



XI CURSOS DE VERANO DE SANTANDER

CALIDAD DE PROCESOS Y PRODUCTOS SOFTWARE

CALIDAD DEL PRODUCTO SOFTWARE ISO/IEC 25000



D. Moisés Rodríguez Monje

16 Julio 2010

Presentación



Moisés Rodríguez Monje

Ingeniero en Informática

Empresa: Alarcos Quality Center

Cargo: Director Técnico

Correo moises.rodriguez@alarcosqualitycenter.com

Web: www.alarcosqualitycenter.com

La principal actividad de AQC se centra en la **consultoría y prestación de servicios en el área de la calidad de los sistemas de la información**. Específicamente en temas relacionados con:

- Creación y mejora de fábricas/departamentos de software
- Calidad de procesos y productos software
- Testing* de sistemas informáticos
- Calidad de datos
- Ingeniería de procesos

- Introducción**
- Calidad del Producto Software**
- Principales normas y estándares de la Calidad del Producto Software**
- Familia ISO/IEC 25000 (SQuaRE)**
- Herramientas para Evaluación de la Calidad del Producto**
- Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software**



□ Introducción

- Calidad del Producto Software
- Principales normas y estándares de la Calidad del Producto Software
- Familia ISO 25000 (SQuaRE)
- Herramientas para Evaluación de la Calidad del Producto
- Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software

Introducción

Antecedentes del Sector



Datos del Crecimiento Experimentado

- **2007:** El sector TI creció un 9%, del cual el software supuso un 17% (AETIC y MITyC, 2007)
- **2008:** Las factorías de software españolas dieron trabajo a 10.744 personas y facturaron 376 millones de euros (AETIC, 2009).
- **2008:** Importancia de la externalización de software, alrededor de 1,3 B\$ (Gartner Group).
- **2009:** Importancia creciente de las certificaciones software.



Country	Number of Appraisals	Maturity Level 1 Reported	Maturity Level 2 Reported	Maturity Level 3 Reported	Maturity Level 4 Reported	Maturity Level 5 Reported	Country	Number of Appraisals	Maturity Level 1 Reported	Maturity Level 2 Reported	Maturity Level 3 Reported	Maturity Level 4 Reported	Maturity Level 5 Reported
Argentina	77		50	18	2	4	Malaysia	71		22	43		6
Australia	36	1	8	7	2	4	Mauritius	10 or fewer					
Austria	10 or fewer						Mexico	86		36	39	3	6
Bahrain	10 or fewer						Morocco	10 or fewer					
Bangladesh	10 or fewer						Nepal	10 or fewer					
Belarus	10 or fewer						Netherlands	14		5	7		1
Belgium	10 or fewer						New Zealand	10 or fewer					
Brazil	144	1	71	57	1	11	Norway	10 or fewer					
Bulgaria	10 or fewer						Pakistan	28	1	21	4		1
Canada	59	1	16	24	5	4	Panama	10 or fewer					
Chile	37		22	12		2	Peru	10 or fewer					
China	1229	1	135	987	36	48	Philippines	23		2	11		8
Colombia	34		12	13	3	2	Poland	10 or fewer					
Costa Rica	10 or fewer						Portugal	14		5	7		1
Czech Republic	10 or fewer						Romania	10 or fewer					
Denmark	10 or fewer						Russia	11			3	3	4
Dominican Republic	10 or fewer						Saudi Arabia	10 or fewer					
Egypt	43	1	22	12	2	3	Singapore	21		4	11	1	4
Finland	10 or fewer						Slovakia	10 or fewer					
France	168	4	98	53	1	2	South Africa	10 or fewer					
Germany	76	9	37	15	1	1	Spain	180	1	108	54	3	4
Greece	10 or fewer						Sri Lanka	14		5	10		
Guatemala	10 or fewer						Sweden	10 or fewer					
Hong Kong	18		2	11		5	Switzerland	10 or fewer					
Hungary	10 or fewer						Taiwan	134	1	76	51	2	2
India	524		17	278	24	189	Taiwan	38		12	24		1
Indonesia	10 or fewer						Thailand	10 or fewer					
Ireland	11		2	4			Tunisia	10 or fewer					
Israel	19		3	11		3	Turkey	16			14		2
Italy	43		19	20			Ukraine	10 or fewer					
Japan	306	18	86	140	13	17	United Arab Emirate	10 or fewer					
Korea, Republic Of	165	1	55	75	15	8	United Kingdom	113	3	50	35	1	4
Latvia	10 or fewer						United States	1582	30	564	610	23	141
Lithuania	10 or fewer						Uruguay	10 or fewer					
							Viet Nam	17			12	2	3

<http://www.sei.cmu.edu/cmmi/casestudies/profiles/pdfs/upload/2010MarCMMI.pdf>



Calidad Software 15504.es

MENÚ PRINCIPAL

- Home
- Objetivo del portal
- Norma ISO 15504 SPICE
- Modelo de procesos ISO 12207:2008
- Organizaciones certificadas
- Niveles de madurez
- CMMI e ISO/IEC 15504
- Herramientas
- Contactar

AYUDAS A LA CERTIFICACIÓN

- Convocatoria 2010 (España)
- Convocatoria 2009 (España)
- Convocatoria 2008 (España)

COMUNIDAD

- Foro
- Formación
- ISO/IEC 15504 en los medios
- Presentaciones de la norma
- Otro material de apoyo
- Noticias
- LinkedIn

Nombre de usuario:

Contraseña:

Home > Organizaciones certificadas > Organizaciones Certificadas en ISO 15504

Organizaciones Certificadas en ISO 15504

A continuación se presenta la lista de empresas certificadas en ISO 15504. Si su empresa está certificada en la norma ISO 15504 y están interesados en aparecer en la lista, pueden enviarnos un correo a editor@iso15504.es.

BÚSQUEDA: 25 (1)

EMPRESA	PAÍS	PROVINCIA	NIVEL DE MADUREZ	MODELO DE PROCESOS	AÑO DE CERTIFICACIÓN	ORGANISMO AUDITOR
Aihambra Eidos	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Ampla Soluciones	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Artyco	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Astvia	ES	CÓRDOBA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Consultia IT	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Ecuia Tecnologías	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Excella	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
GCL Informática	ES	ALBACETE	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Global Solutions Development	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Gotel Data	ES	CUENCA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Indenova	ES	VALENCIA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Insoc Data	ES	ALBACETE	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Isi Diseño de Sistemas de Información	ES	BARCELONA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Mangana Webs	ES	ALBACETE	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Mercanza	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Neoris España	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Pulitair	ES	A CORUÑA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
SADE Consultoría técnica	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Servinet	ES	CUENCA	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Simave Sistemas	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Visure Solutions	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR
Xtream Sistemas de Información Global	ES	MADRID	Nivel 2	ISO/IEC 12207:2008	2010	AENOR

LO MÁS LEÍDO EN EL FORO

- ¿Alguna herramienta para requisitos?
- Organizaciones que certifican SPICE
- Certificación métodos ágiles
- Lead Appraisers vs Evaluadores ISO

ENLACES PATROCINADOS

AENOR

Kybelc consulting

PLAN CAVANZA 2.0
Unase a nuestro grupo de empresas

CURSO GRATUITO ONLINE
DEL 16 AL 23 DE JULIO
PLAZAS LIMITADAS

SI NECESITAS ASESORAMIENTO PARA LA IMPLANTACIÓN O CERTIFICACIÓN EN

<http://www.iso15504.es>



computing.es

AGENDA ENCUNTROS ESPECIALES

REVISTA DIRECTORIO de EMPRESAS NOMBRAMIENTOS ANUARIO EMPRESAS TIC

PORTADA ACTUALIDAD CASOS de ÉXITO ENTREVISTAS INFORMES OPINIÓN SOLUCIONES TENDENCIAS WHITEPAPERS VÍDEOS EXTRA

COMUNICACIONES e-ADMINISTRACIÓN NEGOCIOS INTERNET CAPITAL HUMANO GESTIÓN CIOs PRODUCTOS INFRAESTRUCTURAS SEGURIDAD

Computing > Noticias > Negocios > Gartner revisa a la baja sus previsiones de inversión mundial en TI para 2010

Negocios

Gartner revisa a la baja sus previsiones de inversión mundial en TI para 2010

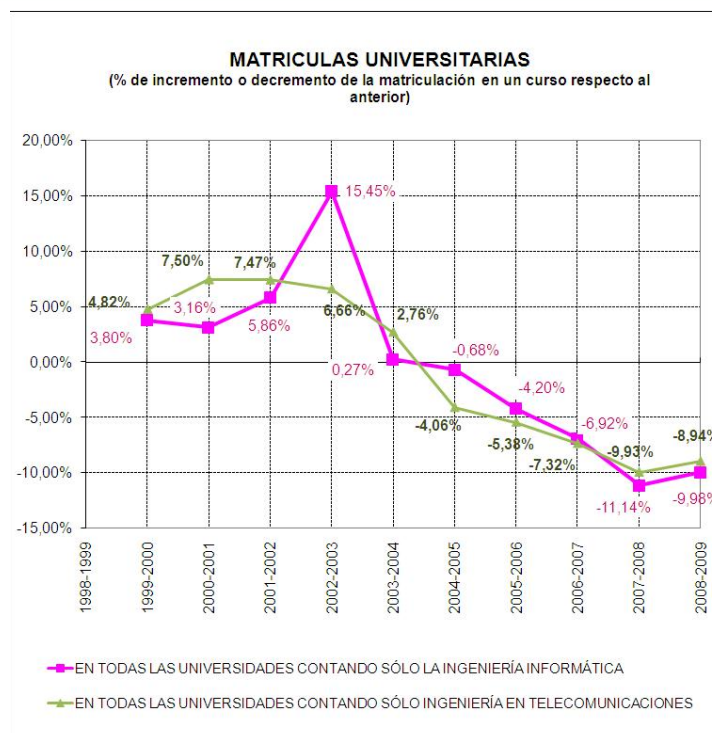
La inversión mundial en TI alcanzará en 2010 la cifra de 3.350 trillones de dólares estadounidenses en 2010. Así lo indican las previsiones de Gartner, que ha revisado a la baja el ratio de incremento esperado respecto a 2009 desde el 5,3 por ciento del primer trimestre a un 3,9 por ciento.

escrito por: Lola Sánchez

Jueves, 01 de julio de 2010

Comentarios Comentar y más Compartir Imprimir Enviar Valorar

<http://www.computing.es/Noticias/201007010021/Gartner-revisa-a-la-baja-sus-previsiones-de-inversion-mundial-en-TI-para-2010.aspx>



<http://www.javiergarzas.com/2010/07/matriculas-ingenieria-informatica-telecomunicaciones.html>



WORLD ECONOMIC FORUM INSEAD The Global Information Technology Report 2009-2010

HOME FRAMEWORK FULL REPORT DATA ANALYSIS PREVIOUS REPORTS

Index Ranking Country Profiles Data Tables Strengths & Weaknesses Compare 2 Countries Historical Trend Map View

Bar Chart Comparison Scatter Chart Comparison Spider Chart Comparison

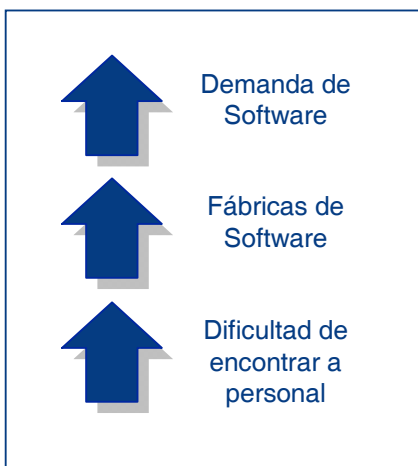
Change index: Choose your view: All countries Custom Selection [Make group] Countries in pre-defined regions [SUB-SAHARAN AFRICA] View

Network Readiness Index

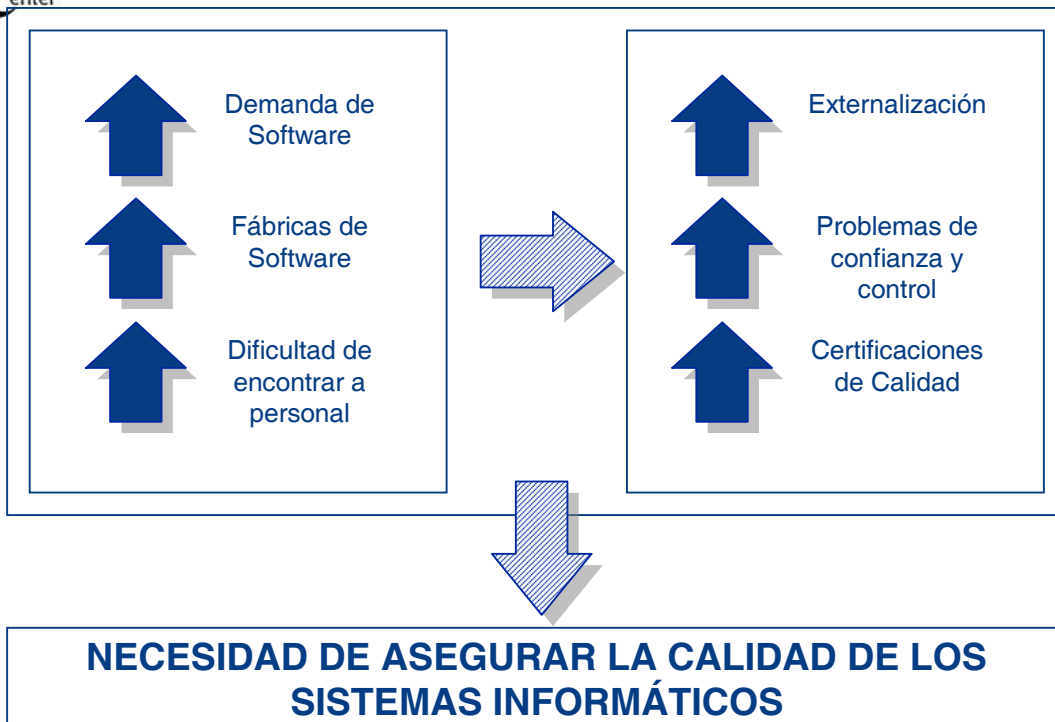
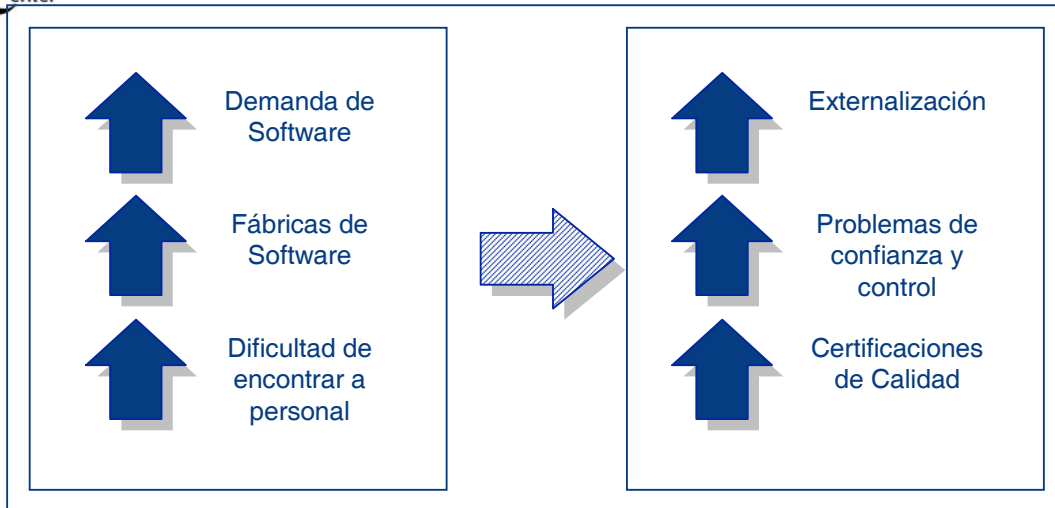
Country	Rank	Country	Rank	Country	Rank
Sweden	1	Barbados	35	Turkey	69
Singapore	2	Czech Republic	36	Egypt	70
Denmark	3	China	37	Bulgaria	71
Switzerland	4	Saudi Arabia	38	Sri Lanka	72
United States	5	Tunisia	39	Macedonia, FYR	73
Finland	6	Chile	40	Dominican Republic	74
Canada	7	Lithuania	41	Senegal	75
Hong Kong SAR	8	Montenegro	42	Kuwait	76
Netherlands	9	India	43	Gambia, The	77
Norway	10	Jordan	44	Mexico	78
Taiwan, China	11	Puerto Rico	45	Trinidad and Tobago	79
Iceland	12	Hungary	46	Russian Federation	80
United Kingdom	13	Thailand	47	El Salvador	81
Germany	14	Italy	48	Ukraine	82
Korea, Rep.	15	Costa Rica	49	Guatemala	83
Australia	16	Oman	50	Serbia	84
Luxembourg	17	Croatia	51	Philippines	85
France	18	Latvia	52	Botswana	86
New Zealand	19	Mauritius	53	Pakistan	87
Austria	20	Vietnam	54	Morocco	88
Japan	21	Slovak Republic	55	Namibia	89
Belgium	22	Greece	56	Kenya	90
United Arab Emirates	23	Uruguay	57	Argentina	91
Ireland	24	Panama	58	Peru	92
Estonia	25	Romania	59	Georgia	93
Malta	26	Colombia	60	Mongolia	94
Malaysia	27	Brazil	61	Albania	95
Israel	28	South Africa	62	Mali	96
Bahrain	29	Brunei Darussalam	63	Zambia	97
Qatar	30	Azerbaijan	64	Ghana	98
Slovenia	31	Poland	65	Nigeria	99
Cyprus	32	Jamaica	66	Guyana	100
Portugal	33	Indonesia	67	Armenia	101
Spain	34	Kazakhstan	68	Mauritania	102

<http://www.networkedreadiness.com/gitr/main/analysis/showindexranking.cfm?vno=a>

Calidad de Producto Software - ISO/IEC 25000

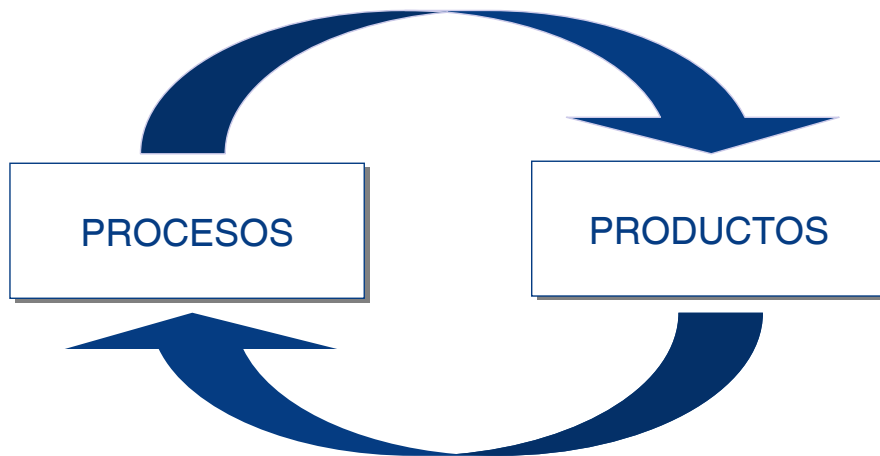


Calidad de Producto Software - ISO/IEC 25000





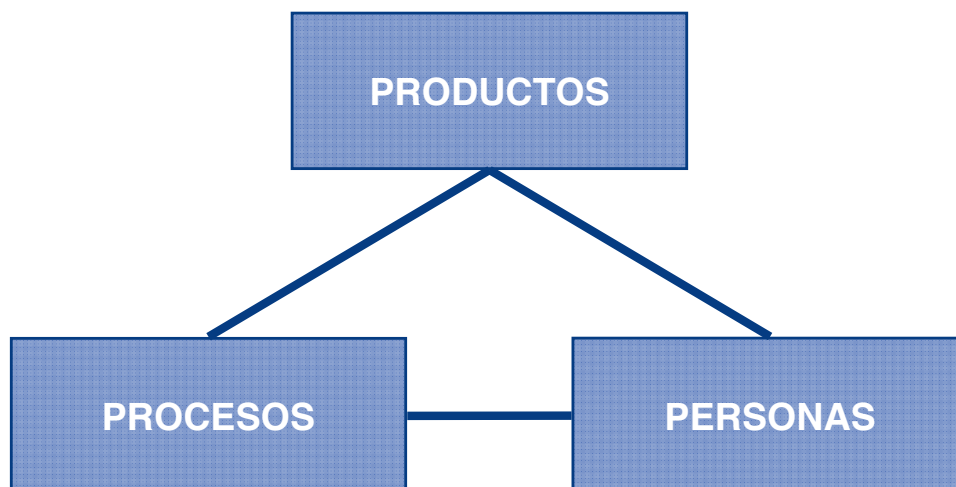
Mejorar el proceso, mejora el producto



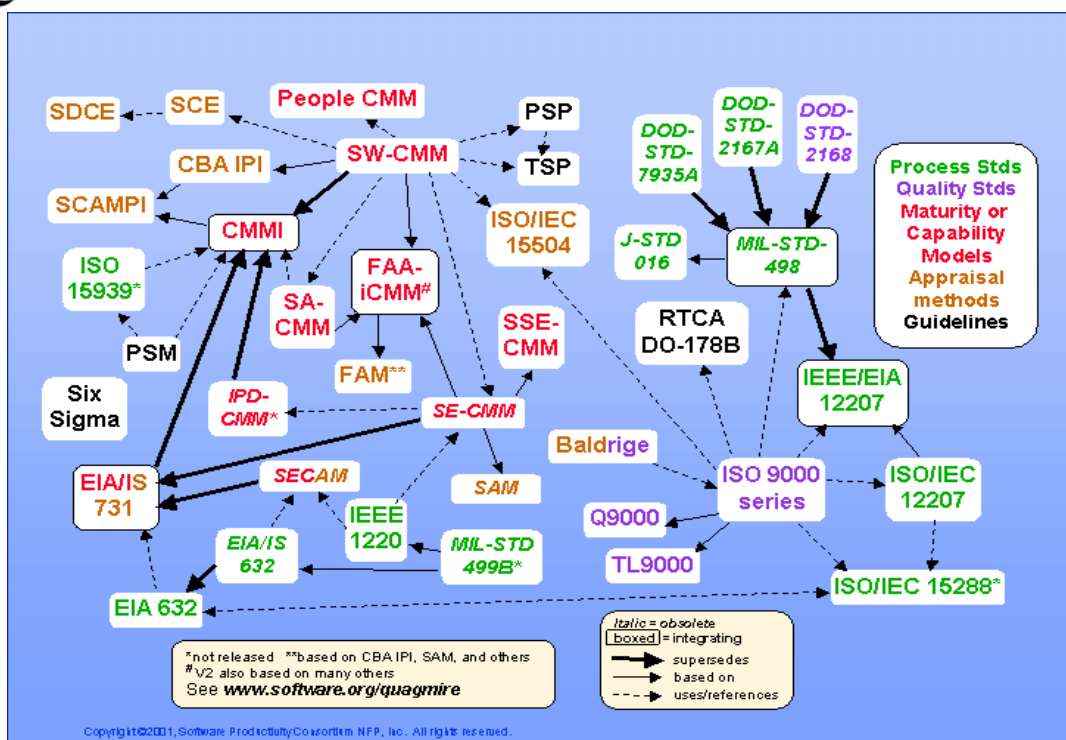
Mejorar el producto, puede facilitar los procesos

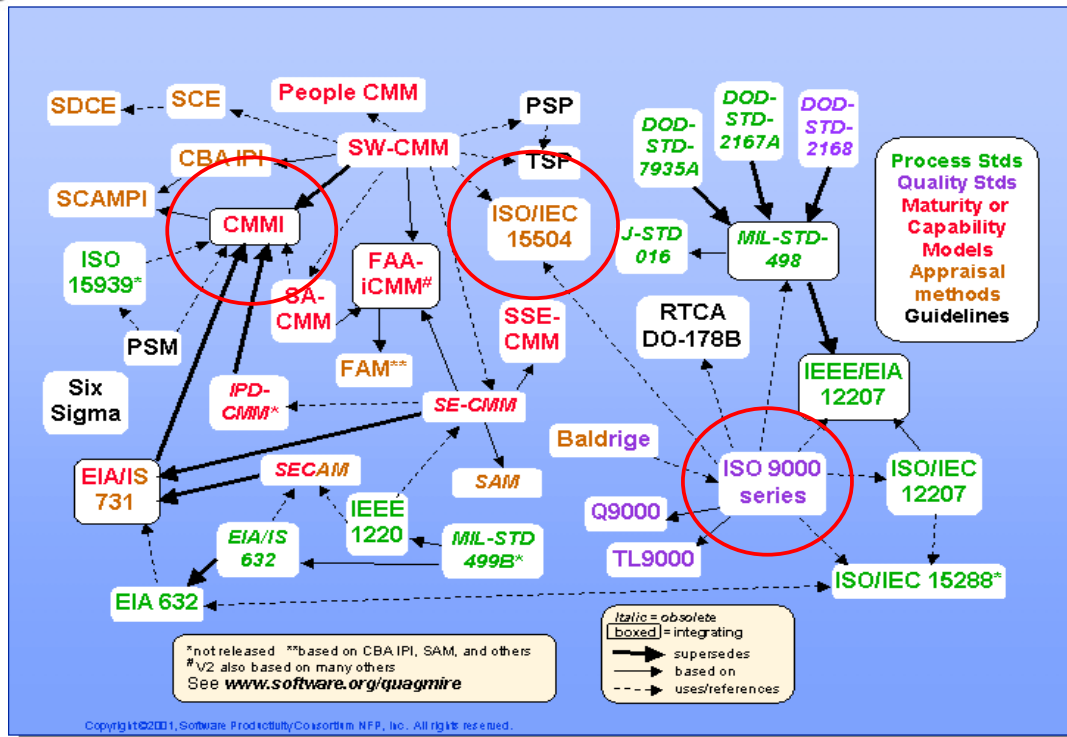


Orientaciones de la Calidad



“Actualmente las principales iniciativas software se centran en **PROCESOS**”





Calidad de Producto Software - ISO/IEC 25000



“Hay poca evidencia en que cumplir un modelo de procesos asegure la calidad del producto, la estandarización de los procesos garantiza la uniformidad en la salida de los mismos, lo que puede incluso institucionalizar la creación de malos productos”

Kitchenham, B. y Pfleeger, S. L. (1996). "Software Quality: The Elusive Target." IEEE Software 20(1): 12-21.

Calidad de Producto Software - ISO/IEC 25000



“Las evaluaciones deberían basarse
en **evidencias directas del
producto**, y no en evidencias
circunstanciales del proceso”

Maibaum, T. y Wassyng, A. 2008. A Product-Focused Approach to Software
Certification. Computer Volume: 41, Issue: 2: 91-93



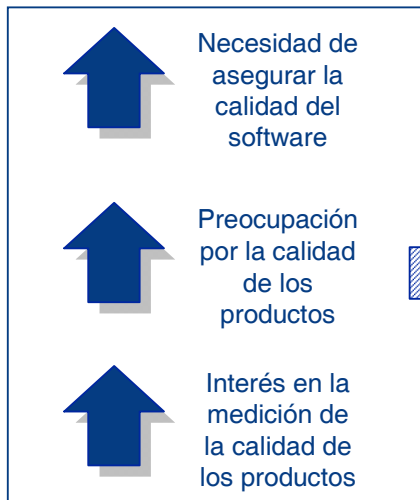
Necesidad de
asegurar la
calidad del
software



Preocupación
por la calidad
de los
productos



Interés en la
medición de
la calidad de
los productos



“Actualmente existe la **necesidad de un entorno (metodológico y tecnológico)** que permita **automatizar la adquisición del producto, la medición de la calidad, el análisis y la presentación de los resultados**”



- Introducción
- Calidad del Producto Software**
- Principales normas y estándares de la Calidad del Producto Software
- Familia ISO 25000 (SQuaRE)
- Herramientas para Evaluación de la Calidad del Producto
- Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software

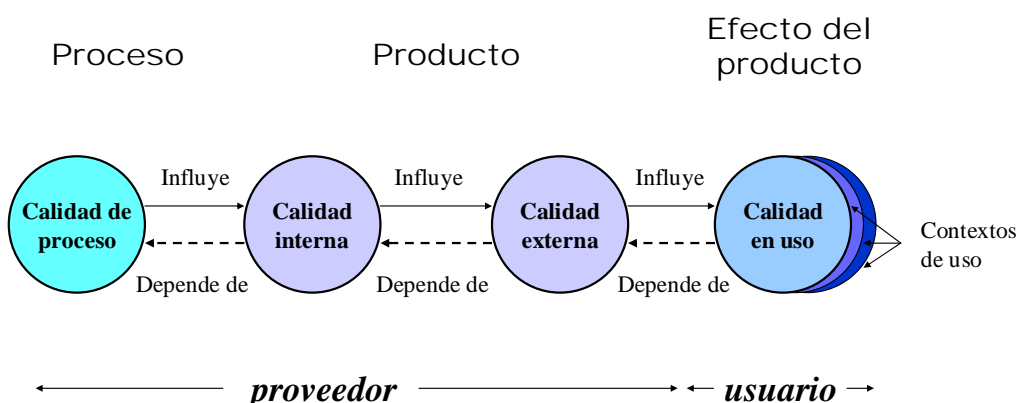
Que se entiende por Calidad...

- “Propiedad o conjunto de propiedades inherentes a algo, que permiten juzgar su valor”. *Real Academia Española.*
- “Grado en el que un conjunto de características inherentes cumple con los requisitos”. *ISO 9000.*
- “Calidad es cumplimiento de requisitos”. *Philip B. Crosby.*
- “Calidad es satisfacción del cliente”. *William E. Deming*
- “Grado en que el producto software satisface las necesidades expresadas o implícitas, cuando es usado bajo condiciones determinadas”. *ISO 25000.*

Cuando no hay Calidad...

- Programas que **no hacen exactamente lo que se espera**
- Proyectos que **no terminan nunca**
- Sistemas informáticos que **no se utilizan por la dificultad de su manejo**
- Productos software que son **imposibles de mantener** cuando desaparece la persona o personas que lo desarrollaron
- Software **poco seguro**

- ❑ **Interna:** medible a partir de las características intrínsecas del propio producto software (como el código fuente).
- ❑ **Externa:** medible a partir del comportamiento del producto software (como durante una prueba)
- ❑ **En uso:** medible durante la utilización efectiva por parte del usuario (en un entorno de pre o producción).

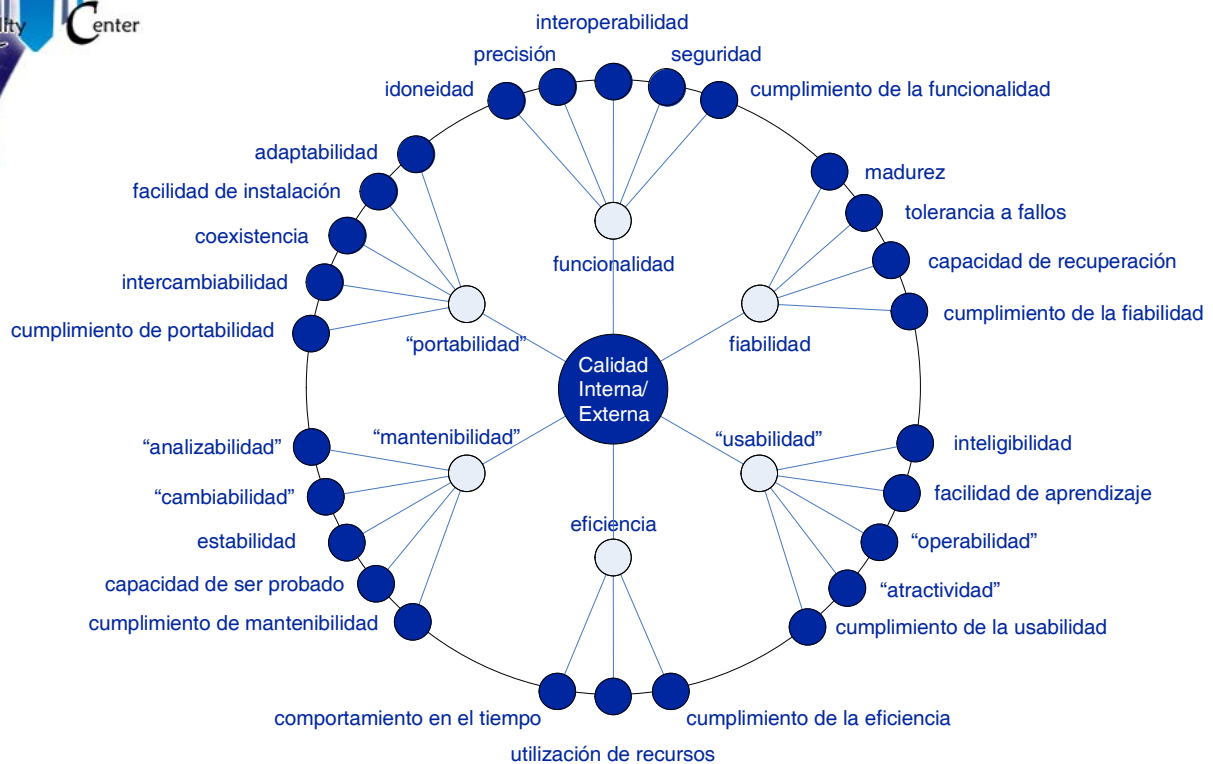
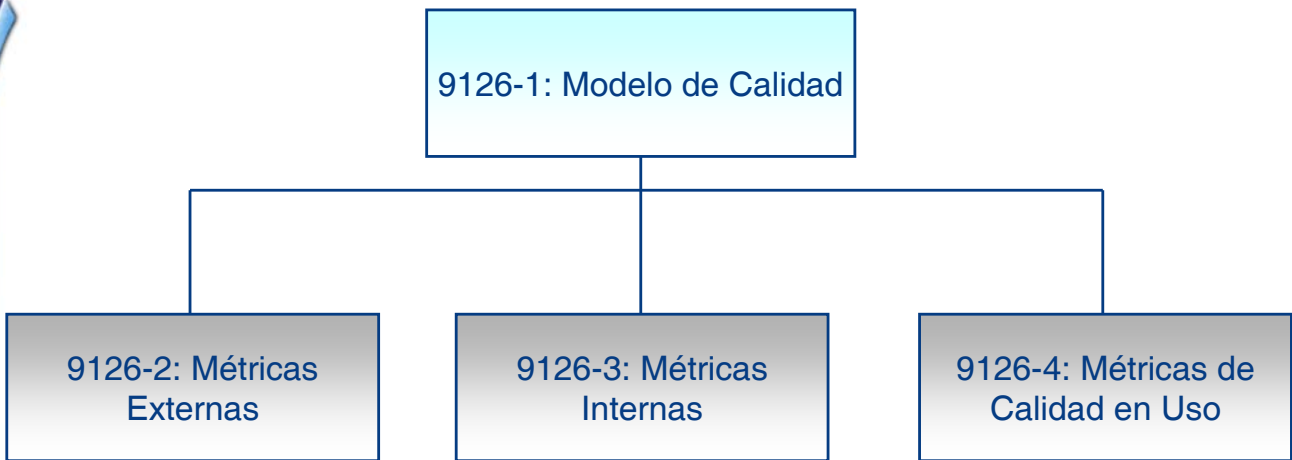


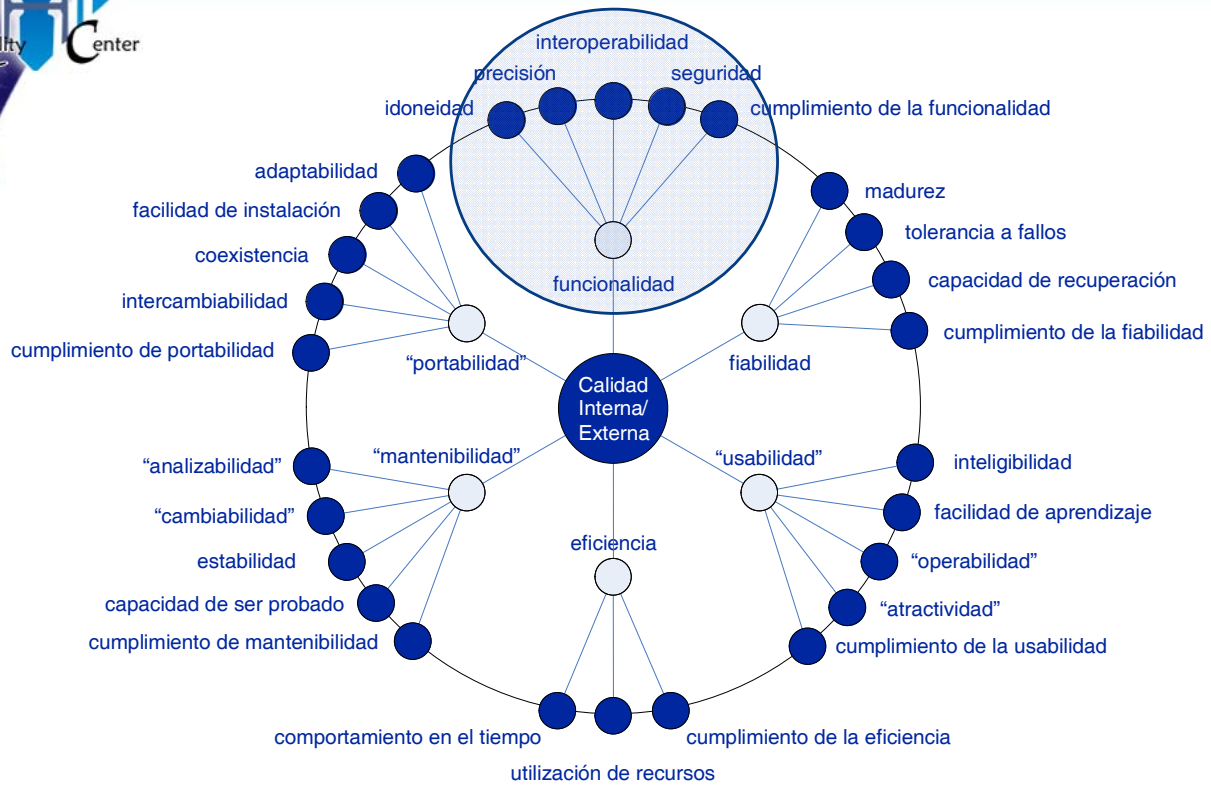
- Es el grado en el que producto software **incorpora un conjunto de características**, de manera que **garantiza el cumplimiento de los requisitos y necesidades del cliente**.
- El objetivo **no es necesariamente alcanzar una calidad perfecta**, sino **la necesaria y suficiente para cada contexto de uso** a la hora de la entrega y del uso **por parte de los usuarios**.
- Es primordial **comprender la existencia de distintas visiones de la calidad (productor, usuario, valor, etc.)**.
- Dada la complejidad de la calidad, es necesario utilizar un **modelo que especifique las características de calidad**

- Introducción
- Calidad del Producto Software
- Principales normas y estándares de la
Calidad del Producto Software**
- Familia ISO 25000 (SQuaRE)
- Herramientas para Evaluación de la Calidad del
Producto
- Ejemplo de Entorno para la Evaluación de la
Calidad del Producto Software

- ISO/IEC 9126:2001**
- ISO/IEC 14598**
- Familia de normas ISO 25000**

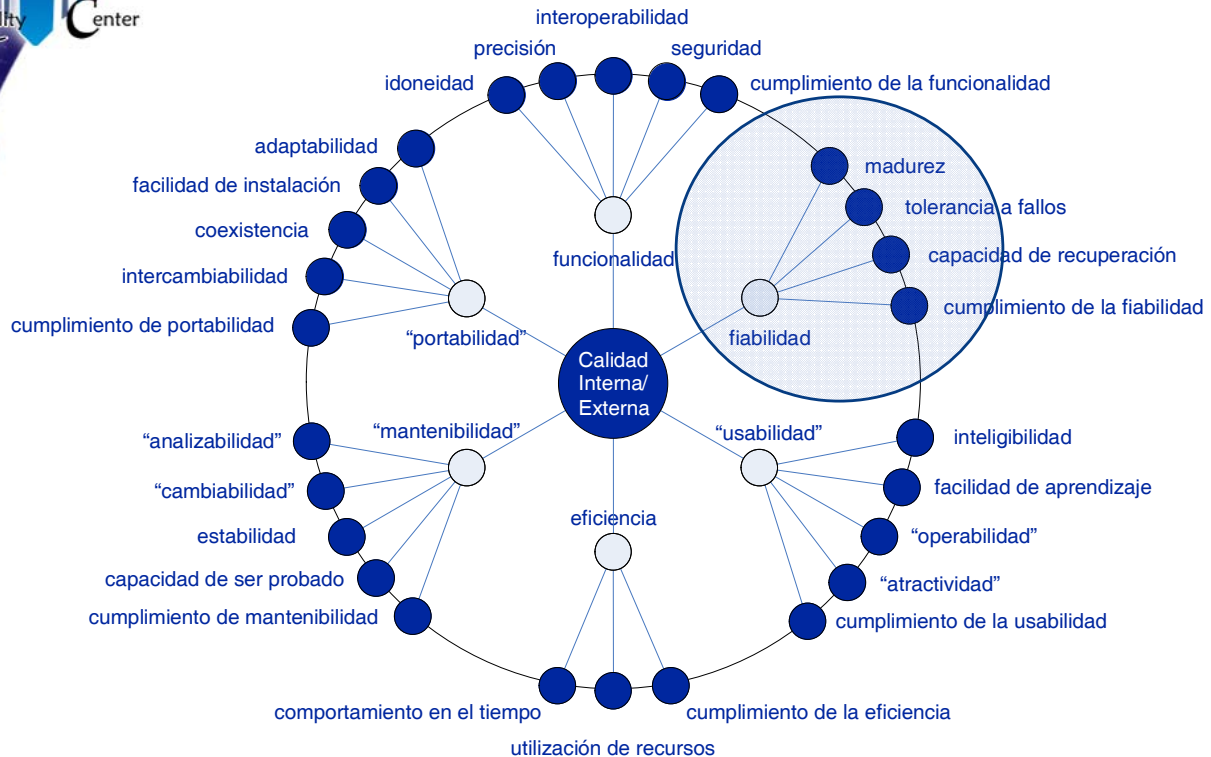
- Primera versión de 1991**
- Revisada y actualizada en 2001**
- Actualmente se encuentra en estado 90.93**
- Formada por 4 partes**
- Establece un modelo de calidad para el producto software.**





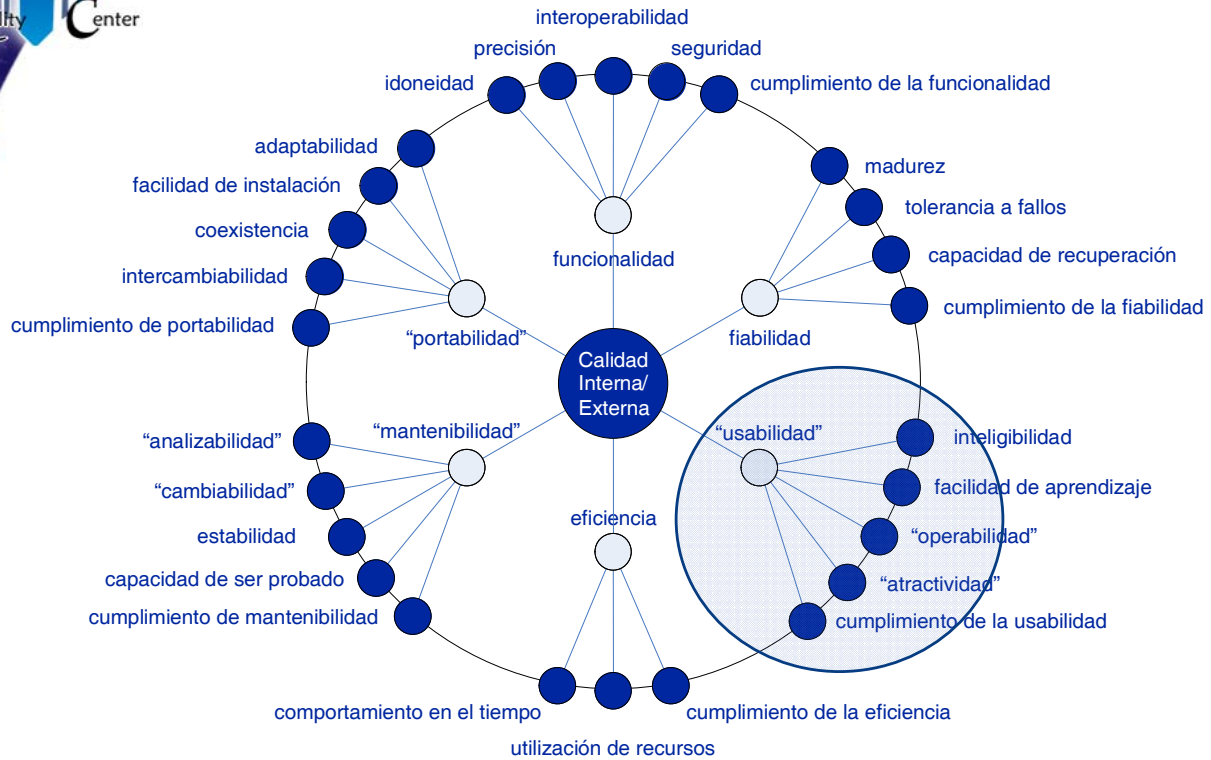
Funcionalidad

- Adecuación:** Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.
- Exactitud:** Capacidad del producto software para proporcionar los resultados o efectos correctos o acordados, con el grado necesario de precisión.
- Interoperabilidad:** Capacidad del producto software para interactuar con uno o más sistemas especificados.
- Seguridad de acceso:** Capacidad del producto software para proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados
- Cumplimiento funcional:** Capacidad del producto software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con funcionalidad.



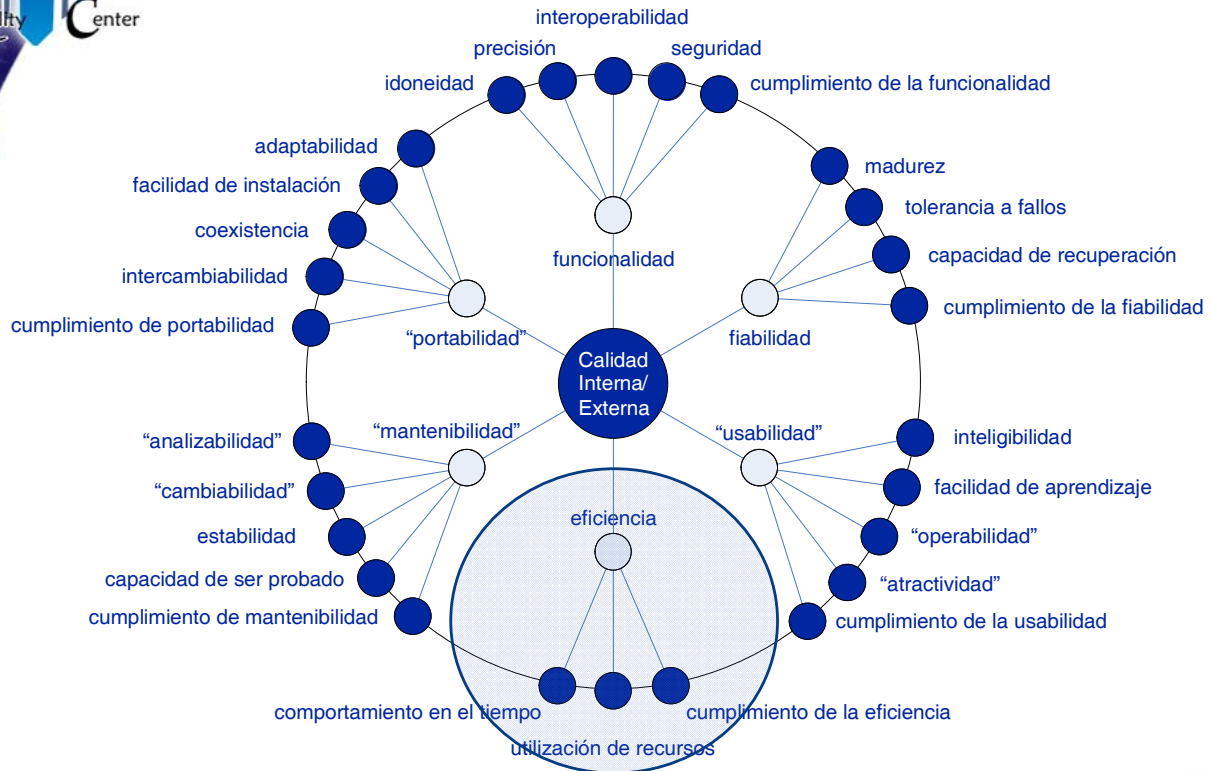
Fiabilidad

- Madurez:** Capacidad del producto software para evitar fallar como resultado de fallos en el software.
- Tolerancia a fallos:** Capacidad del software para mantener un nivel especificado de prestaciones en caso de fallos software o de infringir sus interfaces especificados.
- Capacidad de recuperación:** Capacidad del producto software para reestablecer un nivel de prestaciones especificado y de recuperar los datos directamente afectados en caso de fallo.
- Cumplimiento de la fiabilidad:** Capacidad del producto software para adherirse a normas, convenciones o regulaciones relacionadas con al fiabilidad.



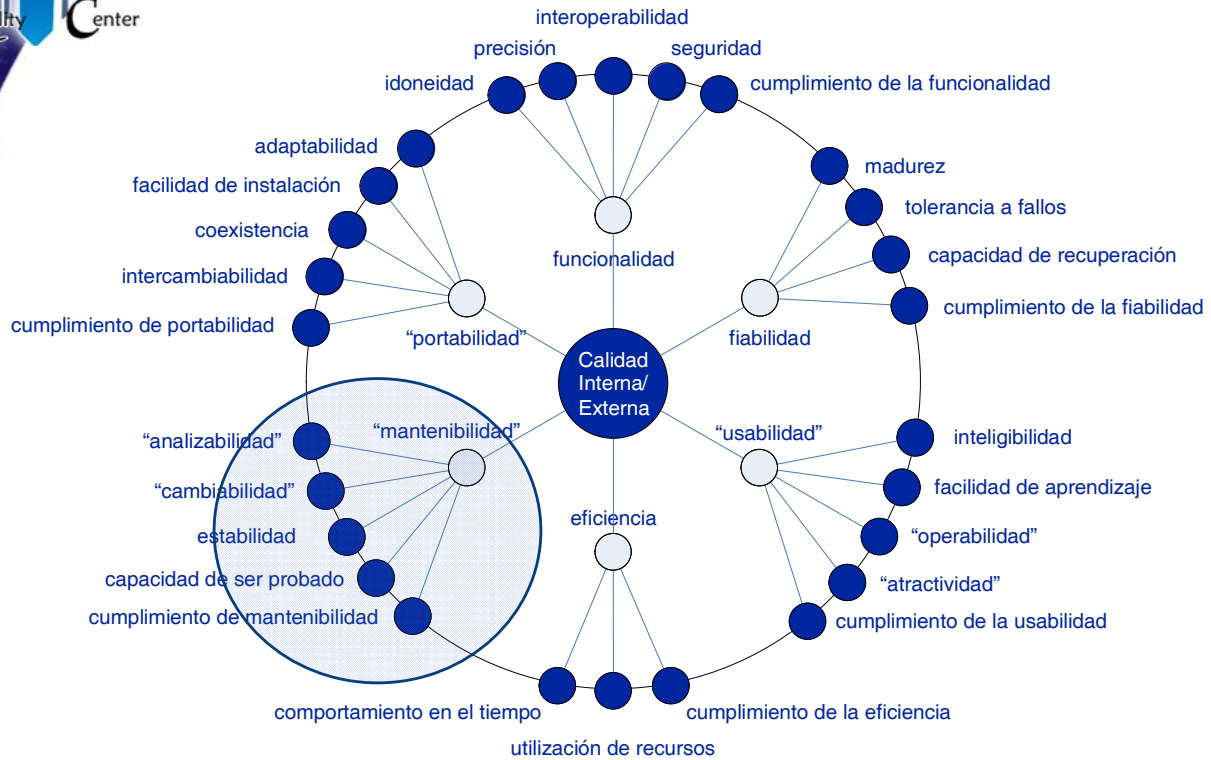
Usabilidad

- Capacidad para ser entendido:** Capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares.
- Capacidad para ser aprendido:** Capacidad del producto software que permite al usuario aprender sobre su aplicación.
- Capacidad para ser operado:** Capacidad del producto software que permite al usuario operarlo y controlarlo.
- Capacidad de atracción:** Capacidad del producto software para ser atractivo al usuario.
- Cumplimiento de la usabilidad:** Capacidad del producto software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.



Eficiencia

- ❑ **Comportamiento temporal:** Capacidad del producto software para proporcionar tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas.
- ❑ **Utilización de recursos:** Capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.
- ❑ **Cumplimiento de la eficiencia:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la eficiencia.

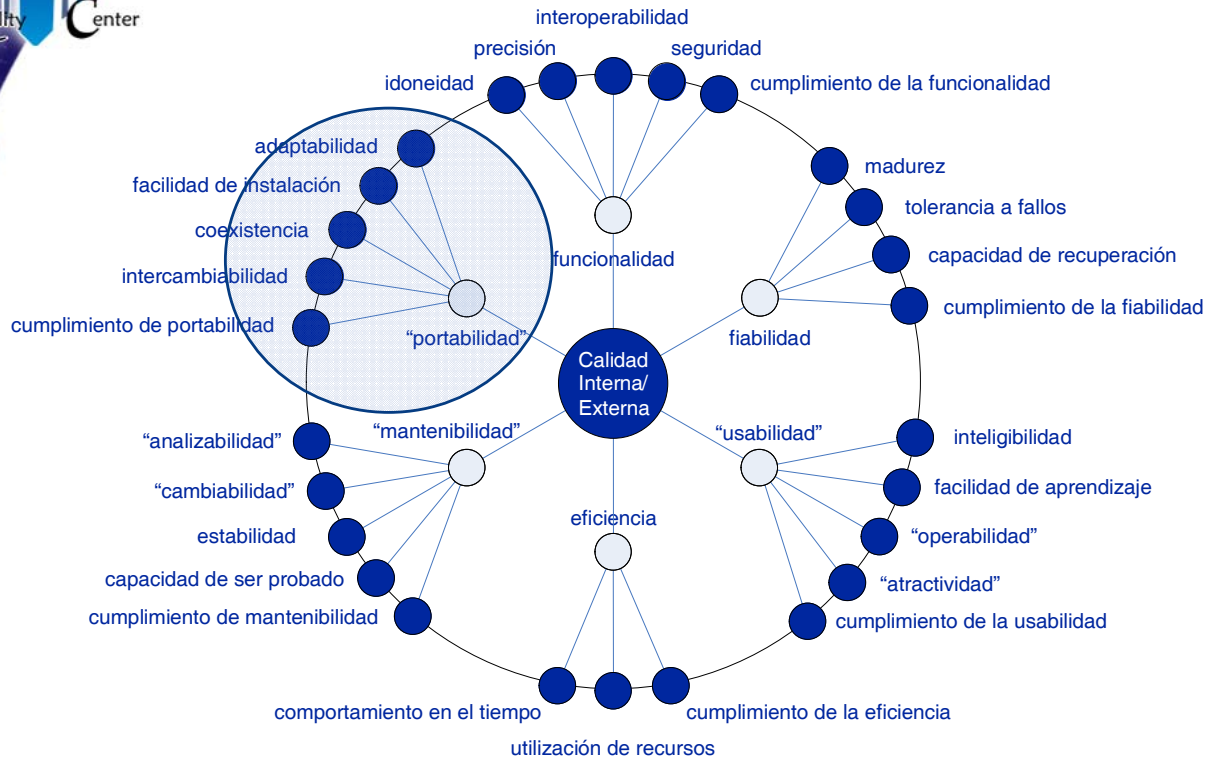


Calidad de Producto Software - ISO/IEC 25000

Mantenibilidad

- Capacidad para ser analizado:** Es la capacidad del producto software para serle diagnosticadas deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas.
- Capacidad para ser cambiado:** Capacidad del producto software que permite que una determinada modificación sea implementada.
- Estabilidad:** Capacidad del producto software para evitar efectos inesperados debidos a modificaciones del software.
- Capacidad para ser probado:** Capacidad del producto software que permite que el software modificado sea validado.
- Cumplimiento de la mantenibilidad:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la mantenibilidad.

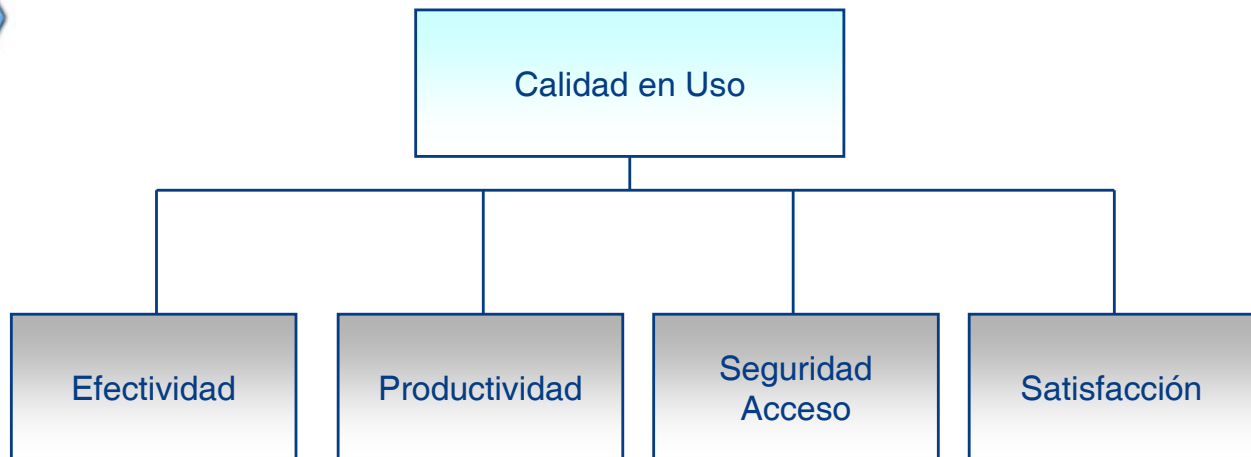
Calidad de Producto Software - ISO/IEC 25000



Portabilidad

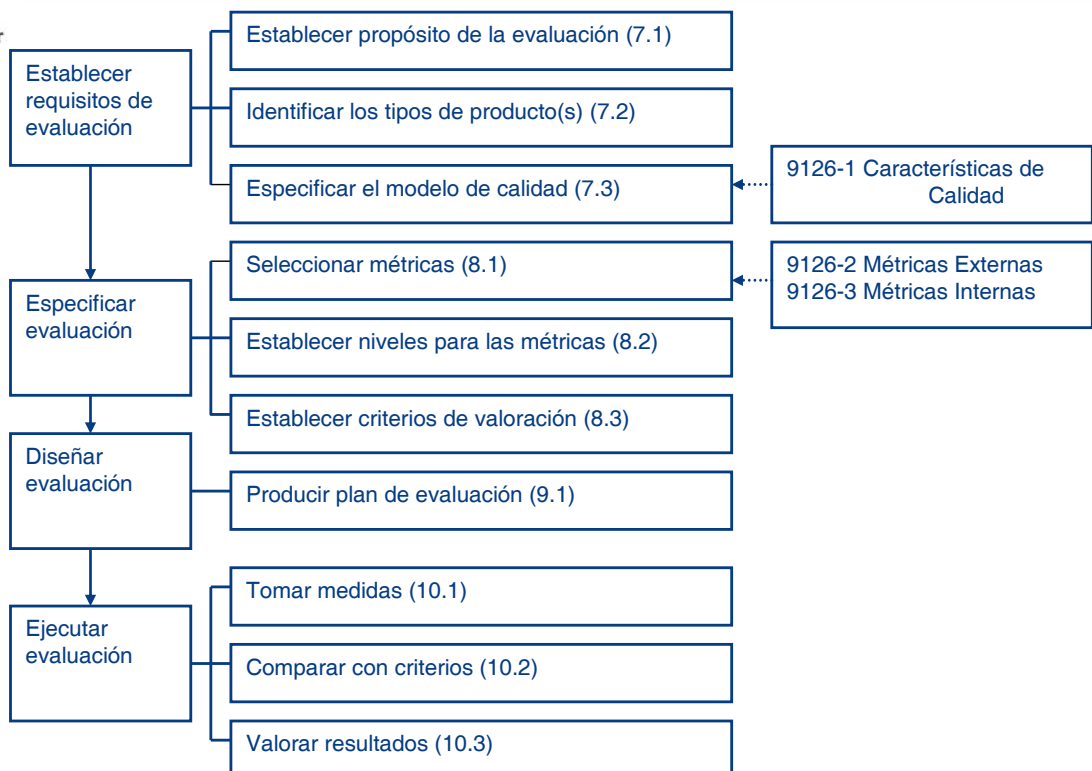
- Adaptabilidad:** Capacidad del producto software para ser adaptado a diferentes entornos especificados, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio software considerado.
- Instalabilidad:** Capacidad del producto software para ser instalado en un entorno especificado.
- Coexistencia:** Capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.
- Capacidad para reemplazar:** Capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno.
- Cumplimiento de la portabilidad:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.

Calidad en Uso



- Norma de 1999
- Actualmente se encuentra en estado 90.60
- Formada por 6 partes
- Establece los procesos para realizar la evaluación de la calidad del producto software
- Directamente relacionada con ISO 9126

- ❑ Parte 1: Visión general
- ❑ Parte 2: Gestión y planificación
- ❑ Parte 3: Proceso para los desarrolladores
- ❑ Parte 4: Proceso para los adquirentes
- ❑ Parte 5: Proceso para los evaluadores
- ❑ Parte 6: Documentación de los módulos de evaluación



SQuaRE

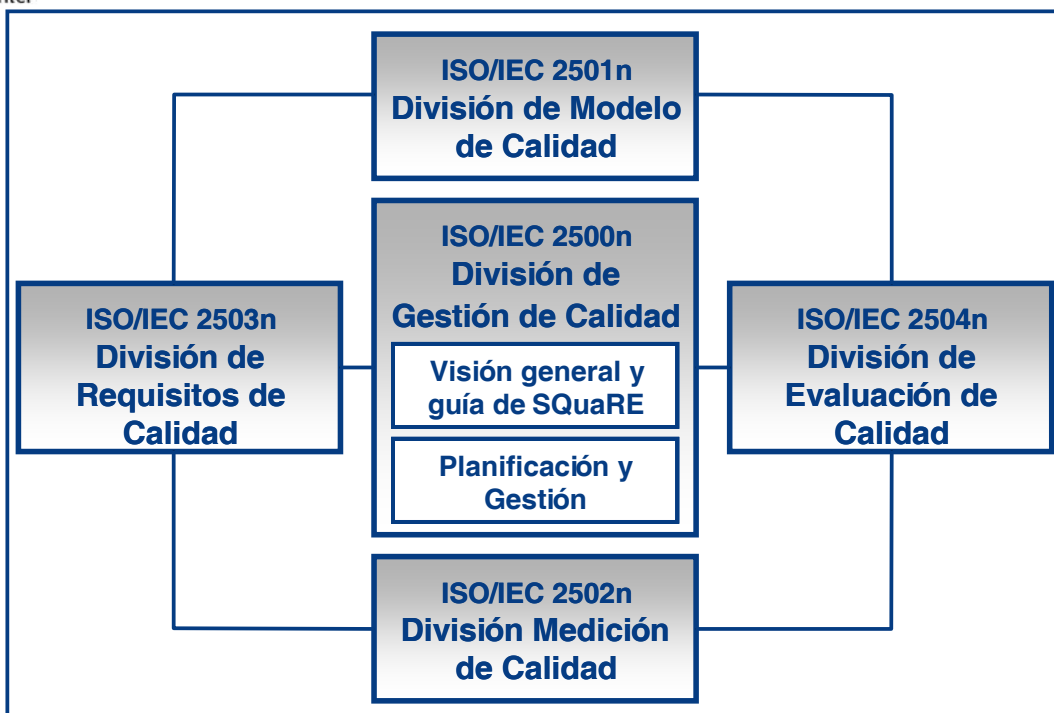
Software Product Quality Requirements and Evaluation

- Nace por las inconsistencias entre ISO 9126 e ISO 14598.

- El objetivo es aglutinar bajo una misma familia el modelo de calidad y el proceso de evaluación.

- Introducción
- Calidad del Producto Software
- Principales normas y estándares de la Calidad del Producto Software
- Familia ISO 25000 (SQuaRE)**
- Herramientas para Evaluación de la Calidad del Producto
- Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software

- ❑ La versión de la primera parte es de 2005.
- ❑ Actualmente está en desarrollo (6 Mayo 2010 ISO 25010 pasa al estado 40.60)
- ❑ Formada por 5 partes (divisiones).
- ❑ Sustituirá a las normas ISO 9126 e ISO 14598.







- ❑ Introducción
- ❑ Calidad del Producto Software
- ❑ Principales normas y estándares de la Calidad del Producto Software
- ❑ Familia ISO 25000 (SQuaRE)
- ❑ **Herramientas para Evaluación de la Calidad del Producto**
- ❑ Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software

Calidad de Producto Software - ISO/IEC 25000

57

Herramientas para Evaluación de la Calidad del Producto

Reflexión



“La construcción de un marco que permita llevar a cabo la medición, requiere tanto de un **soporte metodológico** como de un **soporte tecnológico**”

Lavazza L. 2000. Providing Automated Support for the GQM Measurement Process. IEEE Software 17(3):56-62.

Calidad de Producto Software - ISO/IEC 25000

58

“Para que las métricas puedan ser evaluadas de un **modo práctico, eficiente y exacto** es necesario contar con **herramientas que permitan automatizar la adquisición, la presentación y el análisis** de los valores obtenidos para dichas métricas.”

Giles A, Daich G. 1995. Metrics Tools. Crosstalk, The Journal of Defense Software Engineering.

- Existen múltiples clasificaciones.
- Herramientas de Análisis Dinámico:** aquellas herramientas que realizan el análisis del software ejecutando el código fuente de dicho software.
- Herramientas de Análisis Estático:** aquellas herramientas que llevan a cabo el análisis sin necesidad de ejecutar el software bajo estudio.



NOMBRE	LICENCIA	INTERFAZ	LENGUAJES	ENTRADA	RESULTADO
PMD/CPD	BSD-style	Línea comandos	Java	Código Fuente	Texto plano xml
CheckStyle	LGPL	Línea comandos	Java	Código Fuente	Texto plano xml
Klocwork k7	Software propietario	GUI Línea comandos	C/C++/Java	Byte code Código fuente	HTML
JDepend	BSD	Línea comandos	Java	Byte code	Texto plano XML
JavaNCSS	GNU GPL	Línea comandos	Java	Código fuente	Texto plano XML
McCabe IQ	Software propietario	GUI	Ada, ASM86, C, C#, C++, COBOL, FORTRAN, JAVA, JSP, Perl, PL1, VB, VB.NET	Código fuente	Texto plano Gráficos
Simian	Software propietario	Línea comandos	Java, C#, C, C++, COBOL, Ruby, JSP, ASP, HTML, XML, Visual Basic	Código fuente	Texto plano
CodeSonar	Software propietario	Línea comandos	C/C++ y ADA	Código fuente	HTML
CKJM	Open Source Software	Línea comandos	Java	Byte code	Texto plano

Métricas

Reglas

Pruebas

Codificación

JAVA

JDepend

JavaNCSS



Simian



.Net



FxCop



PartCover

PHP



PHP_CodeSniffer



PHPmd



PHPUnit

Datos del Proyecto

- Nombre:** Ibatis.
- Descripción:** asocia objetos de modelo (JavaBeans) con sentencias SQL o procedimientos almacenados mediante ficheros descriptores XML, simplificando la utilización de bases de datos.
- Número de módulos:** 2
- Tamaño:** 23.528 líneas
- Licencia:** es un proyecto que pertenece y utiliza la licencia de Apache Software Foundation, por lo que es código abierto.
- Página de descarga:** <http://ibatis.apache.org>

Herramientas de Métricas: JavaNCSS

JavaNCSS Metric Results

[package] [object] [method] [explanation]

The following document contains the results of a JavaNCSS metric analysis.
[JavaNCSS web site.](#)

Packages

[package] [object] [method] [explanation]

Packages sorted by NCSS.

Package	Classes	Methods	NCSS	Javadocs	Javadoc lines	Single lines comment	Multi lines comment
org.apache.ibatis.ognl	103	866	8726	272	1778	2871	235
org.apache.ibatis.type	30	107	663	1	6	10	0
org.apache.ibatis.mapping	16	91	631	0	0	4	0
org.apache.ibatis.builder.xml	4	41	594	2	11	4	0
org.apache.ibatis.migration.commands	11	60	561	0	0	2	0
org.apache.ibatis.executor	11	87	558	4	27	9	0
org.apache.ibatis.executor.resultset	3	43	470	0	0	43	0
org.apache.ibatis.cache.decorators	9	100	435	4	14	2	0
org.apache.ibatis.datasource.pooled	4	83	427	30	153	10	0

Herramientas de Métricas: JDepend

```

- <Package name="org.apache.ibatis.plugin">
- <Stats>
  <TotalClasses>7</TotalClasses>
  <ConcreteClasses>4</ConcreteClasses>
  <AbstractClasses>3</AbstractClasses>
  <Ca>2</Ca>
  <Ce>6</Ce>
  <A>0.43</A>
  <I>0.75</I>
  <D>0.18</D>
  <V>1</V>
</Stats>
- <AbstractClasses>
  <Class sourceFile="Interceptor.java">org.apache.ibatis.plugin.Interceptor</Class>
  <Class sourceFile="Intercepts.java">org.apache.ibatis.plugin.Intercepts</Class>
  <Class sourceFile="Signature.java">org.apache.ibatis.plugin.Signature</Class>
</AbstractClasses>
- <ConcreteClasses>
  <Class sourceFile="InterceptorChain.java">org.apache.ibatis.plugin.InterceptorChain</Class>
  <Class sourceFile="Invocation.java">org.apache.ibatis.plugin.Invocation</Class>
  <Class sourceFile="Plugin.java">org.apache.ibatis.plugin.Plugin</Class>
  <Class sourceFile="PluginException.java">org.apache.ibatis.plugin.PluginException</Class>
</ConcreteClasses>
- <DependsUpon>
  <Package>java.lang</Package>
  <Package>java.lang.annotation</Package>
  <Package>java.lang.reflect</Package>
  <Package>java.util</Package>
  <Package>org.apache.ibatis.exceptions</Package>
  <Package>org.apache.ibatis.reflection</Package>
</DependsUpon>
- <UsedBy>
  <Package>org.apache.ibatis.builder.xml</Package>
  <Package>org.apache.ibatis.session</Package>
</UsedBy>
</Package>
  
```

Herramientas de Métricas: Simian

```

<set lineCount="32">
  <block sourceFile="C:\ibatis\ibatis-3-core\src\main\java\org\apache\ibatis\ognl\OgnlParser.java"
    startLineNumber="2238" endLineNumber="2269"/>
  <block sourceFile="C:\ibatis\ibatis-3-core\src\main\java\org\apache\ibatis\ognl\OgnlParser.java"
    startLineNumber="2322" endLineNumber="2353"/>
</set>
<set lineCount="34">
  <block sourceFile="C:\ibatis\ibatis-3-core\src\test\java\domain\blog\Author.java"
    startLineNumber="55" endLineNumber="107"/>
  <block sourceFile="C:\ibatis\ibatis-3-core\src\test\java\domain\blog\ImmutableAuthor.java"
    startLineNumber="22" endLineNumber="74"/>
</set>
<set lineCount="38">
  <block sourceFile="C:\ibatis\ibatis-3-core\src\main\java\org\apache\ibatis\ognl\ParseException.java"
    startLineNumber="147" endLineNumber="187"/>
  <block sourceFile="C:\ibatis\ibatis-3-core\src\main\java\org\apache\ibatis\ognl\TokenMgrError.java"
    startLineNumber="40" endLineNumber="80"/>
</set>
<summary duplicateFileCount="27" duplicateLineCount="1393" duplicateBlockCount="74" totalFileCount="568"
  totalRawLineCount="51281" totalSignificantLineCount="26130" processingTime="1297"/>
</check>
</simian>
  
```

Herramientas de Reglas: PMD

```
<?xml version="1.0" encoding="UTF-8" ?>
<pmd version="4.2.2" timestamp="2010-05-26T10:04:20.597">
- <file name="C:\ibatis\ibatis-3-
  core\src\main\java\org\apache\ibatis\binding\MapperProxy.java">
  <violation beginline="57" endline="61" begincolumn="9" endcolumn="7"
    rule="EmptyCatchBlock" ruleset="Basic Rules"
    package="org.apache.ibatis.binding" class="MapperProxy"
    method="findDeclaringInterface"
    externalInfoUrl="http://pmd.sourceforge.net/rules/basic.html#EmptyCatchBlock"
    priority="3">Avoid empty catch blocks</violation>
  </file>
- <file name="C:\ibatis\ibatis-3-
  core\src\main\java\org\apache\ibatis\builder\MapperBuilderAssistant.java">
  <violation beginline="323" endline="332" begincolumn="7" endcolumn="7"
    rule="CollapsibleIfStatements" ruleset="Basic Rules"
    package="org.apache.ibatis.builder" class="MapperBuilderAssistant"
    method="parseCompositeColumnName"
    externalInfoUrl="http://pmd.sourceforge.net/rules/basic.html#CollapsibleIfStatements"
    priority="3">These nested if statements could be combined</violation>
  <violation beginline="342" endline="344" begincolumn="9" endcolumn="7"
    rule="EmptyCatchBlock" ruleset="Basic Rules"
    package="org.apache.ibatis.builder" class="MapperBuilderAssistant"
    method="resolveResultJavaType"
    externalInfoUrl="http://pmd.sourceforge.net/rules/basic.html#EmptyCatchBlock"
    priority="3">Avoid empty catch blocks</violation>
  </file>
```

Herramientas de Pruebas: Emma

EMMA Coverage Report (generated Wed May 26 10:04:22 CEST 2010)

[all classes]

OVERALL COVERAGE SUMMARY

name	class, %	method, %	block, %	line, %
all classes	94% (343/365)	78% (2146/2739)	71% (43910/61868)	72% (9527.3/13171)

OVERALL STATS SUMMARY

total packages: 48
total executable files: 289
total classes: 365
total methods: 2739
total executable lines: 13171

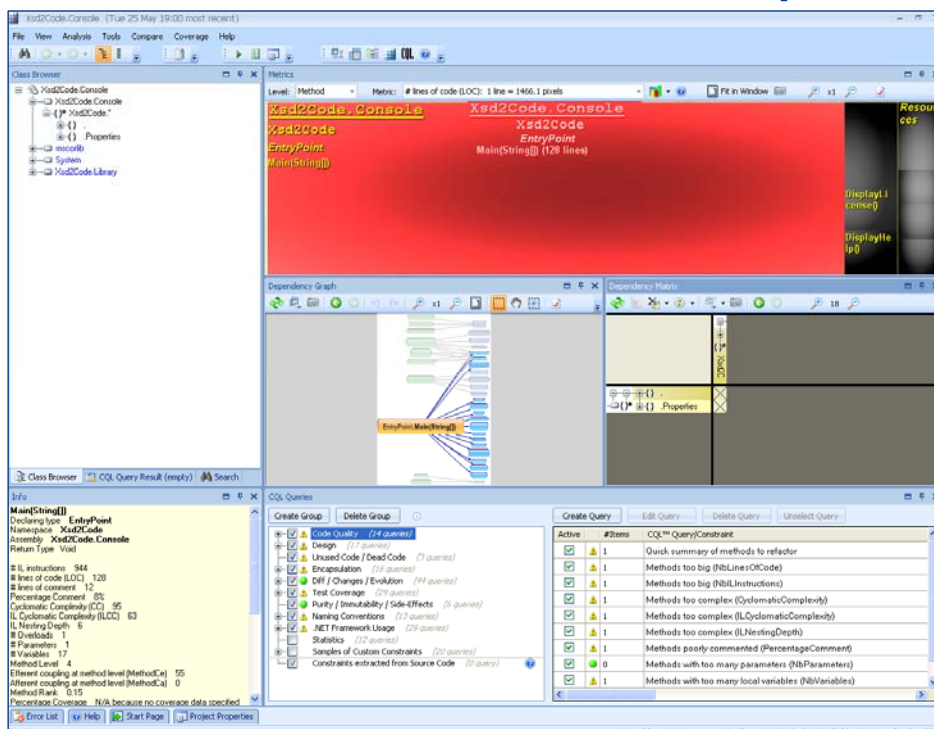
COVERAGE BREAKDOWN BY PACKAGE

name	class, %	method, %	block, %	line, %
org.apache.ibatis.logging.slf4j	100% (1/1)	50% (3/6)	50% (16/32)	50% (6/12)
org.apache.ibatis.reflection.factory	100% (1/1)	100% (6/6)	52% (85/165)	62% (21/34)
org.apache.ibatis.reflection.wrapper	100% (5/5)	94% (33/35)	53% (503/943)	60% (119.5/199)
org.apache.ibatis.datasource.pooled	100% (4/4)	69% (59/85)	57% (914/1607)	63% (218.2/348)
org.apache.ibatis.exceptions	67% (2/3)	50% (5/10)	59% (27/46)	50% (9/18)
org.apache.ibatis.ognl	94% (97/103)	61% (551/899)	60% (17602/29173)	60% (3671.5/6145)
org.apache.ibatis.datasource.unpooled	100% (3/3)	53% (23/43)	61% (255/416)	65% (82.2/126)
org.apache.ibatis.transaction.jdbc	100% (2/2)	100% (10/10)	63% (61/97)	83% (24/29)
org.apache.ibatis.session.defaults	100% (4/4)	79% (34/43)	68% (430/635)	76% (98.5/130)
org.apache.ibatis.executor.parameter	100% (1/1)	100% (3/3)	72% (125/174)	82% (27.2/33)
org.apache.ibatis.reflection.property	100% (2/2)	57% (8/14)	72% (141/196)	77% (25.4/33)
org.apache.ibatis.logging	100% (7/7)	81% (21/26)	72% (113/157)	74% (40/54)
org.apache.ibatis.datasource.jndi	100% (1/1)	100% (4/4)	73% (85/116)	85% (23.8/28)
org.apache.ibatis.migration.commands	92% (11/12)	94% (59/63)	74% (1714/2311)	81% (348.8/432)

Datos del Proyecto

- ❑ **Nombre:** Xsd2Code.
- ❑ **Descripción:** Herramienta que permite generar clases de negocio a partir de esquemas XML (ficheros xsd). Es una evolución de la herramienta xsd.exe que viene con la instalación de Visual Studio.
- ❑ **Número de módulos:** 2. El estudio se ha realizado con el módulo “*Console*”.
- ❑ **Tamaño:** 1.051 líneas
- ❑ **Licencia:** Library General Public License (LGPL).
- ❑ **Página de descarga:** <http://xsd2code.codeplex.com/>

Herramientas de Métricas: NDepend



The screenshot displays the NDepend application interface. The top window shows the 'Metrics' view for the 'Xsd2Code.Console' project, listing metrics such as 'Lines of Code (LOC)', 'Cyclomatic Complexity (CC)', and 'Lines of Comments'. Below this, a 'Dependency Graph' visualizes the relationships between classes and methods. The bottom section features a 'CQL Queries' panel with a list of active queries, including 'Code Quality', 'Design', 'Unused Code / Dead Code', and 'Encapsulation'. The rightmost pane shows a 'Properties' window for a selected element.

Herramientas de Reglas: FxCop

Lista de errores

0 errores 104 advertencias 0 mensajes

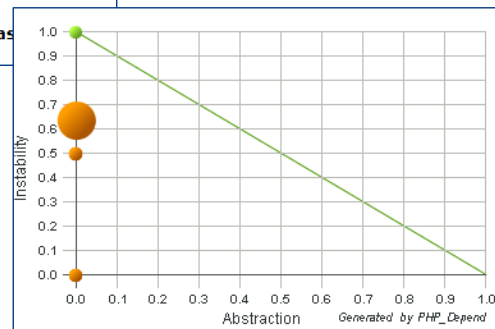
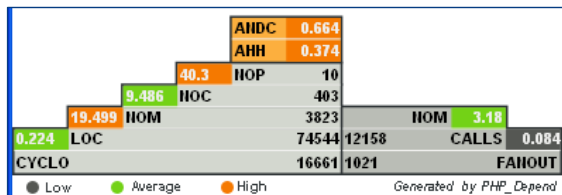
	Descripción	Archivo	Línea	Columna	Proyecto
1	La opción 'keyfile' invalida el atributo 'System.Reflection.AssemblyKeyFileAttribute' especificado en el archivo de código fuente o módulo agregado				Xsd2Code.Library
2	CA1020 : Microsoft.Design : Considere combinar los tipos definidos en 'Xsd2Code' con otro espacio de nombres.				Xsd2Code.Library
3	CA1014 : Microsoft.Design : Marque 'Xsd2Code.Library.dll' con CLSCompliant(true) porque expone tipos visibles externamente.				Xsd2Code.Library
4	CA1017 : Microsoft.Design : Puesto que 'Xsd2Code.Library.dll' expone tipos visibles externamente, márkelo con ComVisible(false) en el nivel de ensamblado y después marque todos los tipos del ensamblado que se deben exponer a clientes COM con ComVisible(true).				Xsd2Code.Library
5	CA1824 : Microsoft.Performance : Puesto que el ensamblado 'Xsd2Code.Library.dll' contiene un archivo de recursos basado en ResX, márkelo con el atributo NeutralResourceLanguage, especificando el lenguaje de los recursos del ensamblado. Esto puede mejorar el rendimiento de las búsquedas la primera vez que se recupera un recurso.				Xsd2Code.Library
6	CA1053 : Microsoft.Design : Quite los constructores públicos de 'Generator'.	Generator.cs	24		Xsd2Code.Library
7	CA1811 : Microsoft.Performance : 'Generator.Process(string, string, GenerationLanguage, CollectionType, bool, bool, List<NamespaceParam>, string, bool, string, string, string, bool, TargetFramework)' parece no tener ningún llamador protegido o público de nivel superior.	Generator.cs	52		Xsd2Code.Library
8	CA1801 : Microsoft.Usage : Nunca se usa el parámetro 'targetNamespace' de 'Generator.Process(string, string, GenerationLanguage, CollectionType, bool, bool, List<NamespaceParam>, string, bool, string, string, string, bool, TargetFramework)'. Quite el parámetro o utilícelo en el cuerpo del método.	Generator.cs	52		Xsd2Code.Library
9	CA1801 : Microsoft.Usage : Nunca se usa el parámetro 'xsdFile' de 'Generator.Process(string, string, GenerationLanguage, CollectionType, bool, bool, List<NamespaceParam>, string, bool, string, string, string, bool, TargetFramework)'. Quite el parámetro o utilícelo en el cuerpo del método.	Generator.cs	52		Xsd2Code.Library
10	CA1031 : Microsoft.Design : Modifique 'Generator.Process(GeneratorParams)' para capturar una excepción más específica que 'Exception' o vuelva a producir la excepción.	Generator.cs	88		Xsd2Code.Library
11	CA2201 : Microsoft.Usage : 'Generator.Validate(object, ValidationEventArgs)' crea una excepción de tipo 'Exception', un tipo de excepción que no es suficientemente específica y que nunca debería producir el código de usuario. Si se produce esta instancia de excepción, use un tipo de excepción diferente.	Generator.cs	145		Xsd2Code.Library

Datos del Proyecto

- Nombre:** Web2Project.
- Descripción:** Aplicación Web para el control y gestión de proyectos.
- Tamaño:** 112.431 líneas.
- Licencia:** Código Abierto.
- Página de descarga:** <http://web2project.net/>

Herramientas de Métricas: PHPDepend

```
<?xml version="1.0" encoding="UTF-8" ?>
<metrics generated="2010-05-17T16:32:49" pdepend="0.9.12" ahh="0.3739837398374"
  andc="0.66397849462366" calls="12158" ccn="15075" ccn2="16661" cloc="26396"
  clsa="0" clsc="403" eloc="74544" fanout="1021" leafs="344" loc="112431" maxDIT="3"
  ncloc="86035" noc="403" nof="317" noi="1" nom="3506" nop="10" roots="35">
- <files>
  <file name="C:\work1\web2project-1.3\src\classes\CustomFields.class.php"
    cloc="35" eloc="549" loc="677" ncloc="642" />
  <file name="C:\work1\web2project-1.3\src\classes\ajax.class.php" cloc="3"
    eloc="12" loc="16" ncloc="13" />
  <file name="C:\work1\web2project-1.3\src\classes\authenticator.class.php"
    cloc="32" eloc="266" loc="333" ncloc="301" />
  <file name="C:\work1\web2project-1.3
    \src\classes\customfieldsparser.class.php" cloc="6" eloc="214" loc="248"
    ncloc="242" />
  <file name="C:\work1\web2project-1.3\src\classes\date.class.php"
    cloc="148" eloc="348" loc="576" ncloc="428" />
  <file name="C:\work1\web2project-1.3\src\classes\event_queue.class.php"
    cloc="44" eloc="142" loc="205" ncloc="161" />
```



Herramientas de Reglas: PHPmd

```
<?xml version="1.0" encoding="UTF-8" ?>
- <pmd version="0.2.5" timestamp="2010-05-17T16:41:09+02:00">
- <file name="C:\work1\web2project-1.3\src\classes\CustomFields.class.php">
  <violation beginline="9" endline="156" rule="TooManyMethods" ruleset="Code Size
    Rules" package="+global"
    externalInfoUrl="http://phpmd.org/rules/codesize.html#toomanymethods"
    class="CustomField" priority="3">This class has too many methods, consider
    refactoring it.</violation>
  <violation beginline="29" endline="36"
    rule="ConstructorWithNameAsEnclosingClass" ruleset="Naming Rules"
    package="+global"
    externalInfoUrl="http://phpmd.org/rules/naming.html#constructorwithnameasenclosi"
    class="CustomField" method="CustomField" priority="3">Classes should not
    have a constructor method with the same name as the class</violation>
  <violation beginline="40" endline="40" rule="ShortVariable" ruleset="Naming Rules"
    externalInfoUrl="http://phpmd.org/rules/naming.html#shortvariable"
    priority="3">Avoid variables with short names like $db</violation>
  <violation beginline="40" endline="40" rule="UnusedLocalVariable"
    ruleset="Unused Code Rules"
    externalInfoUrl="http://phpmd.org/rules/unusedcode.html#unusedlocalvariable"
    priority="3">Avoid unused local variables such as '$db'.</violation>
  <violation beginline="41" endline="41" rule="ShortVariable" ruleset="Naming Rules"
    externalInfoUrl="http://phpmd.org/rules/naming.html#shortvariable"
    priority="3">Avoid variables with short names like $n</violation>
```

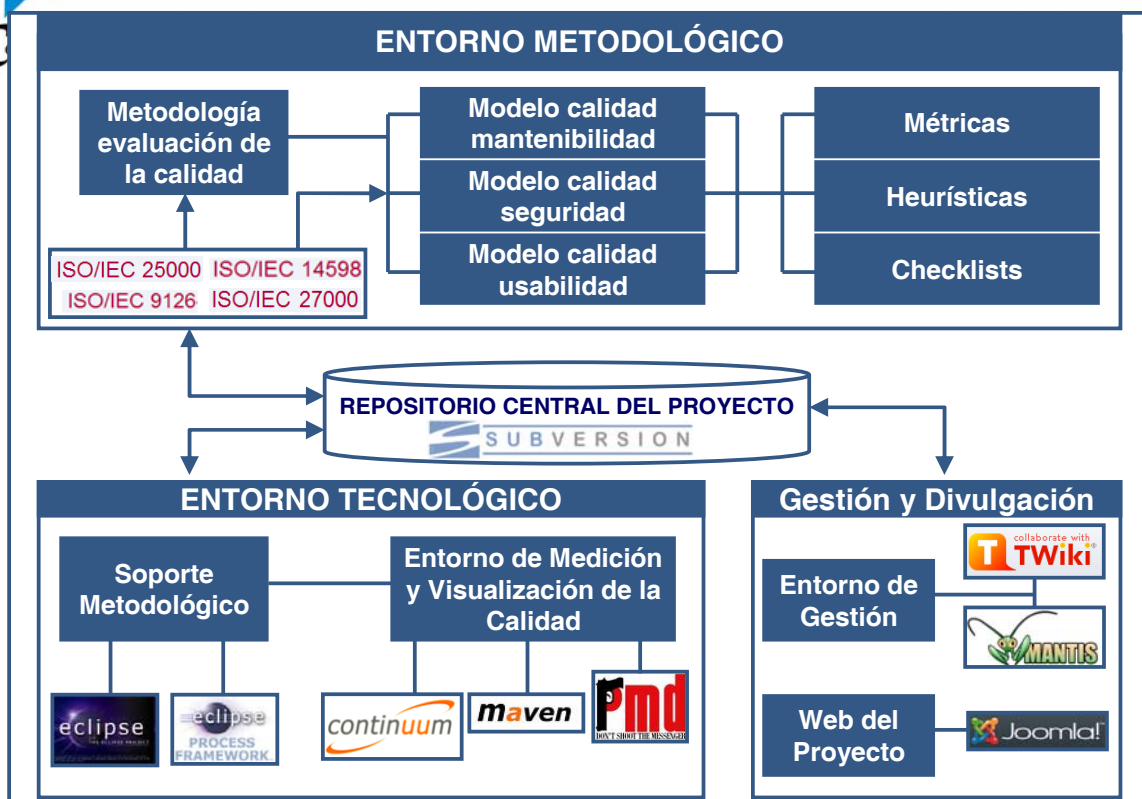
PROBLEMA	IMPACTO
<ul style="list-style-type: none"> <input type="checkbox"/> La mayoría de las herramientas están diseñadas para el trabajo en modo interactivo 	<ul style="list-style-type: none"> <input type="checkbox"/> Alto coste en tiempo y por lo tanto menor número de evaluaciones posibles
<ul style="list-style-type: none"> <input type="checkbox"/> La mayoría de las herramientas solo están especializadas en un subconjunto de métricas 	<ul style="list-style-type: none"> <input type="checkbox"/> Necesidad de utilizar varias herramientas y de formatear posteriormente los resultados
<ul style="list-style-type: none"> <input type="checkbox"/> La mayoría de las herramientas no personalizan los resultados en función del público objetivo (estratégico, táctico, operacional) 	<ul style="list-style-type: none"> <input type="checkbox"/> Necesidad de estudiar y modificar los resultados manualmente en función de a quién se le presenten
<ul style="list-style-type: none"> <input type="checkbox"/> La mayoría de herramientas necesita ser configurada para cada proyecto e incluso para cada evaluación 	<ul style="list-style-type: none"> <input type="checkbox"/> Necesidad de hacer diferentes configuraciones por cada tipo de herramienta con los correspondientes costes de tiempo

- Único entorno para la medición y evaluación del software, ofreciendo una **solución global**.
- Basado en un **componente metodológico** que establezca y defina los requisitos y pasos.
- Con un **componente tecnológico** integrado y **configurable** para **automatizar la adquisición, análisis y presentación de resultados**.
- Orientado a los tres niveles, **operativo, táctico y estratégico**, tanto para **fábricas de software** como para empresas que **externalizan el desarrollo**.

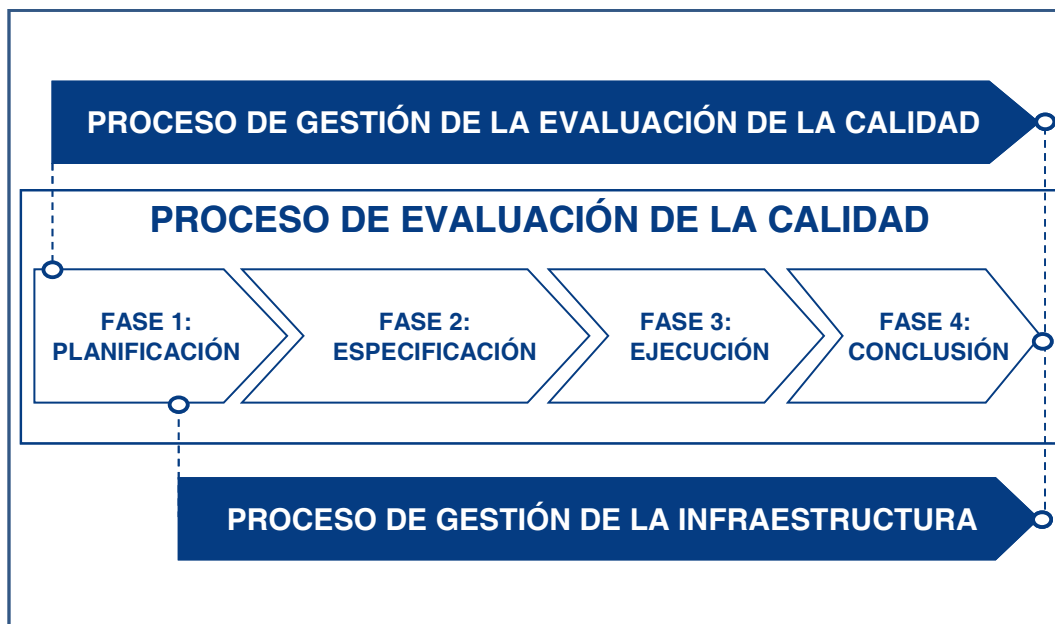
- ❑ Introducción
- ❑ Calidad del Producto Software
- ❑ Principales normas y estándares de la Calidad del Producto Software
- ❑ Familia ISO 25000 (SQuaRE)
- ❑ Herramientas para Evaluación de la Calidad del Producto
- ❑ **Ejemplo de Entorno para la Evaluación de la Calidad del Producto Software**

Ejemplo de Entorno para Evaluación de la Calidad del Producto

Propuesta: Caso Real



ENTORNO METODOLÓGICO		
Metodología para la evaluación de la calidad	Diseño (UML)	Código { Java .Net C PHP
Mantenibilidad	Modelo para la Mantenibilidad	
	Métricas, checklists y heurísticas para la mantenibilidad del diseño	Métricas, checklists y heurísticas para la mantenibilidad del código
Seguridad	Modelo para la Seguridad	
	Métricas, checklists y heurísticas para la seguridad del diseño	Métricas, checklists y heurísticas para la seguridad del código
Usabilidad	Modelo para la Usabilidad	
	Métricas, checklists y heurísticas para la usabilidad del diseño	Métricas, checklists y heurísticas para la usabilidad del código



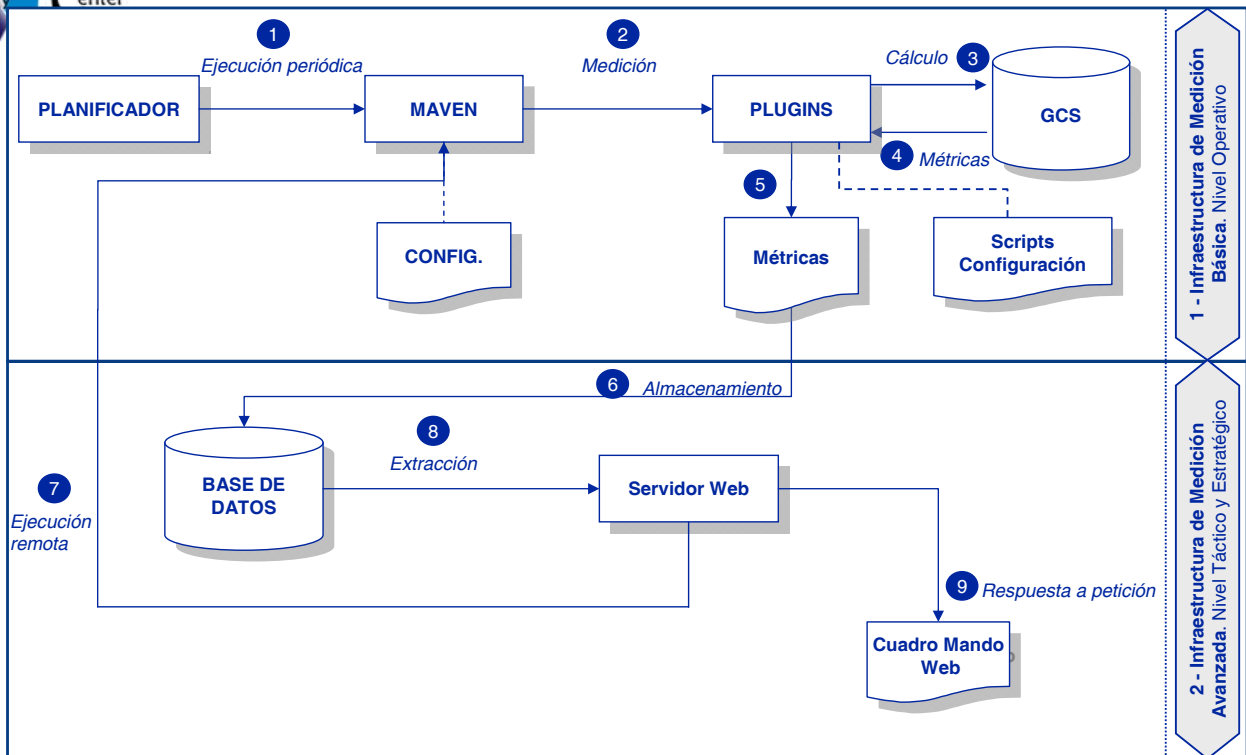
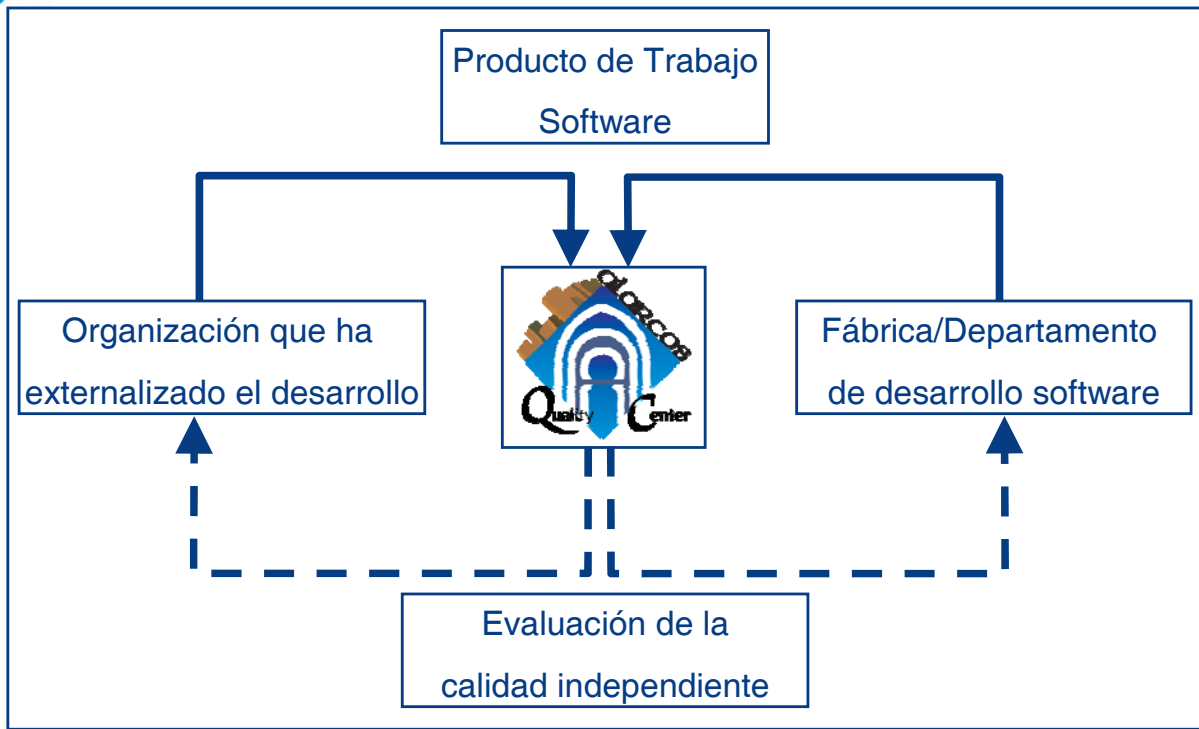












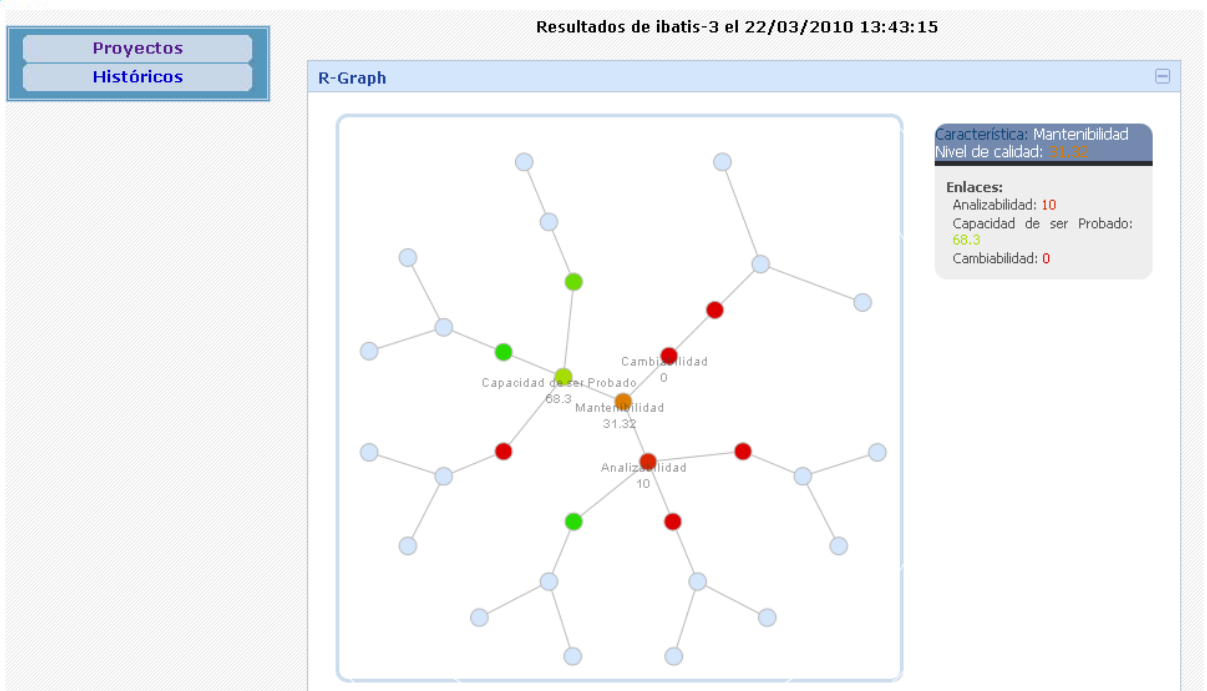
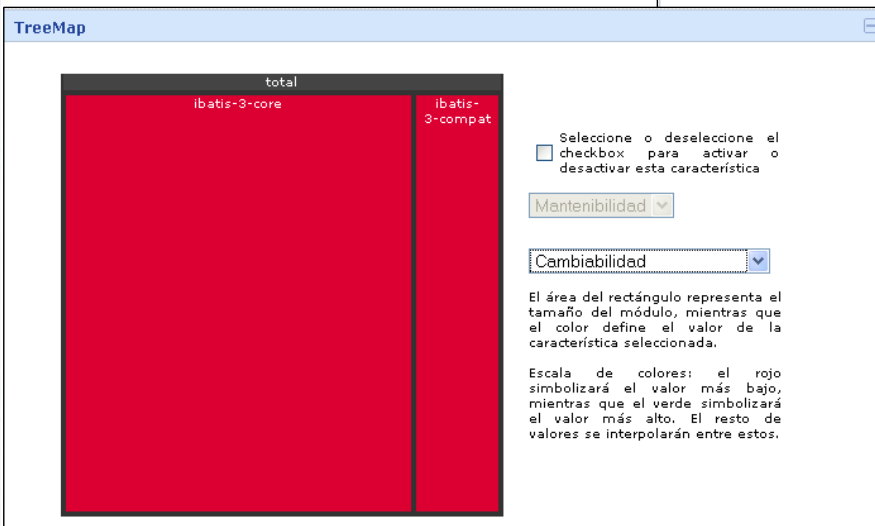
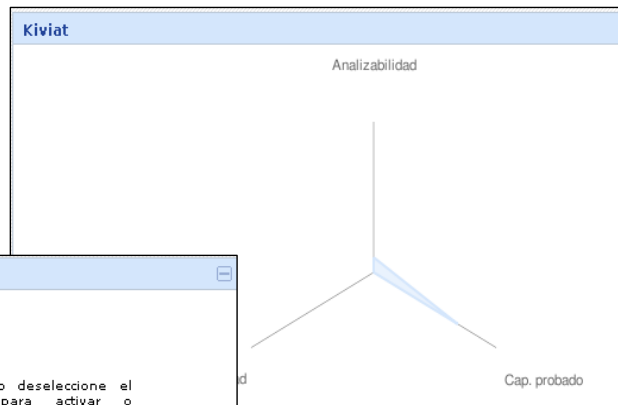
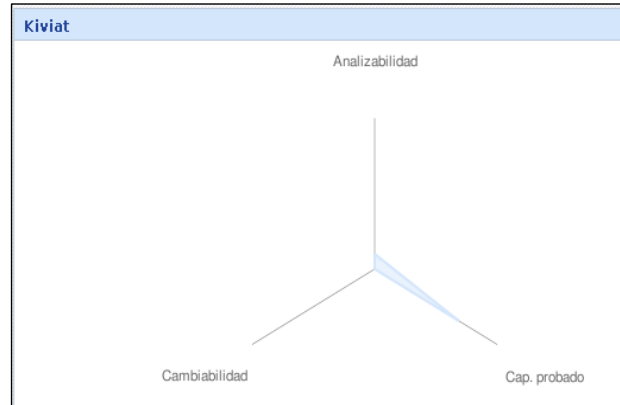


Tabla TreeMap

Proyecto	Mantenibilidad	Analizabilidad	Cap. de ser Probado	Cambiabilidad	
ibatis-3	 31.32	 10	 68.3	 0	Ver Histórico
Xsd2Code.Console	 63.75	 57.5	 0	 70	Ver Histórico
Web2Project	 68.3	 58.5	 0	 33.32	Ver Histórico





Métricas obtenidas	
Métricas de nivel 0	
Métrica	Valor
Mantenibilidad	31.32
Métricas de nivel 1	
Métrica	Valor
Analizabilidad	10
Capacidad de ser Probado	68.3
Cambiabilidad	0
Métricas de nivel 2	
Métrica	Valor
Calidad Normalizada de Densidad de Complejidad Ciclomática	0
Calidad Normalizada de Densidad de Código Duplicado	0
Calidad Normalizada de Densidad de Violaciones de PMD	100
Calidad Normalizada de Densidad de Pruebas Unitarias	0
Calidad Normalizada de Densidad de Errores de Pruebas Unitarias	100
Calidad Normalizada de Ratio de Cobertura de Pruebas Unitarias	83.28
Calidad Normalizada de Densidad de Ciclos	0

Métricas de nivel 3	
Métrica	Valor
Densidad de Complejidad Ciclomática	0.4
Densidad de Código Duplicado	0.05
Densidad de Violaciones de PMD	0.02
Densidad de Pruebas Unitarias	0.13
Densidad de Errores de Pruebas Unitarias	0
Ratio de Cobertura de Pruebas Unitarias	72
Densidad de Ciclos	0.92
Métricas de nivel 4	
Métrica	Valor
Complejidad Ciclomática	7467
Sentencias de Código que no son Comentarios	18901
Líneas Duplicadas	1393
Líneas Significantes	26130
Violaciones de PMD	287
Pruebas Unitarias	979
Errores de Pruebas Unitarias	0
Cobertura de Líneas de Pruebas Unitarias	72
Ciclos	45
Paquetes	49

- Proporciona un entorno de medición de la calidad **acorde a normas internacionales** (ISO 25000/ ISO 9126).
- Permite realizar la evaluación **tanto del código fuente como de los modelos UML**.
- Formado por un **entorno metodológico y tecnológico**.
- Evaluación de la calidad del producto **automática y periódica**.
- Utiliza software libre, no hay licencias de uso**.
- Adaptable a la necesidad del cliente** (herramientas, métricas, modelos y metodología).

- Importancia creciente de la calidad del software**.
- Necesidad de la evaluación del producto software**.
- Importancia de un entorno metodológico y tecnológico para llevar a cabo al medición y evaluación**.
- Problemas de las herramientas existentes en la actualidad**.
- Propuesta de entorno metodológico y tecnológico integrado para la evaluación de la calidad desde las primeras fases del ciclo de vida**.

**“If you don’t know where you are,
a map won’t help”**

Watts H. Humphrey

**“If you don’t know where you are
(and where you want to go), a map
won’t help”**

Watts H. Humphrey

Webs Interesantes

- <http://iso25000.com/> (Portal en español de la ISO 25000)
- <http://www.iso.org> (Portal oficial de ISO)
- <http://twitter.com/calidadsoftware>

Webs de Herramientas

- <http://maven.apache.org/>
- <http://npanday.codeplex.com/Wikipage>
- <http://www.aptest.com/resources.html>

Preguntas...





XI CURSOS DE VERANO DE SANTANDER

CALIDAD DE PROCESOS Y PRODUCTOS SOFTWARE

**CALIDAD DEL PRODUCTO SOFTWARE
ISO/IEC 25000**



16 Julio 2010

D. Moisés Rodríguez Monje