantenibilidad</i>	<i>Portabilidad</i>
<i>Adecuación</i>	<i>Acceso resistente</i>	<i>Compatibilidad OSI</i>	<i>Madurez</i>	<i>Adecuación</i>	<i>Tiempo de respuesta</i>	<i>Capacidad de análisis</i>	<i>Adaptabilidad</i>
<i>Exactitud</i>	<i>Copia resistente</i>	<i>Compatibilidad Software</i>	<i>Tolerancia a fallos</i>	<i>Capacidad de aprendizaje</i>	<i>Utilización de recursos</i>	<i>Capacidad a cambios</i>	<i>Capacidad a instalación</i>
<i>Adherencia a normas</i>	<i>Cifrabilidad</i>	<i>Compatibilidad de datos</i>	<i>Recuperabilidad</i>	<i>Operabilidad</i>	<i>Adherencia a normas</i>	<i>Estabilidad</i>	<i>Capacidad a coexistencia</i>
	<i>Protección resistente</i>	<i>Trazabilidad</i>	<i>Adherencia a normas</i>	<i>Util</i>		<i>Capacidad a testing</i>	<i>Capacidad a reemplazo</i>
	<i>Robustez</i>	<i>Adherencia a normas</i>		<i>Agradable</i>		<i>Adherencia a normas</i>	<i>Adherencia a normas</i>
	<i>Adherencia a normas</i>			<i>Adherencia a normas</i>			

Figura 6: Modelo de calidad para la calidad interna y externa para la versión ISO/IEC 25010 Julio 2007.

<i>Calidad del software (interna y externa)</i>							
<i>Adecuación</i>	<i>Fiabilidad</i>	<i>Eficiencia</i>	<i>Operabilidad</i>	<i>Seguridad</i>	<i>Compatibilidad</i>	<i>Mantenibilidad</i>	<i>Transmisibilidad</i>

<i>n</i> <i>funcional</i>		<i>de</i> <i>rendimiento</i>			<i>dad</i>	<i>i-dad</i>	<i>lidad</i>
<i>Adecuación</i>	<i>Disponibilidad</i>	<i>Tiempo de respuesta</i>	<i>Reconoc. de adecuación</i>	<i>Confidencialidad</i>	<i>Capacidad a reemplazo</i>	<i>Modularidad</i>	<i>Portabilidad</i>
<i>Precisión</i>	<i>Tolerancia a fallos</i>	<i>Utilización de recursos</i>	<i>Capacidad de aprendizaje</i>	<i>Integridad</i>	<i>Capacidad a coexistencia</i>	<i>Reusabilidad</i>	<i>Adaptabilidad</i>
<i>Adherencia a normas</i>	<i>Recuperación</i>	<i>Adherencia a normas</i>	<i>Facilidad de uso</i>	<i>No rechazo</i>	<i>Interoperabilidad</i>	<i>Capacidad a análisis</i>	<i>Capacidad a instalación</i>
	<i>Adherencia a normas</i>		<i>Util</i>	<i>Responsabilidad</i>	<i>Adherencia a normas</i>	<i>Capacidad a cambios</i>	<i>Adherencia a normas</i>
			<i>Atractivo</i>	<i>Autenticidad</i>		<i>Estable</i>	
			<i>Accesible técnicamente</i>	<i>Adherencia a normas</i>		<i>mo- dificación</i>	
			<i>Adherencia a normas</i>			<i>Capacidad a testing</i>	
						<i>Adherencia a normas</i>	

Figura 7: Modelo de calidad para la calidad interna y externa para la versión ISO/IEC 25010 Julio 2008.

SQuaRE también define un modelo de calidad para la calidad de uso del software. Cada versión tiene sus características y subcaracterísticas.

<i>Calidad de uso</i>			
<i>Usabilidad de uso</i>	<i>Contexto de uso</i>	<i>Riesgo de uso</i>	<i>Adaptabilidad de uso</i>
<i>Efectividad de uso</i>	<i>Coincidencia de usuario</i>	<i>Riesgos públicos de uso</i>	<i>Aprendizaje de uso</i>
<i>Productividad de uso</i>	<i>Coincidencia de tarea</i>	<i>Riesgos comerciales de uso</i>	<i>Universalidad de uso</i>
<i>Satisfacción de uso</i>	<i>Coincidencia de entorno</i>	<i>Seguridad de uso</i>	<i>Accesibilidad de uso</i>
<i>Adherencia a normas</i>			

Figura 8: Modelo de calidad para la calidad de uso para la versión ISO/IEC 25010 Mayo 2007.

<i>Calidad de uso</i>				
<i>Usabilidad de uso</i>	<i>Contexto de uso</i>	<i>Seguridad de uso(riesgo humano)</i>	<i>Seguridad de uso</i>	<i>Adaptabilidad de uso</i>
<i>Efectividad de uso</i>	<i>Tipos de usuarios de uso</i>	<i>Riesgo a el operador de uso</i>	<i>Adherencia a normas</i>	<i>Aprendizaje de uso</i>
<i>Productividad</i>	<i>Tareas</i>	<i>Riesgos para el público</i>		<i>Flexibilidad</i>

<i>de uso</i>	<i>de uso</i>	<i>en uso</i>		<i>de uso</i>
<i>Satisfacción</i>	<i>Entornos</i>	<i>Riesgos comerciales de</i>		<i>Accesibilidad</i>
<i>de uso</i>	<i>de uso</i>	<i>uso</i>		<i>de uso</i>
<i>Adherencia a</i>	<i>Adherencia a</i>	<i>Riesgo de corrupción</i>		<i>Adherencia</i>
<i>normas</i>	<i>normas</i>	<i>de software de uso</i>		<i>a normas</i>
		<i>Adherencia</i>		
		<i>a normas</i>		

Figura 9: Modelo de calidad para la calidad de uso para la versión ISO/IEC 25010 Julio 2007.

<i>Calidad de uso</i>		
<i>Usabilidad de</i>	<i>Flexibilidad</i>	<i>Seguridad</i>
<i>uso</i>	<i>de uso</i>	<i>de uso(riesgo humano)</i>
<i>Efectividad</i>	<i>Conformidad de</i>	<i>Seguridad y salud del</i>
<i>de uso</i>	<i>contexto de uso</i>	<i>operador</i>
<i>Eficiencia</i>	<i>Extensión de</i>	<i>Salud y seguridad</i>
<i>de uso</i>	<i>contexto de uso</i>	<i>Pública</i>
<i>Satisfacción de</i>	<i>Accesibilidad de uso</i>	<i>Daño al entorno</i>
<i>uso</i>		<i>de uso</i>
<i>Adherencia a</i>	<i>Adherencia a</i>	<i>Daños comerciales</i>
<i>normas</i>	<i>normas</i>	<i>de uso</i>
		<i>Adherencia</i>
		<i>a normas</i>

Figura 10: Modelo de calidad para la calidad de uso para la versión ISO/IEC 25010 Julio 2008.

3. CRITICA /VALORACION

En este apartado mostraremos la evolución y las diferencias que han tenido las dos normas (ISO 9126 y SQuARE) que hemos estado comentando.

9126-1 contiene un modelo de calidad interno y externo, y un modelo de calidad de uso, pero sin embargo, no describe como declarar los requisitos en lo que concierne a subcaracterísticas; o como para un producto dado, cualquiera de estas subcaracterísticas podrán ser medidas. Una solución para llenar el hueco vacío entre el concepto de subcaracterística y una característica medible, se definen los informes técnicos: 9126-2 (sobre métricas externas), 9126-3 (sobre métricas internas) y 9126-4 (sobre métricas de calidad de uso).

ISO 9126-2 no asigna rangos de valores a las métricas, porque estos valores son definidos para cada producto software o una parte del producto software, dependiendo de muchos factores como: la categoría del software, nivel de integridad y necesidades de los usuarios. ISO 9126-2 y 9126-3 proporcionan un conjunto de métricas de calidad (externas e internas) para ser usadas con el modelo ISO 9126-1. Igualmente, ISO 9126-3 no asigna rangos de valores a las métricas, porque estos valores son definidos por cada producto software.

Pero esta norma (ISO 9126) en conjunto con la norma ISO 14598 tiene una serie de problemas que comentamos a continuación:

- ✓ Problemas causados por cambios en el entorno y avances en las tecnologías de la información.
- ✓ **Necesidades de una nueva arquitectura única y la guía “umbrella”.**
 Los dos estándares internacionales (ISO 9126 y ISO 14598) están estrechamente ligados, por cuanto es difícil recordar todos los números de las normas. Se consideró (organización WG6) que estos hechos podrían ser un factor que haga que sea más difícil la popularidad de dichas normas.
 Una buena arquitectura es siempre importante, no solo para un sistema software sino que también para una serie de estándares. Porque una serie de estándares es también como una especie de sistema software. También existe la necesidad de una buena guía; es importante que las series de estándares tengan un nombre de identificación de la serie, las cuales son fáciles para identificarse y para recordarlas.

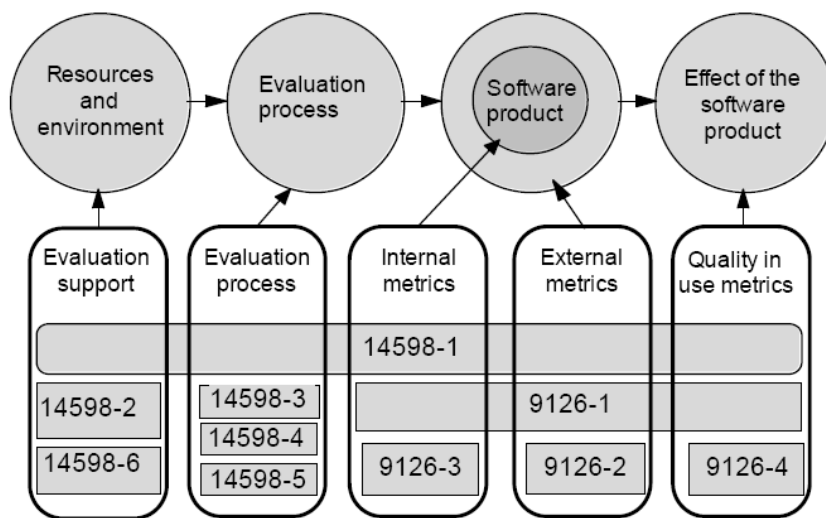


Figura 11: Arquitectura de las series de ISO 9126 y 14598.

- ✓ Se ha realizado las siguientes cuestiones sobre las métricas:
 - ¿Hay esas métricas que son comúnmente usadas para componer distintas clases de métricas?
 - Alguna métrica, la cual es necesaria para la evaluación de calidad de un producto, ¿es usado para medir no un atributo del producto software?
 - ¿Hay métricas que no tienen ninguna correlación directa verificada con características de calidad pero generalmente aprobado como relacionado con la calidad del producto? Por ejemplo: la complejidad ciclométrica, la cohesión,... La organización WG6 llamó a estas métricas, métricas internas puras.
 - Alguna información importante sobre las métricas es distribuida en varias partes de las series, haciendo difícil encontrar y entender toda la información relacionada necesaria para un entendimiento a fondo. Por ejemplo: exigencias generales para métricas son listadas en la ISO 9126-1 como referencia de normativa, mientras que la sección común que explica las métricas en general están en la parte dos, tres y cuatro de la ISO 9126.
- ✓ Como hemos comentado anteriormente, la serie ISO/IEC 9126 fue separada en las series 9126 y 14598 porque el modelo de calidad y las métricas son útiles no solo para la evaluación del producto, sino también para otro objetivo de incluir la especificación de requisitos de calidad. La evaluación de la calidad es posible y significativa cuando

los requisitos de calidad son claramente especificados. Sin embargo, si el estándar de requisitos de calidad es propuesto no como una parte de las series pero como un estándar independiente, entonces esto provocará confusión a los usuarios.

Por lo tanto, SQuaRE nace para solucionar los problemas expuestos anteriormente por las normas ISO 9126 y ISO 14598.

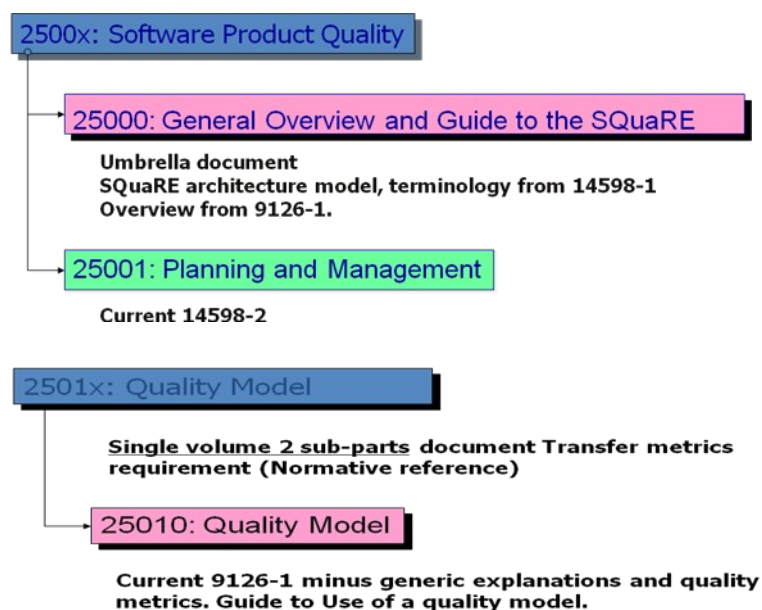
Los mayores beneficios de la serie SQuaRE sobre sus predecesores estándares incluyen:

- La coordinación de dirección sobre la medida y evaluación de calidad del producto software.
- Dirección para la especificación de requisitos de calidad del producto software.
- Armonización con ISO/IEC 15939 en forma de modelo de referencia de modelo de calidad presentado en el estándar SQuaRE.

Esta norma es para usarse en conjunción con las otras partes de los estándares de la serie SQuaRE, y con ISO/IEC 14598 hasta ser reemplazado por las series ISO/IEC 25000. Las diferencias principales entre los estándares de la serie SQuaRE y sus predecesores ISO/IEC 9126 y ISO/IEC 14598 son los siguientes:

- Introducción a un nuevo modelo de referencia general.
- Introducción de guías dedicadas y detalladas para cada división.
- Introducción de elementos de medida de calidad dentro de la división de medida de calidad.
- Introducción de la división de requisitos de calidad.
- Incorporación y revisión de los procesos de evaluación.
- Introducción de guías para uso práctico en forma de ejemplos.
- Y coordinación y armonización del contenido con la ISO/IEC 15939.

A modo esquemático, mostramos a continuación la representación de las distintas divisiones de SQuaRE y las diferencias producidas con respecto a la ISO 9126 y la ISO/IEC 12598:



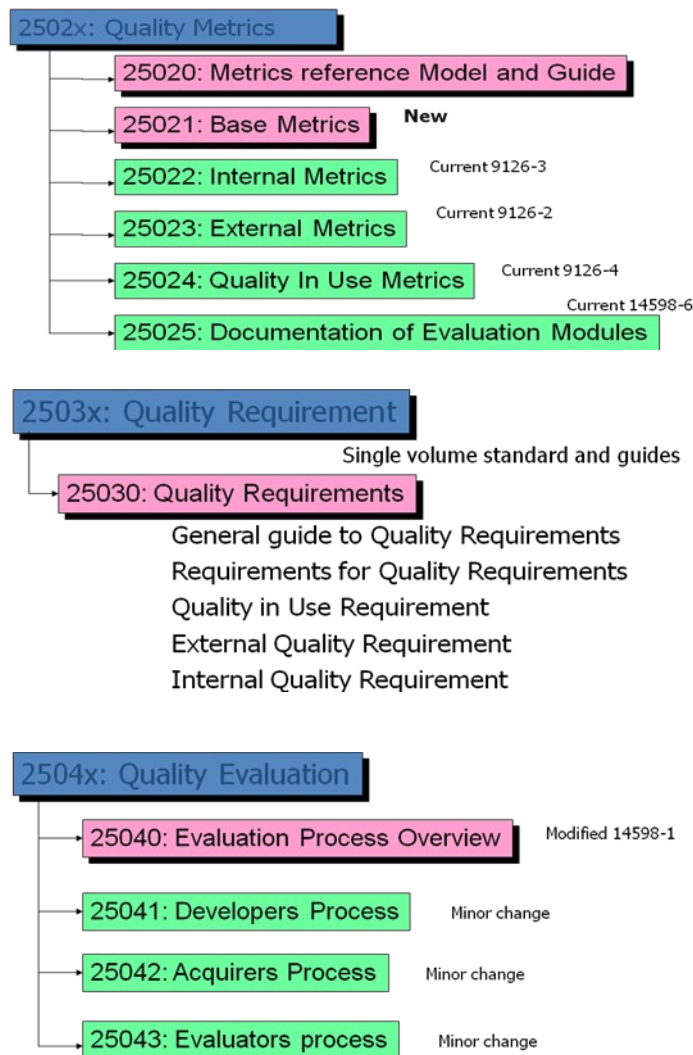


Figura 12: Divisiones de la familia SQaRE.

Viendo la figura, podemos observar que aspectos nuevos de las distintas divisiones son:

- Descripción general y guía a SQaRE. Implica requisitos generales para la calidad del producto software y proporcionar una guía para el uso de los documentos de las series SQaRE (25000).
- Modelo de referencia de métricas y guía (25020). Esta parte da información general sobre las métricas de calidad como una un documento “umbrella” de la división de métricas.
- Métricas base (25021). Esta nueva parte definirá y especificará un conjunto recomendado de métricas base que puede ser usado durante el ciclo de vida del desarrollo del software.
- Requisitos de calidad (25030). Esta parte permite a la calidad de un producto software ser especificado como un requisito de calidad.

En la división que nos trata (25010) vemos que es una revisión de la ISO 9126-1. Donde se describe un modelo de calidad para el producto software, el cual consta de dos partes, el modelo de calidad interna y externa, y el modelo de calidad de uso, guiando a los usuarios a aplicar un modelo a un producto específico.

A continuación mostraremos las diferencias entre la última versión de SQuaRE (Julio 2008) y la ISO 9126-1. En primer lugar, a la calidad interna y externa de ISO 9126-1 en SQuaRE es llamado calidad del producto software, dónde la calidad en uso es ahora una calidad de sistema. Seguidamente mostramos las diferencias entre las características y subcaracterísticas del modelo de calidad interno y externo de la ISO 9126-1 y el modelo de calidad del producto software de SQuaRE:

SQuaRE	ISO/IEC 9126-1	Características
Adecuación funcional	Funcionalidad	El nuevo nombre es más preciso, y no provoca confusiones con otros significados de funcionalidad.
	Interoperabilidad	Movido a Compatibilidad.
	Seguridad	Característica propia de ISO 9126-1.
Disponibilidad	Madurez	Disponibilidad es mucho más importante que madurez.
Robustez		Subcaracterística de SQuaRE.
Eficiencia de rendimiento	Eficiencia	Renombrado para no provocar conflictos con otras definciones.
Operabilidad	Usabilidad	Renombrado para no provocar conflictos con otras definiciones.
Reconocimiento de adecuación	Comprensibilidad	El nuevo nombre de SQuaRE es mucho más preciso.
Facilidad de uso	Operabilidad	Simplemente se ha renombrado.
Util		Nueva subcaracterística de SQuaRE.
Accesibilidad Técnica		Nueva subcaracterística de SQuaRE.
Seguridad	Seguridad	En SQuaRE es una característica, en la ISO 9126-1 es una subcaracterística.
Compatibilidad		No estaba suficientemente declarado en las subcaracterísticas de Portabilidad en la ISO 9126-1.
Interoperabilidad		En la ISO 9126-1 es una subcaracterística de Funcionalidad. Ahora lo es de Compatibilidad.
Modularidad		Nueva subcaracterística de SQuaRE.
Reusabilidad		Nueva subcaracterística de SQuaRE.
Estabilidad a modificaciones	Estabilidad	El nuevo nombre en SQuaRE es mucho más preciso.
Transmisibilidad	Portabilidad	ISO 9126-1 no incluía una subcaracterística para portabilidad.
Portabilidad		Nueva subcaracterística de SQuaRE.
	Coexistencia	Movido desde la portabilidad (en ISO 9126-1) hasta compatibilidad.
	Capacidad de reemplazo	Movido desde la portabilidad (en ISO 9126-1) hasta compatibilidad.

Para el modelo de calidad de uso, existen las siguientes diferencias entre las normas 9126-1 y SQuaRE:

SQuaRE	ISO/IEC 9126-1	Características
Usabilidad en uso		Nueva subcaracterística de SQuaRE.
Flexibilidad en uso		Nueva característica de SQuaRE.
Conformidad del contexto de uso		Nueva subcaracterística de SQuaRE.
Extensión del contexto de uso		Nueva subcaracterística de SQuaRE
Salud y seguridad de Operador		Nueva subcaracterística de SQuaRE
Salud y seguridad público		Nueva subcaracterística de SQuaRE
Daño del entorno de uso		Nueva subcaracterística de SQuaRE
Daños comerciales de uso		Nueva subcaracterística de SQuaRE

Entre las distintas revisiones de SQuaRE solamente se realiza cambios de características y subcaracterísticas en el modelo de calidad interna y externa. Igualmente, se producen cambios de características y subcaracterísticas en el modelo de calidad de uso (ver figuras de la 5 a la 10).

4. Conclusiones

La serie de primera generación ISO 9126 está dividida en tres partes: 9126-1 que contiene un modelo de calidad interna y externa, y un modelo de calidad de uso, 9126-1 que está formado por conjunto de métricas externas y finalmente 9126-3 que está formado por un conjunto de métricas internas. El modelo de calidad está formado por una serie de características y estas a su vez por subcaracterísticas. Las subcaracterísticas pueden ser medidas por internas o externas métricas. Es prácticamente imposible medir todas las internas y externas subcaracterísticas para todo el producto software. No es práctico medir la calidad de uso para todos los posibles escenarios usuario-tarea. La ISO 9126-2 contiene

la terminología relacionada con las medidas de métrica, el uso de las métricas en el proceso del ciclo de vida y conjuntos introductorios básicos de métricas externas y de calidad en uso para cada característica y subcaracterística de calidad de software. ISO 9126-3 proporciona métricas internas para medir los atributos de las características de calidad definidas en 9126-1.

SQuaRE es una revisión de 9126-1 y tiene las mismas características de calidad del software. En el campo de la calidad de software hay dos aspectos importantes, el producto y el proceso. SQuaRE se centra en el producto. Tiene tres revisiones, y cada una de ellas tiene un modelo de calidad del producto software con distintas características y subcaracterísticas. Por lo tanto en la serie SQuaRE ha habido una evolución en dicho modelo, intentando definir lo más preciso posible cada una de las características y subcaracterísticas. Es decir, desde la primera revisión de Mayo de 2007 a la última revisión de Julio de 2008 ha habido una remodelación de las características y subcaracterísticas siendo la última versión la que tiene más precisión en dicho aspecto.

También hemos mostrado las diferencias entre la ISO 9126.1 y la última versión de SQuaRE, demostrando que existe más precisión en esta última norma. Además la versión de SQuaRE es también más completa pues define subcaracterísticas (además de para la calidad interna y externa) para la calidad de uso.

5. **Tendencias futuras**

Tal y como hemos estado comentando las diferencias entre las normas ISO 9126 y SQuaRE, vemos claramente que SQuaRE es la siguiente generación a ISO 9126. Por lo tanto, se seguirá utilizando la ISO 9126 hasta que pueda ser reemplazado totalmente por la familia SQuaRE. Actualmente se está trabajando en nuevas revisiones de la familia SQuaRE, aunque lógicamente todavía tiene que pasar mucho tiempo y recorrer mucho camino la familia SQuaRE para que se tome como estándar en todos los países.

